EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

Instituto Politécnico Nacional
"La Técnica al Servicio de la Patria"

Research in Computing Science

Vol. 150 No. 5
May 2021

Research in Computing Science

Research in Computing Science, Año 20, Volumen 150, No. 5, mayo de 2021, es una publicación mensual, editada por el Instituto Politécnico Nacional, a través del Centro de Investigación en Computación. Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, Ciudad de México, Tel. 57 29 60 00, ext. 56571. https://www.rcs.cic.ipn.mx. Editor responsable: Dr. Grigori Sidorov. Reserva de Derechos al Uso Exclusivo del Título No. 04-2019-082310242100-203. ISSN: en trámite, ambos otorgados por el Instituto Politécnico Nacional de Derecho de Autor. Responsable de la última actualización de este número: el Centro de Investigación en Computación, Dr. Grigori Sidorov, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Fecha de última modificación 01 de mayo de 2021.

Research in Computing Science, year 20, Volume 150, No. 5, May 2021, is published monthly by the Center for Computing Research of IPN.

Volume 150(5)

Advances in Artificial Intelligence

Noé A. Castro-Sánchez (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2021

ISSN: in process

Formerly ISSNs: 1870-4069, 1665-9899

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.rcs.cic.ipn.mx
http://www.ipn.mx
http://www.cic.ipn.mx

Indexed in LATINDEX, DBLP and Periodica

Electronic edition

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

Estudio de hiperparámetros de modelos neuronales en la generación de frases literarias

Luis-Gil Moreno-Jiménez[1], Juan-Manuel Torres-Moreno[1,4], Carlos-Emiliano González-Gallardo[2], Roseli S. Wedemann[3]

[1] Avignon Université,
Laboratoire Informatique d'Avignon,
France

[2] Sorbonne Université,
Laboratoire STIH,
France

[3] Universidade do Estado do Rio de Janeiro,
Brazil

[4] Polytechnique Montréal,
Canada

luis-gil.moreno-jimenez@alumni.univ-avignon.fr,
juan-manuel.torres@univ-avignon.fr,
carlos-emiliano.gonzalez-gallardo@sorbonne-universite.fr,
roseli@ime.uerj.br

Resumen. Los algoritmos de Redes Neuronales son ampliamente utilizados en técnicas de Inteligencia Artificial (IA) para la resolución de problemas en diferentes campos de la ciencia que no pueden ser resueltos por la computación simbólica tradicional. En este trabajo, presentamos un conjunto de experimentos que involucran la implementación de varios modelos basados en word2vec, una representación de palabras basada en la semántica distribucional. Nuestros experimentos se desarrollaron con el propósito de establecer un conjunto de métricas replicables para el entrenamiento de un modelo word2vec capaz de ser integrado a un modelo para generar oraciones literarias coherentes y con un contexto bien definido. En este trabajo, presentamos resultados alentadores del análisis de oraciones generadas con nuestros modelos mediante evaluaciones manuales.

Palabras clave: Redes neuronales artificiales, procesamiento del lenguaje natural, generación de textos literarios, word2vec.

Study of Hyperparameters of Neural Models in the Generation of Literary Sentences

Abstract. Neural Network algorithms are widely used in Artificial Intelligence (AI) techniques for solving problems in different fields of science that cannot be solved by traditional symbolic computation. In this paper, we present a set of

experiments involving the implementation of various models based on word2vec, a word representation based on distributional semantics. Our experiments were developed with the purpose of establishing a set of replicable metrics for training a word2vec model capable of being integrated into a model to generate coherent literary sentences with a well-defined context. In this paper, we present encouraging results from the analysis of sentences generated with our models using manual evaluations.

Keywords: Artificial neural networks, natural language processing, literary text generation, word2vec.

1. Introducción

Los modelos basados en Redes Neuronales (NN por sus siglas en inglés) han sido ampliamente usados en diversos campos de estudio, dada su eficiencia y efectividad en métodos predictivos, para el análisis de datos y tareas de tipo cognitivas. Estos modelos son idóneos cuando se requiere procesar y analizar una cantidad importante de datos. En Procesamiento de Lenguaje Natural (PLN), las NN son comúnmente usadas en tareas de clasificación de textos, recuperación de información, traducción automática, entre otras [6, 9, 17].

En el campo de la literatura, estos modelos neuronales son usados para crear estructuras complejas del lenguaje para la generación de texto literario [18, 20] en un área denominada Creatividad Computacional [3]. En Creatividad Computacional las NN son entrenadas con cantidades importantes de datos a partir de ciertos artefactos artísticos como: la pintura, la música, novelas, cuentos, etc. La intención es detectar patrones para imitar el Proceso Creativo[5] que permite la creación de cada obra artística.

En este trabajo analizamos cómo los diferentes hiperparámetros (dimensión, ventana contextual, ocurrencias, etc.) pueden ser indicadores interesantes para el entrenamiento de modelos basados en representaciones vectoriales Word2vec [8] utilizados en la generación de frases literarias. Con el fin de estudiar el efecto de los hiperparámetros, integramos nuestros experimentos al modelo propuesto en [11], donde se realiza un estudio semántico-contextual para la generación de texto literario.

Word2vec produce un vector numérico (embeddings) para cada palabra dentro de un vocabulario de aprendizaje. Estos vectores son calculados bajo un parametraje específico, de acuerdo a las características del corpus y los objetivos pretendidos. A partir de los embeddings es posible calcular la distancia entre estos, y determinar cuáles están más próximos entre sí, estableciendo un primer análisis de tipo semántico.

Este estudio exploratorio basado en Word2vec puede servir para determinar el parametraje que mejor se ajusta a un corpus literario, la intención es obtener una aproximación semántica adecuada para la generación de frases coherentes. Para ello, buscamos que los embeddings produzcan una semántica de tipo literaria. Por ejemplo,

[5] Término explicado ampliamente en [1], donde establecen algunas directrices para estudiar las habilidades cognitivas del hombre desde un enfoque computacional.

dada la palabra “azul” se esperaría que las palabras cercanas fueran “mar” o “cielo”, y no “verde” o “rojo”.

En la sección 2, presentamos el resumen de algunos trabajos con el mismo enfoque sobre implementaciones de modelos Word2vec. El corpus usado para el entrenamiento de nuestros modelos es descrito en la sección 3. Posteriormente, en la sección 4 detallamos los distintos parámetros usados para el entrenamiento, así como algunas gráficas con resultados.

En la sección 5 mostramos algunas frases generadas por los modelos que consideramos mejor entrenados, de acuerdo a nuestro criterio de dispersión. Los resultados de la evaluación manual efectuada sobre las frases generadas son presentados en la sección 6. Finalmente, en la sección 7 se presentan las conclusiones sobre los resultados.

2. Estado del arte

Los modelos de NN han sido abordados con mayor determinación en los últimos años por investigadores del área de Creatividad Computacional, sobretodo para la generación de texto literario. A continuación presentamos algunos trabajos relacionados.

2.1. Generación de texto no literario

Lebret et al. [6] proponen un algoritmo para generar frases biográficas, que es entrenado con un conjunto de datos biográficos extraídos de Wikipedia[6]. Otro modelo para la generación automática de texto se presenta en [16]. En este trabajo se propone la generación de descripciones para fragmentos de código Java. En el modelo se consideran elementos como: nombre de los constructores, de las llamadas, de las instancias, etc., para generar una descripción apropiada.

Otras implementaciones basadas en Word2vec para la generación automática de textos son propuestas en [4], donde Kharazmi et al. analizaron los embeddings propuestos por su modelo y los combinaron con una serie de n-gramas, con la intención de calcular la coherencia de un documento. Se tiene también el trabajo de Kiddon et al. [5], presentan un modelo de Word2vec para conservar la coherencia en el texto generado.

2.2. Generación de texto literario

Zhang et al. [20] proponen un modelo basado en Redes Neuronales Recurrentes para la generación de poesía en chino, el aprendizaje se basa en el reconocimiento de estructuras lingüísticas. Las estructuras analizadas se convierten en vectores que luego se usan para calcular la probabilidad de aparición de la siguiente estructura.

Otra propuesta basada en NN para poesía es presentada en [12], este modelo recibe una palabra como entrada y construye el texto con elementos relacionados a ésta.

Tabla 1. Características de **MegaLite** y **cGoethe** corpus (K: mil, M: millón).

	MegaLite	**cGoethe**
Frases	9.2 M	19 519
Tokens	99.7 M	340 K
Media por frase	11	17
Letras	604 M	2 M
Media por frase	65	103

Existen investigaciones, como el proyecto MEXICA [13], donde se ha logrado generar textos literarios con una longitud considerable, sobrepasando la barrera a nivel de frases.

MEXICA genera cuentos sobre la época pre-colombina de la mitología Azteca empleando el proceso denominado engagement-reflection propuesto en [15]. Otro algoritmo basado en NN para el análisis contextual fue propuesto en [2] para la generación de historias de ficción.

Finalmente, en [11] se propone un modelo para la generación de frases literarias, donde los embeddings son procesados bajo una interpretación geométrica para seleccionar las palabras que mejor obedecen a un contexto definido y formar frases coherentes y semánticas.

3. Corpus

En este trabajo empleamos dos corpus con textos literarios en Español (algunos traducidos de otros idiomas). El primer corpus es MegaLite [10], que contiene aproximadamente 5 000 documentos (en su mayoría libros), los cuales corresponden a los géneros: cuentos, poesía, teatro y ensayos. Por la cantidad considerable de textos contenidos, el corpus MegaLite fue utilizado para los distintos entrenamientos de las instancias basadas en la representación vectorial Word2vec.

El segundo corpus, cGoethe, está constituido por las obras principales de Johann Wolfgang von Goethe. La utilidad de este corpus es identificar y replicar las asociaciones semánticas que el autor emplea al momento de redactar sus obras. El corpus cGoethe fue usado para la fase de generación textual que se describe en la sección 5, dedicada a experimentos. Para construir ambos corpus, se convirtió cada documento al formato de codificación de caracteres utf8.

Además, se segmentaron los textos de cada documento en una frase por línea empleando la biblioteca NLTK[7] de Python. Posteriormente se seleccionó la novela "Los lamentos del joven Werther" [19] de Goethe, esto por la fuerte carga emocional y ciertos aspectos psicológicos observados en los personajes principales. De esta novela, se eligieron manualmente las frases que se consideraron como "literarias", es decir, se determinó que estas frases contenían un vocabulario estético, además de tener ciertas figuras literarias como la rima, la anáfora, la paráfrasis y otras.

Con estas frases se construyó el subconjunto cWerther utilizado en la fase de generación textual descrito en la sección 5. En la tabla 1 se muestra información

[6] https://www.wikipedia.org

[7] https://www.nltk.org/api/nltk.tokenize.html

Tabla 2. Características del subconjunto **cWerther**.

	cWerther	**Media por frase**
Frases	134	—
Tokens	1 635	12
Letras	9 321	69

Tabla 3. Valores de parámetros para los modelos Word2vec.

Parámetro	**Valores**
Iteraciones (i)	1, 5, 7, 10
Conteo mínimo (m)	3, 5
Tamaño del vector (s)	60, 100
Tamaño de la ventana (w)	5, 6, 7

estadística sobre los corpus MegaLite y cGoethe. El corpus cWerther es descrito en la tabla 2.

4. Representación vectorial de palabras

Uno de los intereses principales de este estudio es analizar el impacto de las representaciones vectoriales de palabras en la generación de frases literarias Por esto entrenamos 48 diferentes modelos de Word2vec utilizando el corpus MegaLite. La tabla 3 muestra los diferentes parámetros y valores utilizados para entrenar los modelos. "Iteraciones"(i) hace referencia al número de ciclos de entrenamiento completados con el corpus MegaLite.

"Conteo mínimo"(m) indica el número mínimo de veces que una palabra debe aparecer en el corpus para ser incluída en el vocabulario del modelo. "Tamaño del vector"(s) especifica la dimensión de los vectores de palabras. Por último, "Tamaño de la ventana"(w) representa la cantidad de palabras adyacentes (contexto) que son relacionadas a una palabra estudiada en una oración durante la fase de entrenamiento.

Entrenamos todos los modelos siguiendo el enfoque de skip-gram [8] con un muestreo negativo de cinco palabras y un umbral de submuestreo igual a 0.001. El tamaño del vocabulario para los modelos con $m = 3$ consta de 295 838 palabras, mientras que para los modelos con $m = 5$ el vocabulario es igual a 222 095 palabras.

Para analizar la relación entre los 48 diferentes modelos de Word2vec, realizamos una evaluación basada en los trabajos de Pierrejean y Tanguy [14] donde se comparan pares de modelos. Este método estima la tasa de variación global entre un par de modelos M_a y M_b a partir de un vocabulario Q.

En primera instancia, definimos $\mathrm{sim}^N_{M_a}(q)$ como el conjunto de las N palabras en M_a más similares (en términos de su similitud coseno) a una palabra objetivo q; de igual forma definimos $\mathrm{sim}^N_{M_b}(q)$. La tasa de variación de una palabra $q \in Q$ depende entonces del número de palabras en la intersección de los conjuntos $\mathrm{sim}^N_{M_a}(q)$

y $\mathrm{sim}^N_{M_b}(q)$, que está definida como:

$$var^N_{M_a,M_b}(q) = 1 - \frac{|\,\mathrm{sim}^N_{M_a}(q) \cap \mathrm{sim}^N_{M_b}(q)|}{N}\,, \tag{1}$$

donde $|\,\mathrm{sim}^N_{M_a}(q) \cap \mathrm{sim}^N_{M_b}(q)|$ corresponde al número de palabras (la cardinalidad) en la intersección. Finalmente, la tasa de variación global $\mathrm{Gvar}^N_{M_a,M_b}$, entre M_a y M_b sobre Q, con cardinalidad $|Q|$, es definida como:

$$\mathrm{Gvar}^N_{M_a,M_b} = \frac{\sum\limits_{q\in Q} \mathrm{var}^N_{M_a,M_b}(q)}{|Q|}\,. \tag{2}$$

Pierrejean y Tanguy [14] condujeron un estudio para determinar el mejor valor de N. Encontraron que para $N = 25$, la correlación media entre las tasas de variación, obtenida a partir de un conjunto de diferentes valores de N era maximizada. Basándose en este resultado, fijamos el valor de N a 25. Para crear el vocabulario Q, seleccionamos de **MegaLite** sólo aquellas palabras con un mínimo de 100 ocurrencias pertecientes a alguna de las siguientes categorías gramaticales: adjetivo, adverbio, sustantivo y verbo.

Después de este proceso de filtrado, el número total de palabras contenidas en Q es 42 681. Por cada una de las siguientes palabra objetivo, obtuvimos las 25 palabras más similares a partir de la similitud coseno de sus representaciones vectoriales: “amor”, “odio”, “tristeza”, “alegría”, “miedo”, “mujer”, “hombre” y “azul”. La figura 1 muestra los diagramas de caja resultantes de los 48 modelos ordenados por similitud coseno media. En esta figura se observa que aquellos modelos con una sola iteración y tamaño del vector más pequeño obtienen resultados más altos.

En contraste, los modelos con varias iteraciones y tamaño del vector más grande obtienen menores resultados. También es posible observar que, en general, los modelos con similitudes promedio más grandes presentan una menor dispersión al compararlos con aquellos modelos con similitudes promedio menores. Este comportamiento puede ser mejor observado en la figura 2, en donde se muestra, por cada modelo, la desviación estándar de las similitudes coseno agrupadas por palabra objetivo.

Al comparar ambas figuras, se puede ver que para los modelos con una sola iteración, los resultados de similitud son más altos y las desviaciones estándar más pequeñas. En constraste, los modelos con varias iteraciones y tamaño del vector más grande presentan resultados de similitud más bajos y desviaciones estándar más altas. De hecho, existe una fuerte correlación lineal negativa igual a -0.96 entre las similitudes coseno agrupadas por palabra objetivo y sus correspondientes desviaciones estándar. Con estos resultados, seleccionamos los mejores y peores modelos en términos de dispersión de similitud de coseno, sobre el conjunto de palabras objetivo.

5. Generación textual

En esta sección describimos brevemente el modelo presentado en [11], ilustrado en la figura 3, para la generación automática de frases literarias. El modelo consiste en dos etapas, siendo la segunda etapa donde integramos nuestra propuesta de modelo de

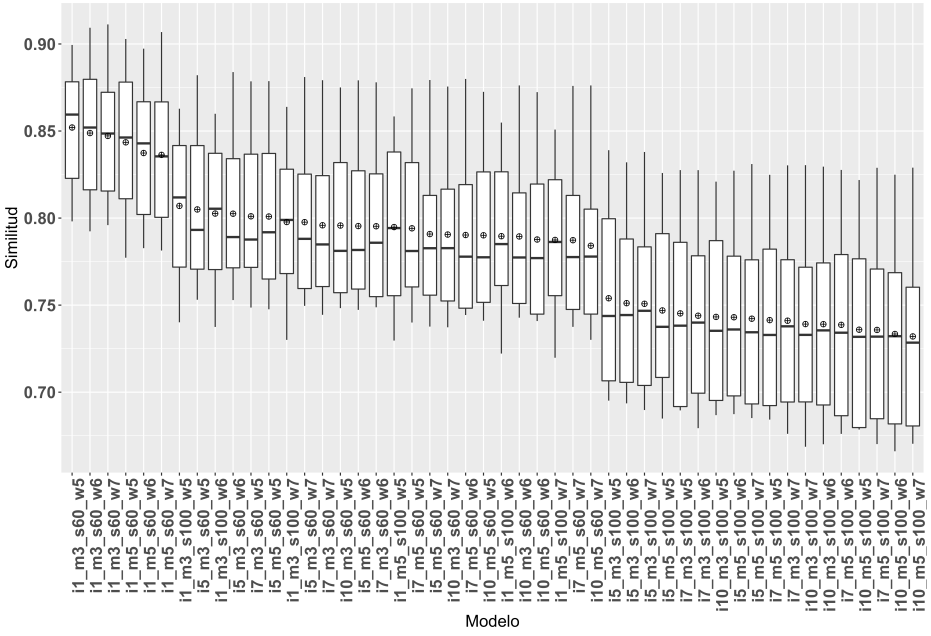


Fig. 1. Similitud coseno por cada modelo Word2vec.

representación vectorial Word2vec. El proceso inicia con una palabra (query) provista por el usuario. Esta palabra define el contexto para la generación de la nueva frase. La frase generada se construye reemplazando el vocablo de una frase original escrita por un humano, ésta es extraída del conjunto cWerther. Esta frase debe conservar el estilo y el contenido emocional que el autor expresa en la frase original.

Durante la primera etapa, el conjunto cWerther es procesado con FreeLing[8], un analizador morfosintáctico. Por cada frase original se detectan y reemplazan las palabras léxicas[9] por sus etiquetas gramaticales (POS) correspondientes. El resto de las palabras son conservadas. Como resultado una Estructura Gramatical semi-Vacía (EGV) es generada, esta EGV se compone entonces de etiquetas POS y palabras funcionales (artículos, conectores, adverbios, etc.).

Este método, basado en estructuras lingüísticas fijas, se conoce como Canned Text y ha sido empleado en trabajos donde se proponen modelos para la generación rápida de diálogos y frases cortas [18, 7]. En la segunda etapa, las etiquetas POS de la EGV se reemplazan con un vocabulario semánticamente ad-hoc.

Es decir, los sustantivos deben acercarse más al nuevo contexto, definido por el usuario, mientras que los verbos y adjetivos deben estar más próximos al contexto de la frase original. El vocabulario utilizado para el proceso de reemplazo se obtiene de los modelos Word2vec descritos en la sección 4. En esta etapa, el modelo generativo emplea tres elementos importantes:

– **PO**: La palabra que correspondía a la frase original.

[8] FreeLing se encuentra disponible en http://PLN.lsi.upc.edu/freeling

[9] Sustantivos, verbos principales y adjetivos.

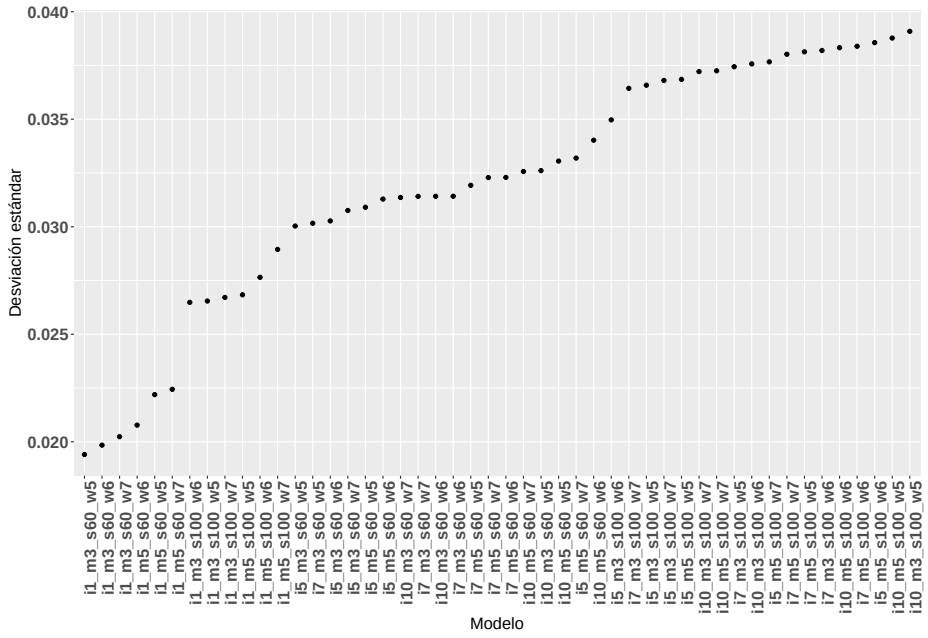


Fig. 2. Desviación estándar por cada modelo Word2vec.

– **VG**: El vocabulario extraído del corpus cGoethe.

– **Q**: El nuevo contexto.

A través de operaciones matemáticas y cálculos de distancias entre vectores, podemos utilizar los modelos Word2vec para elegir del vocabulario VG la palabra más próxima a PO, si se trata de un verbo o adjetivo; si se trata de un sustativo, se elige la palabra más cercana a Q. Más detalles del modelo generativo pueden consultarse en [11].

A continuación mostramos algunas frases generadas en la fase experimental de este proyecto. Las frases fueron generadas usando los cuatro modelos de Word2vec considerados como relevantes según nuestro criterio de dispersión (ver sección 4). Las instancias elegidas son: i1_m3_s60_w5 (i=1, m=3, s=60, w=5), i1_m3_s60_w6 (i=1, m=3, s=60, w=6), i10_m5_s100_w5 (i=10, m=5, s=100, w=5) y i10_m3_s100_w5 (i=10, m=3, s=100, w=5).

Por motivos prácticos, de aquí en adelante llamaremos a estas instancias: m3s60w5, m3s60w6, m5s100 w5 y m3s100w5, respectivamente. En los ejemplos a continuación se muestran las frases originales con las palabras reemplazables en negrita, seguidas por las nuevas frases guiadas por (Q).

– Instancia m3s60w5.
  Frase original: Pero en sus [cercanías] la [naturaleza] [brilla].
  – f(AMOR) = Pero en sus penas forma la gloria.
  – f(ODIO) = Pero en sus mejillas forma la alegría.

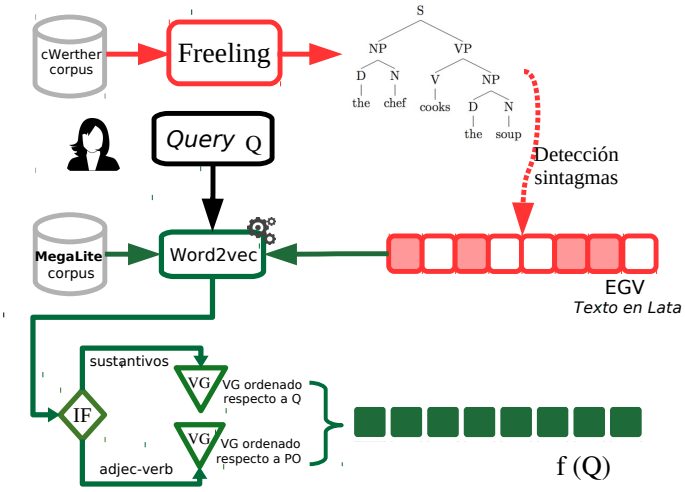


Fig. 3. Estructura general del modelo generativo de frases literarias.

  - f(MUJER) = Pero en sus hermanas desaparece la mujer.
- Instancia m3s60w6.
  Frase original: El [buen] o [mal] [humor] no [obedece] a nuestra [voluntad].
  - f(AMOR) = El seguro o loco amor no cuenta a nuestra alma.
  - f(ODIO) = El necesario o loco amor no considera a nuestra desesperación.
  - f(MUJER) = El único o buen padre no considera a nuestra amiga.
- Instancia m5s100w5.
  Frase original: El [llanto] se [agolpó] en mis [ojos].
  - f(AMOR) = El amor se agolpó en mis deseos.
  - f(ODIO) = El miedo se agolpó en mis sentimientos.
  - f(MUJER) = El niño se agolpó en mis padres.
- Instancia m3s100w5.
  Frase original: ¡Qué [gracia], qué [agilidad] en sus [movimientos]!
  - f(AMOR) = ¡Qué pasión, qué felicidad en sus deseos!
  - f(ODIO) = ¡Qué desesperación, qué angustia en sus sentimimentos!
  - f(MUJER) = ¡Qué mujer, qué criada en sus hijos!

Los ejemplos anteriores fueron generados a partir de frases originales tomadas aleatoriamente del corpus cWerther. Se observa que, en algunos casos, algunas palabras no son reemplazadas, esto ocurre cuando una palabra no es encontrada dentro del modelo Word2vec.

6. Evaluación

Se efectuó una evaluación de los modelos Word2vec entrenados en los experimentos de la sección 5. La evaluación fue manual y para ello se consideró un número asequible de frases que el evaluador debía analizar. Las frases se generaron utilizando

Tabla 4. Distribución de las frases evaluadas.

Descripción	Primer Conjunto	Segundo Conjunto	Tercer Conjunto	Total
Número de frases	16	16	8	40
Modelos Word2vec	**m3s60w5**	**m3s100w5**	**Frase original**	—
Contexto (queries)	AMOR, ODIO, MUJER		N/A	—

dos modelos Word2vec, el de mayor y menor valor de dispersión, m3s100w5 y m3s60w5 respectivamente.

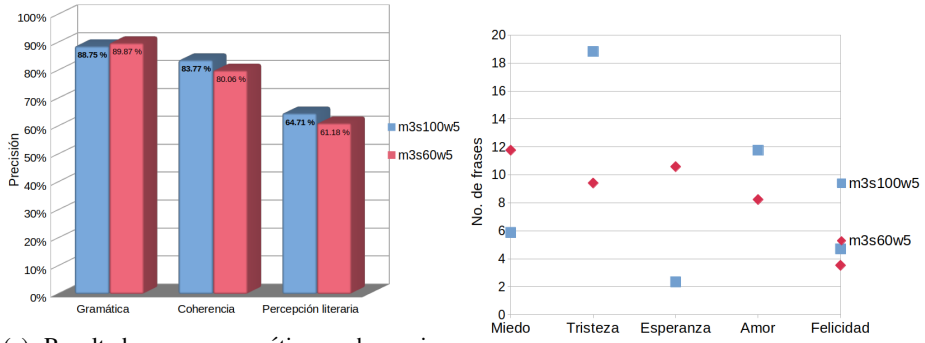
Seis personas fueron elegidas como evaluadores, estas son hablantes nativos del español, con grados universitarios y cuentan con cierta experiencia literaria, sea como lectores frecuentes o escritores amateurs. Se les pidió evaluar 40 frases (todas en español): 16 del modelo m3s60w5, 16 del modelo m3s100w5 y 8 frases humanas tomadas directamente de la obra traducida: Los lamentos del joven Werther. Los criterios a evaluar fueron:

- **Gramática**: Conjugaciones correctas y adecuada relación entre género y número.
- **Coherencia**: Las palabras en la frase deben situarse dentro del mismo contexto.
- **Percepción literaria**: Indicar si se consideraban las frases como literarias o no.
- **Percepción emocional**: Indicar la presencia de alguna de las siguientes emociones: Miedo, Tristeza, Esperanza, Amor y Felicidad.

Las frases de ambas instancias fueron mezcladas junto con las frases humanas sin ninguna preferencia antes de ser entregadas a los evaluadores. La distribución de las frases evaluadas se muestra en la tabla 4. Para los criterios: Gramática y Coherencia, se calculó un valor de precisión, definido como el número de palabras correctas entre el número total de palabras reemplazables (léxicas), esto para cada frase. Luego se tomó el valor promedio de precisión calculado entre todas las frases para obtener un valor representativo para cada criterio, los resultados se muentran en la figura 4a.

Se puede observar que el modelo m3s100w5 (en azul) obtuvo mejores resultados para Coherencia con un $83.77\,\%$, mientras que para Gramática, su desempeño fue menor en comparación con el modelo m3s60w5 (en rojo), sin embargo, el resultado es alentador con una precisión de $88.75\,\%$. Para el criterio de Percepción literaria, el modelo m3s100w5 también obtuvo el mejor resultado, con un $64.7\,\%$ de las frases percibidas como literarias. Los resultados obtenidos muestran una mejora a los presentados en [11] donde si bien, para Gramática, las evaluaciones fueron aceptables; para Coherencia, las frases no fueron bien percibidas.

Se puede apreciar que el modelo m3s60w5 arroja embeddings más cercanos a un query determinado, esto genera frases más coherentes y permite manipular el nuevo contexto de la frase según se desee. Siguiendo nuestra estrategia (ver sección 5) de aproximar los sustantivos al nuevo contexto, dejando los verbos y adjetivos próximos al contexto original, conseguimos generar una frase con un contexto nuevo, preservando la carga emocional original. Esto puede ser observado en la figura 4b, donde se aprecian los resultados del modelo m3s100w5, con frases fuertemente relacionadas a



(a) Resultados para gramática, coherencia y percepción literaria.

(b) Percepción emocional.

Fig. 4. Resultados de la evaluación manual en frases generadas.

las emociones “amor” y “tristeza”, siendo estas las principales emociones percibidas en la obra “Los lamentos del joven Werther”.

Para el modelo m3s60w5, la percepción parece no seguir un patrón regular, generando frases con una carga emocional aleatoria.

7. Conclusiones y trabajo futuro

En este trabajo hemos presentado resultados de experimentos con distintas implementaciones basadas en Word2vec para la generación de texto literario. Los resultados sugieren que la dispersión calculada entre los embeddings puede ser un indicador importante de la asociación semántica dentro de un espacio vectorial. El hiperparámetro discriminatorio que impacta en la dispersión de los modelos es el número de dimensiones[10] de la representación. De acuerdo a la literatura, a mayor tamaño del corpus, es necesaria una mayor dimensión para obtener una mejor exactitud en tareas de asociación de palabras [8]. Lo cual también es una intuición razonable. Nuestros resultados con los evaluadores humanos confirman esta intuición.

Las instancias seleccionadas para nuestros experimentos demostraron un comportamiento adecuado, generando frases más coherentes que las reportadas en [11]. La instancia con menor dispersión generó frases con una carga emocional y una coherencia aceptable. Esto significa que bajo un entrenamiento idóneo, un modelo Word2vec puede ayudar a la generación de frases con un contexto bien definido. En el futuro pretendemos extender los experimentos utilizando corpus periodísticos, técnicos o científicos. Por otra parte, pensamos también estudiar la inclusión de la rima combinada con los modelos Word2vec presentados para generar parejas de frases literarias.

Agradecimientos. Este trabajo fue parcialmente financiado por el Consejo Nacional de Ciencia y Tecnología (CONACYT, Mexico), beca número 661101 y por la Fédération de Recherche AGORANTIC – Avignon Université, (France).

[10] Este hiperparámetro es considerado al configurar el entrenamiento de un modelo Word2vec.

Referencias

1. Boden, M. A.: The creative mind: Myths and Mechanisms. Routledge (2004)
2. Clark, E., Ji, Y., Smith, N. A.: Neural text generation in stories using entity representations as context. North American Chapter of the Association for Computational Linguistics, vol. 1, pp. 2250–2260 (2018)
3. Colton, S., Wiggins, G. A.: Computational creativity: The final frontier? In: 20th European Conference on Artificial Intelligence. Association for Computational Linguistics, pp. 21–26 (2012)
4. Kharazmi, M. A., Kharazmi, M. Z.: Text coherence new method using word2vec sentence vectors and most likely n-grams. In: 3rd Iranian Conference on Intelligent Systems and Signal Processing, pp. 105–109 (2017) doi: 10.1109/ICSPIS.2017.8311598
5. Kiddon, C., Zettlemoyer, L., Choi, Y.: Globally coherent text generation with neural checklist models. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 329–339 (2016)
6. Lebret, R., Grangier, D., Auli, M.: Neural text generation from structured data with application to the biography domain. Empirical Methods in Natural Language Processing, pp. 1203–1213 (2016) doi: 10.48550/ARXIV.1603.07771
7. McRoy, S., Channarukul, S., Ali, S.: An augmented template-based approach to text realization. Natural Language Engineering, vol. 9, pp. 381–420 (2003)
8. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: 1st International Conference on Learning Representations, International Conference on Learning (2013) doi: 10.48550/ARXIV.1301.3781
9. Molins, P., Lapalme, G.: JSrealB: A bilingual text realizer for web programming. In: 15th European Workshop on Natural Language Generation, Association for Computational Linguistics, pp. 109–111 (2015)
10. Moreno, L. G., Torres, J. M.: Megalite: A new spanish literature corpus for nlp tasks. In: 8th International Conference on Artificial Intelligence and Applications (2021)
11. Moreno, L. G., Torres, J. M., Wedemann, R. S.: Literary natural language generation with psychological traits. In: Natural Language Processing and Information Systems, Springer International Publishing, pp. 193–204 (2020)
12. Oliveira, H. G., Cardoso, A.: Poetry generation with poetryme. Computational Creativity Research: Towards Creative Machines Atlantis Thinking Machines, vol. 7 (2015)
13. Pérez y Pérez, R.: Creatividad Computacional. Editorial Patria (2015)
14. Pierrejean, B., Tanguy, L.: Reproducibility of word embeddings: Identifying stable and unstable zones in the semantic space. Traitement Automatique des Langues Naturelles, vol. 1 (2018)
15. Sharples, M.: How We Write: Writing as creative design. Routledge (1996)
16. Sridhara, G., Hill, E., Muppaneni, D., Pollock, L., Vijay Shanker, K.: Towards automatically generating summary comments for java methods. In: IEEE International Conference on Automated Software Engineering, Association for Computing Machinery, pp. 43–52 (2010)
17. Torres, J. M.: Automatic Text Summarization. Wiley (2014)
18. van Deemter, K., Theune, M., Krahmer, E.: Real versus template-based natural language generation: A false opposition? Computational Linguistics, vol. 31, no. 1, pp. 15–24 (2005)
19. von Goethe, J. W.: The Sorrows of Young Werther. Penguin (1774)
20. Zhang, X., Lapata, M.: Chinese poetry generation with recurrent neural networks. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 670–680 (2014) doi: 10.3115/v1/D14-1074

ISSN 1870-4069

Descripción de los cambios de tonalidad en el color de neoplasias utilizando filtros de Gabor

Mariela Cruz Ángeles[1], Raúl Santiago Montero[1], Ernesto Bribiesca[2], Juan Francisco Mosiño[1], María del Rosario Baltazar Flores[1]

[1] Instituto Tecnológico de León,
Maestría en Ciencias de la Computación,
México

[2] Universidad Nacional Autónoma de México,
Instituto de Investigaciones en Matemáticas Aplicadas y Sistemas,
México

{m14200488, raul.santiago, juanfrancisco.mosino,
rosario.baltazar}@leon.tecnm.mx, bribiesca@iimas.unam.mx

Resumen. El cambio del color en las lesiones de la piel, también conocidas como neoplasias, es una de las características que permite una identificación visual entre las malignas y las benignas. Tanto la tonalidad de la neoplasia como su número de cambios han sido descritos, de manera cualitativa, a partir del análisis visual que realizan los especialistas. No obstante, se ha llegado a la conclusión de que entre más tonalidades se identifiquen en la lesión hay más sospecha de malignidad. En el presente trabajo se realiza una descripción cuantitativa que busca parametrizar los cambios de tonalidad que puede tener una neoplasia. Para ello se utiliza el filtro de Gabor bajo una configuración empírica, que utiliza la convolución del filtro como un dato, para estimar la dispersión de intensidad de los valores presentes en una imagen digital que contiene neoplasias benignas y malignas. Los resultados experimentales sugieren que al analizar la dispersión de los datos es posible obtener clasificaciones por arriba del 70 % en las bases de datos ISIC Archive y PH2. La clasificación se obtuvo mediante un clasificador KNN y pruebas de 5 pliegues, teniendo así un 71.74 % de precisión.

Palabras clave: Melanoma, clasificador, filtro de Gabor.

Description of Tonality Changes in the Color of Neoplasms Using Gabor Filters

Abstract. The color change in skin lesions, also known as neoplasms, is one of the characteristics that allow visual classification between malignant and benign neoplasms. The different skin tones found in the neoplasm have already been described qualitatively from the visual analysis carried out by specialists. However, it has been concluded that the finding of different tonalities in a lesion increases the probability of it being malignant. In the present work, a quantitative description is made seeking the parametrization of tonality

changes that a neoplasm may have. Thus, the Gabor filter is used under an empirical configuration, which uses the convolution of this filter as data to estimate the intensity dispersion from the values found within a digital image containing benign and malignant neoplasms. The experimental results suggest that by analyzing the dispersion of the data, it is possible to obtain the correct classification in over 70% of the cases in the ISIC Archive and PH2 databases. The classification was created by a KNN classifier and 5-fold testing, thus having a 71.74% accuracy.

Keywords: Melanoma, classifier, Gabor filter.

1. Introducción

El melanoma es un tumor con o sin la capacidad de producir pigmento en la piel, cuando se tiene sospecha de malignidad en una lesión cutánea, los especialistas deben medir el grado de poder discriminante de acuerdo a las características presentes en la anomalía. Melanoma, nevos comunes y nevos atípicos; son tipos de cáncer de piel que presentan características en común [14]. El melanoma es un tipo de cáncer de piel que se caracteriza por generar metástasis rápidamente cuando el tumor tiene contacto con la sangre.

Las células sanguíneas contaminadas se esparcen en el cuerpo provocando la aparición de cáncer en otros órganos [2, 4]. Si se detecta en etapas tempranas este tipo de neoplasia maligna, disminuye el riesgo de que afecte a otros órganos. Los especialistas en cáncer de piel utilizan métodos de detección de melanoma que se sustentan en analizar la presencia o ausencia de ciertos criterios dentro de la lesión[20]; dentro de éstos métodos destacan la regla de las 7 características y la regla ABCD, esta última se basa en los criterios de asimetría (A), borde (B), color (C) y estructura diferencial (D), que presente la neoplasia, ésta regla mejoró la precisión diagnóstica cuando se aplicó a diapositivas clínicas [11].

Estos procedimientos se soportan fundamentalmente en un análisis visual por parte del especialista [20]. Sin embargo y de acuerdo con los datos proporcionados por Heinze-Martin, en el año 2018 se reportaba una tasa de 0.31 de médicos especialistas en México por cada 100,000 habitantes con una especialidad en oncología médica [5] y no hay estadísticas de la proporción de especialistas en cáncer de piel. Este hecho afecta claramente una detección temprana de melanoma, el poder apoyarse de un dispositivo confiable y capaz de predecir si una lesión es benigna o maligna, ayudaría a los especialistas a filtrar el número de casos sospechosos.

La falta de especialistas en el área no es privativo de México, sino un fenómeno global. Como consecuencia se han desarrollado múltiples técnicas que ayuden a la detección de los diversos cánceres de piel, en sus orígenes se hacía uso de aceite sobre la piel y una lupa para poder examinar el área de pigmentación. Al paso de los años se ha implementado el uso de las herramientas tecnológicas para visualizar mejor las lesiones, de aquí nace el uso de la dermatoscopia; los dermatoscopios permiten el aumento de 10 veces de la piel [10] el uso de luz integrada y un lente que pueda aumentar la visualización hace que el especialista tenga más herramientas de mejora en su trabajo de diagnóstico.

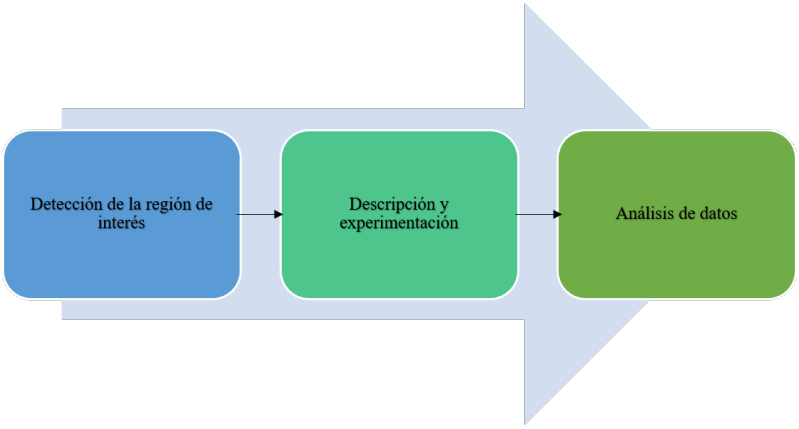


Fig. 1. Metodología propuesta.

Piccolo [12] informa que médicos con 5 años de experiencia en detección de cáncer, con el uso del dermatoscopio, aumentaron de un 71 % a un 90 % de sensibilidad y especificidad en el diagnóstico. Además, de acuerdo con estudios realizados por Vestergard [19], la dermatoscopia es más precisa que el examen ocular sin más herramientas para el diagnóstico de melanoma cutáneo cuando se realiza en el entorno clínico. La dermatoscopia permite visualizar con una imagen amplificada las características de la región de interés (ROI, por sus siglas en inglés), a partir de esa imagen se implementa un procesamiento digital para hacer un prediagnóstico de malignidad automático de melanoma.

La mayoría de esos procesos automáticos se ven en la necesidad de realizar una segmentación de la ROI, es decir, separar la zona de piel no pigmentada de la sí pigmentada, para hacer un análisis de características, con esto realizan mediciones automáticas que le permitirán al especialista dar un diagnóstico más cuantitativo y menos cualitativo de la lesión [3, 14, 20]. El realizar estos análisis de forma cuantitativa, automática y masiva con el objetivo de generar pronóstico de malignidad en la lesión cutánea ha impulsado el uso de técnicas de visión por computadora.

La parte central para realizar el pronóstico de malignidad es segmentar la ROI y aplicar la regla ABCD. Una vez segmentada la región de interés se ha podido describir, cuantitativamente tanto el borde como la asimetría de la ROI [17, 18]. Sin embargo, la cuantificación del color en la lesión no se ha trabajado tanto en comparación con los otros descriptores de la regla ABCD. En este trabajo se aplica un procedimiento fundamentado en el filtro de Gabor que describe el color en la lesión como una variación de la textura.

El filtro es aplicado a dos clases de imágenes: neoplasias benignas y neoplasias malignas. Se ha concluido de diversas investigaciones [7, 16, 8, 15] que entre más colores existan en la ROI a analizar, será mayor la sospecha de malignidad de la misma. El hecho de utilizar la textura dentro de un clasificador como una descripción del color, capacita a nuestra propuesta de generar un descriptor con el suficiente poder discriminante para proporcionar un pronóstico confiable.

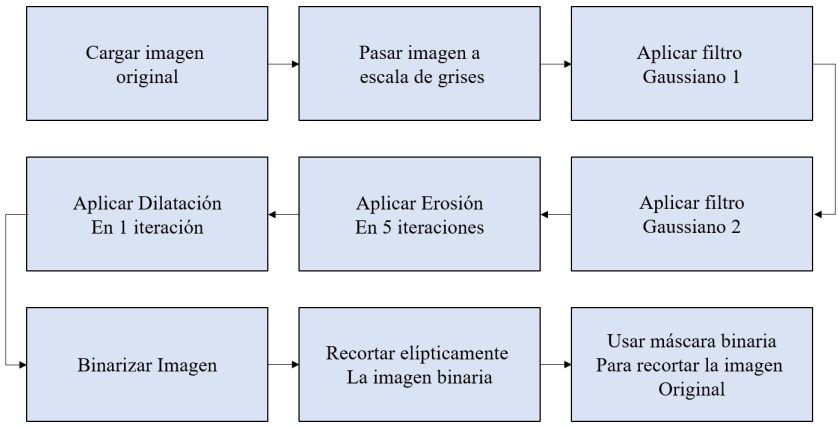


Fig. 2. Algoritmo de segmentación semi automática.

El punto central de la presente investigación, consiste en analizar la capacidad discriminante de la textura presente en cada tipo de imagen, para ello se toman los valores de la convolución entre el filtro de Gabor y la imagen. Posteriormente se obtienen sus parámetros de dispersión y tendencia central para poder establecer un umbral de clasificación.

Cuando el filtro de Gabor recorre la imagen, los valores de dispersión obtenidos muestran que a mayor dispersión, hay mayor probabilidad de malignidad. La propuesta fue aplicada a las bases de datos ISIC Archive y PH2 en un clasificador KNN, únicamente con el color como descriptor, se obtuvo un 71.74 % de clasificación entre neoplasias benignas y malignas.

Esto nos indica que al combinar este descriptor con los demás involucrados en la regla ABCD, se tendrá mayor poder discriminante y un mayor porcentaje de clasificación en una máquina de aprendizaje. En la tercera sección: Metodología, se muestra detalladamente cada una de las etapas del procedimiento propuesto, así como los algoritmos que fueron utilizados para los diferentes procedimientos automatizados. Además, en esta sección también se presentan las fases de experimentación y resultados, en conjunto con el análisis y la interpretación de los resultados.

2. Trabajos relacionados

La detección temprana de melanoma está siendo un problema, la derivación médica que se lleva a cabo desde que se tiene sospecha de una pigmentación hasta que se confirma el diagnóstico es muy tardada [1]. Por esa misma razón se ha buscado la implementación de procedimientos de detección automatizados, se han creado bases de datos de imágenes con neoplasias benignas y malignas para tener a disposición las instancias de prueba necesarias al momento de desarrollar un clasificador. La base de datos PH2 es una de las más completas en su área, consta de 200 imágenes dermatoscópicas junto con las anotaciones médicas correspondientes, está compuesta de 80 nevos comunes, 80 nevos atípicos y 40 melanomas malignos [9].

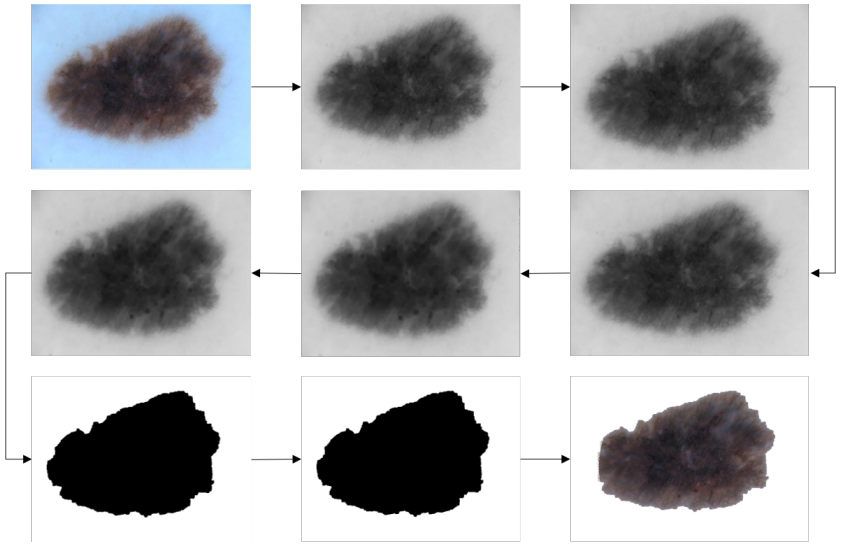


Fig. 3. Vista gráfica de segmentación semi automática.

En el presente trabajo se utilizan como instancias de prueba 200 imágenes de neoplasias malignas de la base de datos PH2 y 200 imágenes de neoplasias benignas de la base de datos ISIC Archive [6], ésta última pone a disposición tanto imágenes dermatoscópicas malignas como benignas. En 2015 se publicó el artículo [14] mostrando los resultados de una investigación de diferenciación entre melanoma y nevos displásticos (que son 2 tipos de neoplasias diferentes), la evaluación reveló el potencial de la textura para la diferenciación de melanoma y nevos displásticos.

Pérez [8] muestra en su trabajo que la variación de color y el histograma de color es una característica que se ha utilizado ampliamente en el pasado para la detección de las características del melanoma, además se pueden observar las técnicas que se estaban utilizando para la detección de melanoma, de las cuales destacan algunos estudios que utilizaron momentos de color e histogramas de color.

Una tendencia creciente en los últimos años es el desarrollo de características visuales de alto nivel (clínicamente significativas) como la asimetría del color, la variación del color, la clasificación del color, la detección del color y la cuantificación del color, esta última tiene como objetivo reducir el número de colores, se usa comúnmente como un paso de preprocesamiento para diversas tareas de procesamiento de gráficos e imágenes. La mayoría de los métodos de cuantificación están basados en algoritmos de agrupamiento de datos.

Los filtros de Gabor han sido empleados en multitud de aplicaciones de procesamiento de imágenes, entre ellas clasificación y segmentación de texturas, reconocimiento de imágenes y objetos. Aplicando filtros de Gabor en diferentes escalas y orientaciones, los patrones de textura pueden ser eficientemente descritos en el dominio frecuencial y localizados en el dominio espacial [13]. En este trabajo se configuró empíricamente un filtro de Gabor, experimentando el cambio de solo dos de sus variables: el ángulo y el tamaño de ventana.

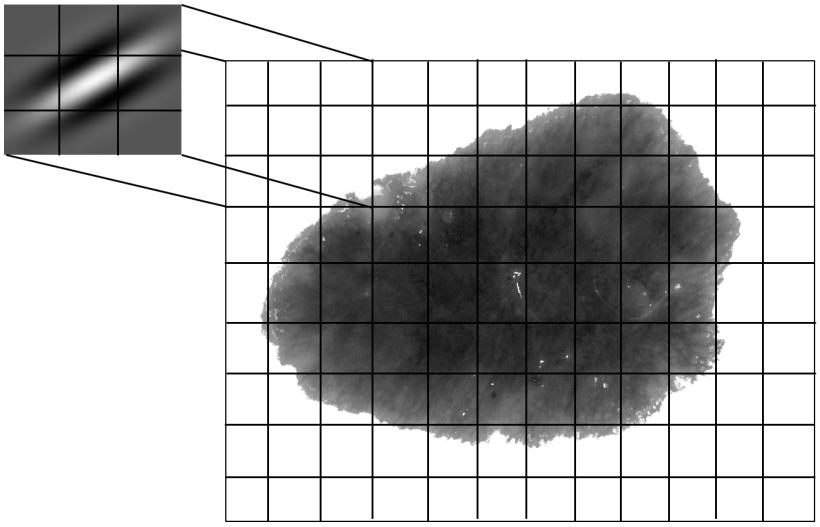


Fig. 4. Ejemplo de recorrido de filtro de Gabor en una imagen.

3. Metodología

Para realizar el presente trabajo se llevaron a cabo 3 etapas teniendo como primera: Detección de la región de interés, donde se describe cómo es que se llevó a cabo la detección de la ROI, así como los métodos realizados. En seguida se tiene la etapa: Descripción y experimentación, donde se muestran los métodos aplicados para la descripción del color y la experimentación involucrada. Por último, se tiene la etapa de: Análisis de datos, en ella se muestran los resultados obtenidos de la experimentación, así como la interpretación de los mismos. Todo este proceso se ejemplifica graficamente en la Figura 1.

3.1. Detección de la región de interés (ROI)

Para la segmentación de la ROI, se programó un algoritmo descrito en las Figuras 2 y 3: el procesamiento de la imagen consiste en lo siguiente: primeramente se carga la imagen original y se pasa a escala de grises, a esa misma imagen se le aplica un filtro gaussiano con tamaño de ventana 13×13 y desviación estándar de 1.5, una vez aplicado el primer filtro gaussiano se aplica uno nuevo cambiando únicamente el valor de la desviación estándar por 1.0.

Se toma la imagen filtrada y se aplica erosión en 5 iteraciones con tamaño de ventana de 3×3, a esa imagen erosionada se aplica dilatación en una iteración y se binariza para obtener una máscara que posteriormente se recorta elípticamente para eliminar ruido. Finalmente se utiliza la máscara binaria generada para hacer un recorte en la imagen original a partir de sus valores binarios, si el valor es 0 se conserva el valor original RGB del píxel y si el valor es 1 se cambia por 255, dejando así la ROI con valores RGB y el fondo de la imagen en color blanco.

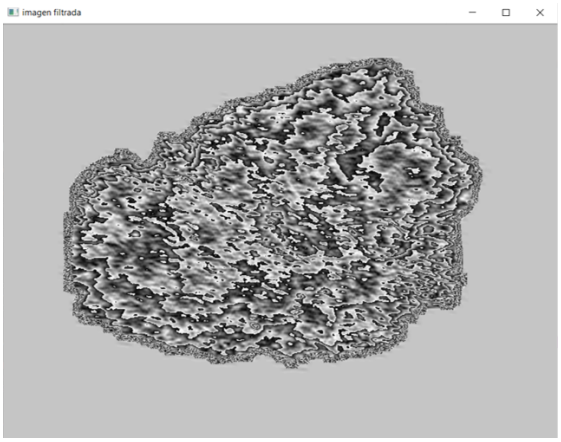


Fig. 5. Ejemplo de imagen filtrada.

Tabla 1. Resultados de parámetros estadísticos.

Tamaño máscara	Ángulo	Benignos		Malignos	
		Media	Desv. Std	Media	Desv. Std
3x3	**60**	**185.51**	**40.49**	**180.15**	**46.75**
	70	187.92	41.27	182.37	47.49
	80	183.93	40.00	178.67	46.25
	100	187.12	41.00	181.63	47.25
	140	184.71	40.24	179.41	46.50
6x6	60	224.23	57.44	216.40	62.70
	70	120.24	32.86	121.02	37.29
	80	86.43	37.90	89.93	42.21
	100	225.80	59.62	218.38	63.91
	140	135.31	32.93	134.80	37.28
9x9	60	68.93	43.80	73.97	48.30
	70	49.43	50.48	56.08	55.11
	80	222.42	63.10	217.04	63.94
	100	134.47	33.41	133.98	37.70
	140	181.30	43.04	176.78	47.65

3.2. Descripción y experimentación

La descripción de la textura se puede llevar a cabo de 3 formas: estadística, basada en modelos y basada en filtrado [13], en este trabajo se combinó la parte de filtrado y estadística. Una vez que se obtuvieron las imágenes segmentadas se debía aplicar el filtro de Gabor porque el descriptor principal para el clasificador es la textura, se propuso realizar un recorrido en la configuración del filtro cambiando sus parámetros de ángulo y tamaño de ventana. Básicamente el recorrido que se realiza es el que se muestra en la Figura número 4, la ventana del filtro de Gabor recorre la imagen y va modificando el valor de los pixeles. En la Figura 5 se muestra la imagen una vez que se ha filtrado.

Tabla 2. Valores ejemplo de desviación estándar.

Clase	Desviación estándar				
Malignos	75.23	55.7	49.21	72.25	71.75
Benignos	12.52	15.23	30.82	22.54	17.84

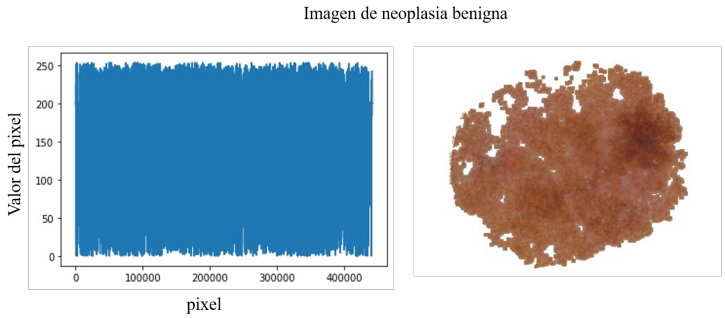


Fig. 6. Ejemplo de dispersión de valores por clase.

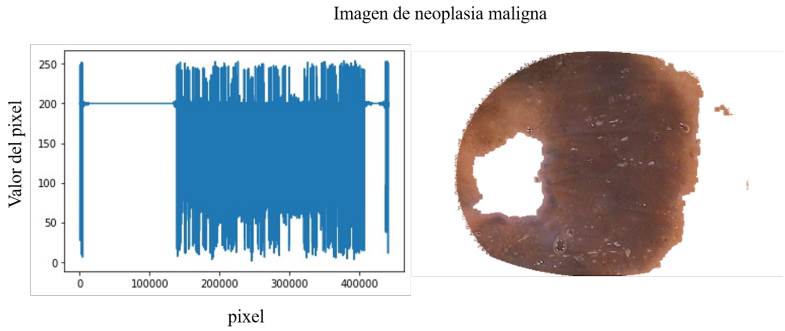


Fig. 7. Ejemplo de dispersión de valores por clase.

Se aplicó el filtro de Gabor a las dos clases de imágenes y se obtuvieron los valores de los pixeles de cada una de ellas para obtener su media y desviación estándar, por último se tomó un promedio de sus parámetros y los resultados se muestran en la Tabla 1: en las columnas 1 y 2 se muestran los tamaños de ventana y ángulos que se configuraron, en la tercera y cuarta columna se pueden ver los promedios de los estadísticos de la clase benignos, y en las últimas dos columnas se observan los estadísticos de la clase malignos.

3.3. Análisis de datos

Observando los resultados obtenidos en la Tabla 1, se puede concluir que la desviación estándar es el parámetro que marca una notable diferencia entre clases. Se calculó una diferencia entre los valores obtenidos en la clase benignos y en la clase malignos y se pudo ver que la configuración de tamaño de ventana 3×3 con

ángulo de 60°, fue la que marcó una diferencia mayor entre clases, así que se tomó esa configuración para utilizarla en un clasificador KNN, éste arrojó un 71.74 % de efectividad en la clasificación, únicamente tomando en cuenta los valores de dispersión para entrenarlo. Cabe mencionar que no se está utilizando otro descriptor más que el del color y que solo se realiza una predicción de malignidad o benignidad con base en el entrenamiento del clasificador.

En la Tabla 2, se presentan los valores de la desviación estándar de 5 imágenes de la clase benignos y 5 imágenes de la clase malignos para hacer notar la variación que hay en sus parámetros entre una y otra clase, además en la Figura 6 se muestra gráficamente la variación de los valores de los píxeles en una imagen de la clase benignos, se observa un comportamiento uniforme con menos variaciones en los valores de los pixeles, en comparación con la gráfica de la Figura 7 que corresponde a la clase malginos, ya que hay más variación entre los valores de los pixeles porque hay más variación en los colores presentes en la imagen de neoplasias malignas.

4. Conclusiones y trabajo futuro

De acuerdo con los resultados obtenidos se ha demostrado que usar la dispersión de los datos como descriptor entre clases, sí permite su clasificación teniendo una efectividad de 71.74 %. Es importante destacar que en este trabajo no se están utilizando bancos de filtro de Gabor, sino un solo filtro de Gabor con una configuración específica.

En el trabajo futuro se planea combinar éste descriptor con el uso de algún otro descriptor de Asimetría, Borde o estructura Diferencial, con la finalidad de evaluar su efectividad en la clasificación; además se tiene pensador programar un algoritmo evolutivo para encontrar la configuración adecuada de los parámetros del filtro de Gabor o el banco de filtros de Gabor que genere un porcentaje de clasificación mayor.

Referencias

1. Armengot, M., Martínez, V., Pitarch, G.: Prioridad de derivación de los casos de melanoma desde atención primaria. Piel, Formación continuada en dermatología, vol. 35, no. 4 (2019) doi: 10.1016/j.piel.2019.08.010
2. Celebi, M. E., Kingravi, H. A., Uddin, B., Iyatomi, H., Aslandogan, Y. A., Stoecker, W. V., Moss, R. H.: A methodological approach to the classification of dermoscopy images. Computerized Medical imaging and graphics, vol. 31, no. 6, pp. 362–373 (2007) doi: 10.1016/j.compmedimag.2007.01.003
3. Cruz, V. M.: Detección de melanomas a partir de imágenes dermatoscópicas (2018)
4. Goldstein, G.: Diagnosis and management of malignant melanoma. American family physician, vol. 63, no. 7, pp. 1359–1374 (2001)
5. Heinze, G., Olmedo, V. H., Bazán, G., Bernard, N. A., Guízar, D. P.: Medical specialists in Mexico. Gaceta Médica de México, vol. 154, no. 3, pp. 342–351 (2018)
6. Isic Archive (2019) Disponible en: https://isic-archive.com
7. Iyatomi, H.: Computer-based diagnosis of pigmented skin lesions. New developments in biomedical engineering, pp. 183–200 (2010)
8. Madooei, A., Drew, M. S.: Incorporating colour information for computer-aided diagnosis of melanoma from dermoscopy images: A retrospective survey and critical analysis. International Journal of Biomedical Imaging, pp. 1–18 (2016)

9. Mendonça, T., Marçal, A, Barat, C., Ferreira, P.: PH2: A public database for the analysis of dermoscopic images. Dermoscopy Image Analysis, pp. 419–439 (2015) doi: 10.1201/b19107-14
10. Moya Peñafiel, M. J.: Melanoma maligno cutáneo en una mujer indígena del municipio de Alto Baudó, Chocó, Colombia. Médicas UIS, vol. 27, no. 1 (2014)
11. Nachbar, F., Stolz, W., Merkle, T., Cognetta, A. B., Vogt, T., Landthaler, M., Bilek, P., Braun-Falco, O., Plewig, G.: The ABCD rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions. Journal of the American Academy of Dermatology, vol. 30, no. 4, pp. 551–559 (1994) doi: 10.1016/s0190-9622(94)70061-3
12. Piccolo, D., Ferrari, A., Peris, K., Daidone, R., Ruggeri, B., Chimenti, S.: Dermoscopic diagnosis by a trained clinician vs. a clinician with minimal dermoscopy training vs. computer-aided diagnosis of 341 pigmented skin lesions: A comparative study. British Journal of Dermatology, vol. 147, no. 3, pp. 481–486 (2002) doi: 10.1046/j.1365-2133.2002.04978.x
13. Pérez, J. A., Serrano, C., Acha, B.: Clasificación de lesiones de piel basada en filtros de Gabor y color. In: Congreso Anual de la Sociedad Española de Ingeniería Biomédica, pp. 125–128 (2009)
14. Rastgoo, M., Garcia, R., Morel, O., Marzani, F.: Automatic differentiation of melanoma from dysplastic nevi. Computerized Medical Imaging and Graphics: The official journal of the Computerized, vol. 43, pp. 44–52 (2015) doi: 10.1016/j.compmedimag.2015.02.011
15. Rodríguez, R., Montoya, G., Roldán, R., Carlos, B.: Principios básicos de dermatoscopia. Dermatología Revista Mexicana, vol. 58, no. 3, pp. 300–304 (2014)
16. Rubegni, P., Cevenini, G., Burroni, M., Perotti, R., Dell'Eva, G., Sbano, P., Miracco, C., Luzi, P., Tosi, P., Barbini, P., Andreassi, L.: Automated diagnosis of pigmented skin lesions. International Journal of Cancer, vol. 101, no. 6, pp. 576–580 (2002) doi: 10.1002/ijc.10620
17. Sancen, A., Santiago, R., Sossa, H., Perez, F. J., Martinez, J. J., Padilla, J. A.: Quantitative evaluation of binary digital region asymmetry with application to skin lesion detection. BMC Medical Informatics and Decision Making, vol. 18, no. 1, pp. 1–11 (2018) doi: 10.1186/s12911-018-0641-7
18. Santiago, R., Ortíz, A. M.: Discriminación de lesiones de piel mediante la descripción cuantitativa del borde y asimetría utilizando el concepto de compacidad. Innovación y Desarrollo Tecnológico Revista Digital, vol. 7, no. 2 (2015)
19. Vestergaard, M. E., Macaskill, P., Holt, P. E., Menzies, S. W.: Dermoscopy compared with naked eye examination for the diagnosis of primary melanoma: A meta-analysis of studies performed in a clinical setting. British Journal of Dermatology, vol. 159, no. 3, pp. 669–676 (2008) doi: 10.1111/j.1365-2133.2008.08713.x
20. Zaballos, D., Carrera, C., Puig, S., Malvehy, J.: Criterios dermatoscópicos para el diagnóstico del melanoma. Medicina Cutánea Ibero-Latino-Americana, vol. 32, no. 1, pp. 3–17 (2004)

ISSN 1870-4069

Estimación de valores de hemoglobina a partir del análisis de espectros FTIR de muestras de saliva de personas diabéticas

Miguel Sánchez Brito[1], Mónica M. Mata Miranda[2], Gustavo J. Vázquez Zapién[2]

[1] TecNM/Instituto Tecnológico de Aguascalientes, México

[2] Escuela Militar de Medicina, Centro Militar de Ciencias de la Salud, Secretaría de la Defensa Nacional, México

{miguel_sanchezbrito, gus1202}@hotmail.com, mmcmaribel@gmail.com

Resumen. La mala alimentación en adición con factores como el sedentarismo ha posicionado a la diabetes como una de las cuatro principales enfermedades no transmisibles a nivel mundial de acuerdo con la Organización Mundial de la Salud (OMS), siendo la diabetes tipo 2 la variante más frecuente de esta enfermedad [1]. La Asociación Americana de Diabetes (ADA) señala, esta enfermedad no es mortal, sin embargo, para evitar complicaciones que deriven en tal conclusión además de tener una buena calidad de vida es crucial un monitoreo continuo de los niveles de glucosa en sangre de los pacientes. De acuerdo con la ADA, una de las metodologías más confiables para monitorear los niveles de glucosa es la llamada prueba de hemoglobina glicosilada (A1C), la cual analiza el nivel de glicosilación de la hemoglobina (proteína exclusiva del torrente sanguíneo). Las molestias derivadas de un prolongado periodo de ayuno, así como la necesidad de repetir la prueba en más de una ocasión han permitido realizar investigaciones con la finalidad de proponer metodologías alternativas para asistir en la tarea de monitoreo de la glucosa en sangre [2]. En el presente trabajo evaluamos la posibilidad de emplear espectros obtenidos mediante espectroscopia infrarroja por transformada de Fourier (FTIR) de muestras de saliva para estimar los niveles de hemoglobina en pacientes previamente diagnosticados con diabetes tipo 2, para ello empleamos diversas técnicas comúnmente empleadas en el área de aprendizaje máquina.

Palabras clave: Aprendizaje máquina, espectroscopia FTIR, saliva, diabetes, hemoglobina.

Hemoglobin Values Estimation from the FTIR Spectra Analysis of Saliva Samples of Diabetic People

Abstract. Poor diet in addition to factors such as sedentary lifestyle has positioned diabetes as one of the four main non-communicable diseases worldwide, according to the World Health Organization (WHO), with type 2

diabetes being the most frequent variant of this disease [1]. The American Diabetes Association (ADA) points out, this disease is not fatal, however, to avoid complications that lead to such a conclusion, in addition to having a good quality of life, continuous monitoring of patients' blood glucose levels is crucial. According to the ADA, one of the most reliable methodologies for monitoring glucose levels is the called glycosylated hemoglobin test (A1C), which analyzes the level of glycosylation of hemoglobin (a protein unique to the bloodstream). The discomfort derived from a prolonged period of fasting, as well as the need to repeat the test on more than one occasion, have made it possible to carry out research with the aim of proposing alternative methodologies to assist in the task of monitoring blood glucose [2]. In the present work we evaluate the possibility of using Fourier transform infrared (FTIR) spectra of saliva samples to estimate hemoglobin levels in patients previously diagnosed with type 2 diabetes, for which we use various techniques commonly used in the area of machine learning.

Keywords: Machine learning, FTIR spectroscopy, saliva, diabetes, hemoglobin.

1. Introducción

La diabetes se ha colocado como una de las principales enfermedades no transmisibles a nivel mundial de acuerdo con lo señalado por la OMS afectando alrededor de 422 millones de personas en todo el mundo [1].

Una persona con diabetes tiene de dos a tres veces más probabilidades de sufrir afectaciones cardiacas, en mujeres diabéticas embarazadas un mal control de la diabetes puede provocar muerte fetal y otras complicaciones, además, la diabetes es causante de fallas renales, amputaciones de extremidades, perdidas de visión y afectaciones en las terminales nerviosas de los pacientes.

La diabetes es una enfermedad crónica, compleja, que requiere el cuidado médico continuo, con estrategias multifactoriales en la reducción del riesgo y control glicémico, de acuerdo con la Asociación Americana de Diabetes (ADA) las cuatro metodologías para llevar a cabo un correcto control glicémico son, la prueba A1C, *Fasting Plasma Glucose* (FPG), *Oral Glucose Tolerance Test* (OGTT), y *Random Plasma Glucose Test* (RPGT) [2]. A pesar de que ninguna de las cuatro metodologías mencionadas es considerada como *gold standard*, la prueba A1C es empleada con mayor frecuencia debido al tiempo de vida de la hemoglobina en el cuerpo humano, entre 2 y 3 meses aproximadamente, lo que permite conocer un historial de ese lapso de tiempo del comportamiento de los valores de glucosa de los pacientes a diferencia de las otras pruebas que reflejan el nivel de glucosa en el momento de la toma de la muestra. De acuerdo a lo publicado por [3], el tiempo estimado para la obtención de resultados de la prueba A1C va de entre 6 y 9 minutos y debido a la necesidad de mezclar la muestra con reactivos en conjunto con diversos pre procesamientos particulares el costo estimado de la misma es de entre 7 y 9 dólares además de requerir personal capacitado para realizar los procedimientos necesarios.

Si bien la hemoglobina es una proteína exclusiva del torrente sanguíneo, es posible encontrar otras proteínas que tienen la capacidad de vincularse con moléculas de glucosa (glicosilación) en distintos fluidos corporales como la saliva [4, 7].

La espectroscopia FTIR es una técnica que permite conocer la composición molecular de una muestra a partir de las vibraciones los enlaces de los elementos

químicos que la conforman; la muestra es sometida a radiación con frecuencias (Hertz) electromagnéticas pertenecientes a la región del espectro infrarrojo [8, 9], específicamente para el análisis de muestras biológicas se emplean frecuencias de la región media del infrarrojo (de 11.9 a 119.9 THz aproximadamente) [10, 11. 12].

La interacción de la muestra con las distintas frecuencias provoca las vibraciones de los enlaces que la conforman, cada enlace vibra con una frecuencia específica en función de los elementos que vincula y de su estructura (enlace sencillo, doble o triple); estas vibraciones se registran en un vector denominado *espectro FTIR*. El espectro FTIR de una muestra biológica ha sido ya segmentado en distintas regiones atribuidas a los principales grupos macromoleculares que la conforman: lípidos, proteínas y carbohidratos y ácidos nucleicos [13, 14].

A partir del análisis de los espectros FTIR mediante técnicas comúnmente empleadas en el área de aprendizaje máquina, se propone en el presente trabajo una metodología no invasiva para su asociación con valores de hemoglobina registrados por 258 pacientes. Los resultados obtenidos sugieren la viabilidad de emplear espectroscopia FTIR con muestras de saliva para estimar los niveles de hemoglobina de un paciente, al no requerir de reactivos ni pre procesamientos complejos la metodología propuesta podría ser una opción más ágil y accesible que la forma tradicional de realizar el estudio mediante muestras de sangre.

2. Métodos

En el periodo de 2019-2020 se recolectaron 258 muestras de saliva de pacientes previamente diagnosticados con diabetes tipo 2 en la Unidad de Especialidades Médicas (UEM) de la Secretaría de la Defensa Nacional (SEDENA). Los criterios de inclusión fueron pacientes de más de 18 años de edad que tuvieran un periodo de ayuno de cuando menos 8 horas.

Los criterios de exclusión contemplados fueron pacientes que se hubiera cepillado o enjuagado la cavidad oral con enjuague bucal antes de la toma de la muestra y pacientes que tuvieran algún tratamiento de ortodoncia u otro tratamiento oral.

Las muestras de saliva se procesaron con el espectrómetro Jasco FTIR-6600 empleando el accesorio *attenuated total reflectance* (ATR), con una resolución de 4 cm^{-1} y 120 escaneos como sugiere [8] y posteriormente normalizados mediante la técnica *standard normal variate* (SNV) (1) siguiendo lo señalado por [15]. Empleando la configuración mencionada se obtuvieron espectros constituidos por 3736 variables o números de onda que representan las frecuencias con las que fue analizada la muestra:

$$z = \frac{x-\mu}{\sigma}. \quad (1)$$

En (1), x es un valor real obtenido mediante espectroscopia FTIR, μ es la media del espectro que se analiza y σ la desviación estándar de la señal obtenida (espectro).

La complejidad del uso de espectros FTIR en el diagnóstico clínico radica en la naturaleza de las muestras orgánicas y en la cantidad de componentes de las mismas, al compartir gran parte de sus enlaces y al estar conformada por una gran cantidad de moléculas, los valores registrados en el espectro FTIR contemplan los mismos enlaces de todas las moléculas que constituyen la muestra que se analiza, tal es el caso de las proteínas, en la región 1700-1600 cm^{-1} (50.96472 a 47.96679 THz) del espectro FTIR

se registran vibraciones principalmente producidas por el doble enlace carbono oxígeno (C=O) atribuidos a las proteínas (específicamente a la amida I), sin embargo, si la muestra que se analiza está constituida por más de un tipo de proteínas, los enlaces C=O de todas ellas aportan a la medida registra en el espectro FTIR en esa región, además, si otro componente de la muestra también tiene ese enlace de igual manera se registrará en el espectro FTIR [13, 16].

Lo anterior ha dificultado la adopción de una metodología fiable para analizar espectros de muestras biológicas complejas con fines de apoyar en el diagnóstico clínico. En [15] se recopilan un conjunto de técnicas de clasificación/regresión que han permitido obtener buenos resultados en la caracterización de poblaciones a partir de espectros FTIR, las técnicas mencionadas son K-Vecinos más cercanos (K-NN), Análisis Discriminante Lineal (LDA), Modelos de regresión multi-variable (MLRM), Máquinas de Soporte Vectorial (SVM) y Redes Neuronales Artificiales (ANN). Debido a que el objetivo del presente trabajo fue analizar la posibilidad de estimar una variable numérica (valor de hemoglobina glicosilada) no categórica, se evaluó el desempeño de modelos de los modelos de regresión MLRM, SVM y ANN.

La estructura de la ANN con la que se experimentó corresponde a la estructura más simple como se sugiere en [17] con la finalidad de tener una perspectiva general de esta metodología y posteriormente modificarla incluyendo más capas o neuronas. La configuración consistió de una sola capa con una neurona que implementa una función de activación del tipo tangente hiperbólica.

La SVM con la que se experimentó implementa una kernel polinomial de grado 2 con un costo penalización (C) por violación de restricciones de 1 y un valore ξ de la función de pérdida de 0.1. El MLRM es similar al modelo de regresión simple con la salvedad de que considera dos o más variables independientes (todos los valores del espectro FTIR para el presente caso de estudio). Para evaluar los modelos se emplearon las técnicas de validación cruzada *Leave-One-Out* (LOOCV) y Hold out con una distribución 80-20 (para esta última), es decir, el 80% de las muestras se emplearon en el entrenamiento y 20% se usaron para el proceso de evaluación.

3. Resultados

El género, edad promedio y promedio de años de la evolución de la diabetes de los pacientes participantes se presentan en la Tabla 1. Los valores de hemoglobina registrados por los pacientes oscilan entre 5 y 11 mg/dl de acuerdo a los registros de la UEM. En la Fig. 1, se proporciona un histograma con la distribución de los valores registrados, en ella se puede apreciar que la mayoría de los pacientes presentaron valores de entre 5.5 y 6.0 mg/dl valor que de acuerdo con lo señalado por la ADA corresponde a personas diabéticas controladas.

Para la realización del estudio se agruparon los valores de hemoglobina en incrementos de 0.5 mg/dl, así, por ejemplo, el grupo 1 considera espectros de personas que registraron valores de entre 5 y 5.5 mg/dl. Los grupos quedaron conformados como se indica en la Tabla 2. En la Fig. 2 se presenta un espectro promedio de cada uno de los 12 grupos conformados. En la Fig. 3, se presentan los resultados obtenidos mediante un diagrama de cajas con la finalidad de identificar si las metodologías generan algún valor atípico y evaluar el comportamiento de las salidas de los modelos de regresión

con la finalidad seleccionar el que permita obtener la mayor distancia entre grupos con rangos lo mejor posible definidos para los valores de salida. Con fundamento en la Fig 3. es posible determinar que considerando el modelo de segmentación LOOCV de la base de datos para los procesos de entrenamiento-evaluación, los mejores resultados se obtienen mediante ANN, ya que las salidas del modelo presentan rangos definidos y únicos para los 12 grupos de valores de hemoglobina, adicionalmente podemos inferir que los resultados obtenidos presentan una fuerte correlación inversa ya que el grupo 1 que contempla espectros con los menores valores de hemoglobina registrados por los pacientes (5-5.5 mg/dl) producen los valores de salida más altos mientras que el grupo 12 que contempla espectros con los mayores valores de hemoglobina registrados por los pacientes (10.6-10.9 mg/dl) registran los valores de salida más bajos.

Puntualmente los rangos de los valores de salida de ANN se presentan en la Tabla El porcentaje correlación obtenido tras comparar los valores de salida de ANN con los valores reales de hemoglobina registrados por las pacientes fue de -99%, lo cual confirma la fuerte correlación inversa; el grado de relación permite inferir que si se incorpora un modelo de regresión lineal simple (LM) sería posible tener un valor aproximado al valor definido de los grupos (de 1 a 12). Los resultados de evaluar mediante un modelo de regresión las salidas de ANN y los valores de los conjuntos definidos para agrupar los distintos valores de hemoglobina (de 1 a 12) se presentan en la Fig. 4, en ella es posible apreciar que los valores estimados mediante ANN+LM se aproximan de buena manera a los valores reales asignados.

A través del modelo ANN+LM fue posible obtener un coeficiente de determinación (r^2) de 0.99 y un valor para la raíz del error cuadrático medio (RMSE) de 0.055 lo que sugiere que considerando la metodología LOOCV, es posible asociar espectros FTIR de muestras de saliva con valores de hemoglobina de pacientes diabéticos. Además de no requerir ningún reactivo para el preprocesamiento de la muestra, el tiempo de procesamiento de las 258 muestras mediante ANN+LM fue de aproximadamente 125 segundos lo que indica que la presente metodología podría ser empleada de buena manera para realizar este estudio de una manera más ágil y a menor costo que la metodología tradicional basada en sangre.

A pesar de que es posible encontrar diversos trabajos que emplean el modelo LOOCV para evaluar las técnicas de clasificación /regresión, en trabajos como los de [15] sugieren que los resultados obtenidos mediante LOOCV se deben al denominado *overffiting* de los métodos, señalando incluso que LOOCV resulta conveniente con una base de datos de hasta 20 muestras.

Por otra parte, trabajos como los de [18, 19] señalan que LOOCV resulta conveniente cuando se cuenta con una cantidad de muestras inferior a la cantidad de variables que componen a cada señal, tal es el caso del presente trabajo ya que se cuenta con un total de 258 espectro y cada espectro se componen de 3736 variables.

Con la finalidad de evaluar la metodología propuesta en el presente trabajo, se segmentó la base de datos considerando una distribución 80-20 (hold out) para realizar los procesos de entrenamiento y evaluación respectivamente, de esta forma el subconjunto de entrenamiento quedó conformado por 206 espectros y el subconjunto de evaluación por 52, los resultados se presentan en la Fig. 5, en ella se puede apreciar que a diferencia de la metodología LOOCV, con hold out el traslape entre clases aparece para todos los modelos de regresión. Los valores de r^2 y RMSE obtenidos mediante hold out se presentan en la Tabla 4.

Tabla 1. Información de los pacientes analizados.

Género		Edad promedio (años)	Tiempo de evolución de la enfermedad (años)
Masculino	119	60±11	11±8
Femenino	139		

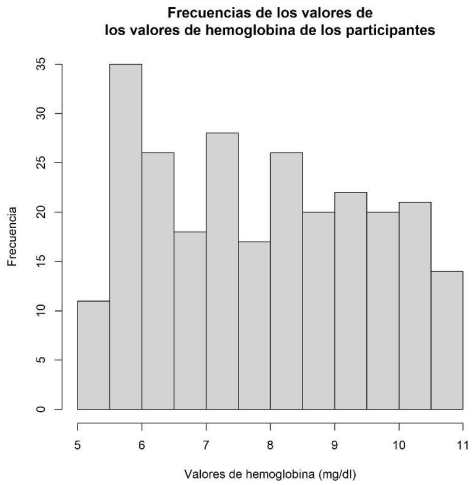


Fig. 1. Distribución de los valores de hemoglobina de los pacientes participantes.

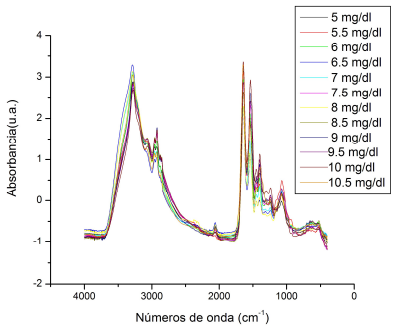


Fig. 2. Espectros promedio de cada una de las poblaciones conformadas a partir de las 258 muestras.

Con fundamento en la Fig. 5 y Tabla 4 puede señalarse SVM como la mejor opción considerando hold out. La necesidad del constante monitoreo del nivel de glucosa en sangre de personas diagnosticadas con diabetes ha permitido que distintos trabajos de investigación encaminen sus objetivos a proponer metodologías más accesibles y/o que ocasionen una menor incomodidad física del paciente.

En el presente trabajo se evaluó la viabilidad de emplear espectros FTIR de muestras de saliva con la finalidad de estimar los valores de hemoglobina de un paciente diabético. El uso de espectroscopia FTIR con la finalidad de estimar los valores de

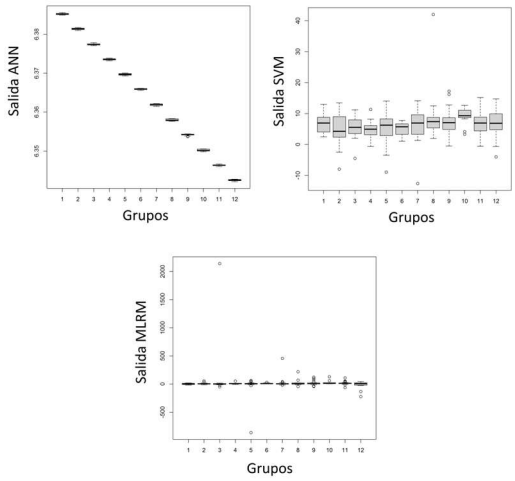


Fig. 1. Salidas obtenidas de los modelos evaluados considerando LOOCV.

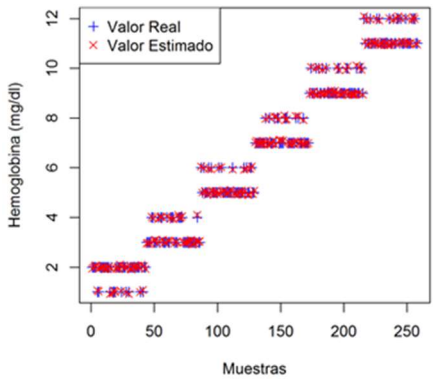


Fig. 4. Comparación de valores de reales de hemoglobina contra los valores estimados mediante un modelo ANN+LM.

hemoglobina de pacientes diabéticos ha sido abordado por trabajos como los de [20]–[23] con particulares diferencias con respecto al objetivo del presente trabajo.

En [20] se propone una metodología basada en regresión lineal que permite obtener una r^2 de 0.99 lo que sugiere que es una técnica muy exacta, sin embargo, los autores analizan una solución artificial no un fluido corporal; al tener control sobre los componentes con los que se realiza esta solución, las variaciones en los espectros FTIR estarán en función de la cantidad de glucosa añadida a cada muestra de la solución y el comportamiento de los fluidos del cuerpo humano presentan una variación más compleja dependiendo de las condiciones actuales del paciente, por lo que el modelo de regresión propuesto en [20] pudiera presentar complicaciones al momento de

Tabla 2. Agrupación de espectros en función de niveles de hemoglobina de los pacientes.

Grupo	Rango de glucosa (mg/dl)	Espectros del grupo	Grupo	Rango de glucosa (mg/dl)	Espectros del grupo
1	5-5.5	11	7	8-8.5	32
2	5.6-5.9	32	8	8.6-8.9	11
3	6-6.5	29	9	9-9.5	31
4	6.6-6.9	14	10	9.6-9.9	12
5	7-7.5	32	11	10-10.5	29
6	7.6-7.9	11	12	10.6-10.9	14

Tabla 1. Rangos de salida de ANN.

Grupo	Rango de Glucosa (Mg/Dl)	Espectros del Grupo	Rangos de Salida
1	5-5.5	11	6.384928–6.385555
2	5.6-5.9	32	6.381706-6.380961
3	6 - 6.5	29	6.377074–6.377816
4	6.6-6.9	14	6.373185-6.373867
5	7-7.5	32	6.369281-6.370033
6	7.6-7.9	11	6.365584-6.366089
7	8-8.5	32	6.361489-6.362205
8	8.6-8.9	11	6.357612-6.358347
9	9-9.5	31	6.353717-6.35441
10	9.6-9.9	12	6.349809-6.350548
11	10-10.5	29	6.346014-6.346679
12	10.6-10.9	14	6.342136-6.342791

evaluarlo con muestras reales. Los autores de [21] también emplean un modelo de regresión para estimar los valores de hemoglobina de pacientes, pero a través de espectros obtenido de la piel y labios de pacientes diabéticos. El modelo de [21] permitió obtener una r^2 de 0.62 en una población de 28 pacientes.

Uno de los principales inconvenientes señalado por los autores de [21] es la metodología que emplearon para estimar su r^2 ya que es una técnica propuesta por ellos, no la metodología tradicional, aunque reconocen que, para emplear su método en un ambiente real, su técnica de estimación de hemoglobina debe evaluarse apropiadamente siguiendo una metodología convencional, lo anterior impide contrastar los resultados obtenidos en el presente trabajo con [21].

En [22] se propone un modelo de regresión por mínimos cuadrados parciales para estimar los valores de hemoglobina de 232 muestras de sangre de distintos pacientes, si bien el modelo empleado permite obtener una r^2 de 0.96, los autores requirieron de hemolizar la muestra para facilitar el análisis de las mismas pese a que en la sangre es posible encontrar hemoglobina a diferencia del bio fluido (saliva) empleado en el presente proyecto.

Aunque la técnica propuesta en [22] podría agilizar la estimación de los valores de hemoglobina, la necesidad de usar reactivos para el pre procesamiento de la muestra y el malestar físico ocasionado a los pacientes por la extracción de sangre permanece constante. En [23], de manera similar a [22], se analizaron 232 muestras de sangre,

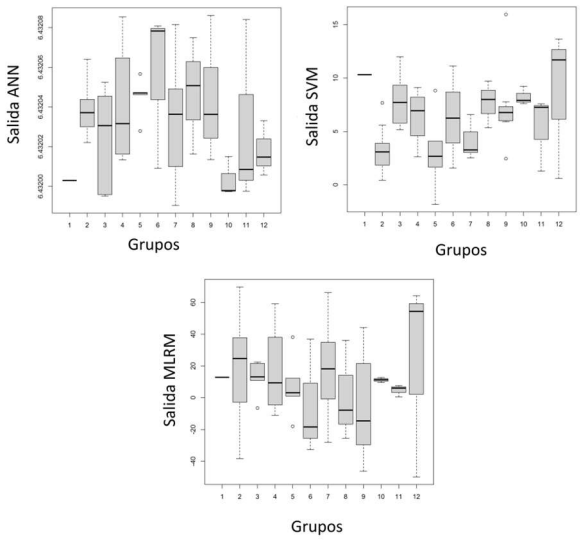


Fig. 2. Resultados obtenidos mediante los modelos regresión considerando la metodología hold out 80-20.

Tabla 2 Métricas obtenidas a partir de los modelos ANN, SVM y MLRM considerando la metodología hold out.

Método	r^2	RMSE
ANN	0.007	3.28
SVM	0.055	4.25
MLRM	0.013	29.47

pero empleando un modelo de regresión lineal. A través de su modelo reportan una r^2 de 0.94. Para el estudio de [23], las muestras también se hemolizaron lo que ocasiona los mismos inconvenientes que los señalados para [22].

A pesar de que los resultados obtenidos en el presente trabajo empleando hold out son menores a los resultados expuestos por [22, 23] la naturaleza del fluido que se analiza resulta considerablemente distinta, en nuestro caso, se analiza saliva, fluido en el que no hay presencia de hemoglobina, además, la muestra analizada no es sometida a ningún pre procesamiento con algún reactivo y la cantidad de muestras analizada en el presente estudio es superior.

Aunado a lo anterior, siguiendo la metodología de validación cruzada LOOCV fue posible proponer una técnica basada en redes neuronales y regresión lineal simple que permite obtener un valor de r^2 superior a los trabajos relacionados.

4. Conclusiones y trabajo a futuro

En el presente trabajo se evaluó la posibilidad de asociar espectros FTIR con valores de hemoglobina glicosilada en pacientes diabéticos. Se evaluaron los modelos de

regresión configurados a partir de ANN, SVM y MLRM mediante las metodologías de validación cruzada LOOCV y Hold out con una distribución 80-20.

Considerando la metodología LOOCV fue posible conseguir un valor de r^2 de 0.99 y un RMSE de 0.055 mg/dl acoplando un ANN y LM para la estimación de los valores de hemoglobina de un paciente a partir del análisis de espectros FTIR de muestras de saliva, estos valores sugieren que la metodología propuesta resulta mejor opción que las propuestas por trabajos similares, sin embargo, al emplear una segmentación hold out los mejores resultados fueron obtenidos mediante SVM permitiendo obtener valores de 0.055 y 4.25 para r^2 y RMSE respectivamente.

A partir de estos valores concluimos que la metodología propuesta resulta viable, pero es necesario recabar un mayor número de muestras para poder obtener valores similares al modelo ANN+LM, pero con una segmentación hold out de la base de datos.

En este mismo sentido, es necesario experimentar con técnicas de pre procesamiento y selección de atributos con la finalidad de reducir la cantidad de números de onda o frecuencias requeridas para segmentar las poblaciones no sólo para reducir el tiempo de procesamiento sino para poder identificar las frecuencias (números de onda) fundamentales y así construir un dispositivo portable que emita únicamente esas frecuencias y no todas las de la región media del espectro IR, este dispositivo evaluaría los valores obtenidos median ANN+LM (o el método más adecuado) y permitiría estimar los valores de hemoglobina del paciente usando saliva, esta medición resulta más relevante que los valores que se obtienen mediante los glucómetros tradiciones en donde las mediciones de glucosa reflejan únicamente el estado actual del paciente y no un historial de entre 2 y 3 meses como los valores de hemoglobina.

Referencias

1. Diabetes [Internet]. [cited 2021 Mar 22]. Available from: https://www.who.int/news-room/fact-sheets/detail/diabetes
2. Diagnosis | ADA [Internet]. [cited 2021 Mar 22]. Available from: https://www.diabetes.org/a1c/diagnosis
3. Whitley, H. P., Yong, E. V., Rasinen, C.: Selecting an A1C Point-of-Care Instrument. Diabetes Spectrum, vol 28, no. 3, pp. 201–208 (2015) doi: 10.2337/diaspect.28.3.201
4. Paluszkiewicz, C., Pięta, E., Woźniak, M., Piergies, N., Koniewska, A., Ścierski, W., Misiołek, M., Kwiatek, W. M.: Saliva as a first-line diagnostic tool: A spectral challenge for identification of cancer biomarkers. Journal of Molecular Liquids, vol. 307 (2020) doi: 10.1016/j.molliq.2020.112961
5. Saitou, M., Gaylord, E. A., Xu, E., May, A. J., Neznanova, L., Nathanm S., Grawe, A., Chang, J., Ryan, W., Ruhl, S., Knox, S. M., Gokcumen, O.: Functional specialization of human salivary glands and origins of proteins intrinsic to human saliva. Cell Reports, vol. 33, no. 7, pp. 108402 (2020) doi: 10.1016/j.celrep.2020.108402
6. Miller, C. S., Foley, J. D., Bailey, A. L., Campell, C. L., Humphries, R. L., Christodoulides, N., Floriano, P. N., Simmons, G., Bhagwandin, B., Jacobson, J. W., Redding, S. W., Ebersole, J. L., McDevitt, J. T.: Current developments in salivary diagnostics. Biomarkers in Medicine, vol. 4, no. 1, pp. 171–89 (2010)doi: 10.2217/bmm.09.68
7. Mirzaii-Dizgah, M. H., Mirzaii-Dizgah M. R., Mirzaii-Dizgah, I.: Serum and saliva total tau protein as a marker for relapsing-remitting multiple sclerosis. Medical Hypotheses, vol. 135, pp. 109476 (2020) doi: 10.1016/j.mehy.2019.109476
8. Smith, B. C.: Fundamentals of Fourier transform infrared spectroscopy. CRC Press (2011)
9. Smith, B. C.: Infrared spectral interpretation: A systematic approach. CRC Press (2018)

10. Mid-IR Spectroscopy. ScienceDirect, [Internet]. [cited 2021 Mar 22]. Available from: https://www.sciencedirect.com/topics/chemistry/mid-ir-spectroscopy
11. Shi, J., Wong, T. T. W., He, Y., Li, L., Zhang, R., Yung, C. S., Hwang, J., Maslov, K., Wang, L. V.: High-resolution, high-contrast mid-infrared imaging of fresh biological samples with ultraviolet-localized photoacoustic microscopy. Nature Photonics, vol. 13, no. 9, pp. 609–615 (2019) doi: 10.1038/s41566-019-0441-3
12. Türker-Kaya, S., Huck, C. W.: A review of mid-infrared and near-infrared imaging: Principles, concepts and applications in plant tissue analysis. Molecules, vol. 22, no. 1, pp. 168 (2017) doi: 10.3390/molecules22010168
13. Bel'skaya, L. V., Sarf, E. A., Makarova, N. A.: Use of fourier transform IR spectroscopy for the study of saliva composition. Journal of Applied Spectroscopy, vol. 85, no. 3, pp. 445–451 (2018) doi: 10.1007/s10812-018-0670-0
14. Takamura, A., Watanabe, K., Akutsu, T., Ozawa. T.: Soft and robust identification of body fluid using fourier transform infrared spectroscopy and chemometric strategies for forensic analysis. Scientific Reports, vol. 8 (2018) doi: 10.1038/s41598-018-26873-9
15. Morais, C. L. M., Lima, K. M. G., Singh, M., Martin, F. L.: Tutorial: Multivariate classification for vibrational spectroscopy in biological samples. Nature Protocols, vol. 15, no. 7, pp. 2143–2162 (2020) doi: 10.1038/s41596-020-0322-8
16. Naseer, K., Ali, S., Qazi, J.: ATR-FTIR spectroscopy as the future of diagnostics: A systematic review of the approach using bio-fluids. Applied Spectroscopy, vol. 56, no. 2, pp. 85–97 (2020) doi: 10.1080/05704928.2020.1738453
17. Lantz, B.: Machine Learning with R: Expert techniques for predictive modeling. Packt Publishing (2013)
18. Liò, P., Zuliani, P.: Automated reasoning for systems biology and medicine. Springer, vol. 30 (2019)
19. Andrade-Garda, J.: Basic chemometric techniques in atomic spectroscopy. Rroyal society of chemistry (2013)
20. Shan, X., Chen, L., Yuan, Y., Liu, C., Zhang, X., Sheng, Y., Xu, F.: Quantitative analysis of hemoglobin content in polymeric nanoparticles as blood substitutes using Fourier transform infrared spectroscopy. Journal of Materials Scienc, vol. 21, no. 1, pp. 241–249 (2010) doi: 10.1007/s10856-009-3864-4
21. Yoshida, S., Yoshida, M., Yamamoto, M., Takeda, J.: Optical screening of diabetes mellitus using non-invasive Fourier-transform infrared spectroscopy technique for human lip. Journal of Pharmaceutical and Biomedical Analysis, vol. 76, pp. 169–176 (2013) doi: 10.1016/j.jpba.2012.12.009
22. Pan, T., Li, M., Chen, J., Xue, H.: Quantification of glycated hemoglobin indicator HbA1c through near-infrared spectroscopy. Journal of Innovative Optical Health Sciences, vol. 7, no. 4, pp. 1350060 (2014) doi: 10.1142/S1793545813500600
23. Han, Y., Pan, T., Zhou, H., Yuanm, R.: ATR-FTIR spectroscopy with equidistant combination PLS method applied for rapid determination of glycated hemoglobin. Analytical Methods, vol. 10, no. 28, pp. 3455–3461 (2018)

ISSN 1870-4069

Propuesta de distribución aérea de vacunas COVID-19 en México para estados con alta tasa de incremento de casos

Miguel Ángel Walle-Vázquez[1 2], Santiago Omar Caballero Morales[1], Erick Leobardo Álvarez-Aros[1], Daniel Alejandro González-Bandala[3]

[1] Universidad Popular Autónoma del Estado de Puebla A.C
México

[2] Universidad Autónoma de Tamaulipas,
Facultad de Ingeniería y Ciencias,
México

[3] Universidad Autónoma de Tamaulipas,
Facultad de Comercio y Administración Victoria,
México

miguelangel.walle@upaep.edu.mx, {santiagoomar.caballero,
erickleobardo.alvarez}@upaep.mx, danielgoba84@gmail.com

Resumen. En el año 2020 inició una etapa de medidas gubernamentales para controlar la diseminación del COVID-19 en México. Esto con el objetivo de evitar el colapso de hospitales y demás servicios de atención médica, al mismo tiempo que mantener las actividades económicas en los diferentes sectores. Un avance importante al final de 2020 fue la creación de vacunas para este virus, lo que inició la etapa crítica de distribución de estas a todos los países y regiones del mundo. En los primeros meses de 2021 se inició la aplicación y distribución en la zona central de México. Sin embargo, esta distribución está sujeta a la recepción en un solo punto del país, el cual es el Aeropuerto Internacional de la Ciudad de México. Con el objetivo de aumentar la distribución de estas vacunas, se propone un esquema de entrega aérea en los estados con mayor tasa de crecimiento de casos por contagio. Esto para controlar aquellas regiones que potencialmente contribuyen con la mayor diseminación del virus, y para agilizar la entrega de vacunas a estados aledaños. Los resultados obtenidos, basados en un modelo logístico de ubicación y cobertura, dan soporte a la recomendación de hacer envíos principales a 6 estados para una cobertura eficiente del territorio nacional.

Palabras clave: COVID-19, distribución de vacunas, logística, ubicación y asignación de instalaciones.

Proposal for the Air Distribution of COVID-19 Vaccines in Mexico for States with a High Rate of Increase in Cases

Abstract. In 2020, a stage of government measures began to control the spread of COVID-19 in Mexico. This with the aim of avoiding the collapse of hospitals

and other health care services, while maintaining economic activities in the different sectors. An important advance at the end of 2020 was the creation of vaccines for this virus, which began the critical stage of their distribution to all countries and regions of the world. In the first months of 2021, the application and distribution began in the central area of Mexico. However, this distribution is subject to reception at a single point in the country, which is the Mexico City International Airport. In order to increase the distribution of these vaccines, an air delivery scheme is proposed in the states with the highest growth rate of cases due to contagion. This is to control those regions that potentially contribute to the greatest spread of the virus, and to expedite the delivery of vaccines to neighboring states. The results obtained, based on a logistics model of location and coverage, support the recommendation of making main shipments to 6 states for efficient coverage of the national territory.

Keywords: COVID-19, vaccine distribution, logistics, location and allocation of facilities.

1. Introducción

En diciembre de 2019 diversos casos de neumonía de etiología desconocida comenzaron a presentarse en Wuhan, la capital de la provincia de Hubei en China. El 7 de enero de 2020, el Centro de Prevención y Control de Enfermedades de China identificó esta afectación como síndrome respiratorio severo agudo por coronavirus 2 (SARS-CoV-2). Los casos en China fueron solo el comienzo de una pandemia mundial que la Organización Mundial de la Salud (OMS) nombró como Coronavirus 2019 (COVID-19) [1]. Pandemia que ha puesto a prueba los servicios de salud pública en el mundo y hace imprescindible que los países cuenten con sistemas de salud fuertes, con infraestructura y recursos necesarios para enfrentarla [2].

Para el 11 de marzo de 2020, la Organización Mundial de la Salud (OMS) declaró la pandemia de COVID-19, la primera no debida a influenza [3]. Desde su aparición, los gobiernos han estado interesados en encontrar la forma de mitigar la enfermedad por COVID-19, dado que, cuando una pandemia de esta magnitud ocurre, los servicios de salud se saturan, incrementando así la tasa de mortalidad [4]. Por su parte en México, el Consejo de Salubridad General, órgano que tiene el carácter de autoridad sanitaria del país, reconoce oficialmente el 23 de marzo de 2020, la epidemia de enfermedad por COVID-19 en México, como una enfermedad grave de atención prioritaria [5].

En julio de 2020, se reportó en más de 200 países, arriba de 10 millones de casos, solo la mitad era de la región de las Américas. En ese entonces, se esperaban contar con las vacunas necesarias para controlar la pandemia, y disminuir su impacto en la salud, la economía y la sociedad [3].

Sin embargo, debido a la alta tasa de contagio del COVID-19, la ausencia de tratamientos efectivos para reducir su tasa de mortalidad, y la dificultad para producir una vacuna efectiva y segura en el corto plazo representó un reto sin precedentes para todos los países. Aunque a finales del 2020 se tuvieron las primeras vacunas en fase III cuya efectividad se estimó arriba del 90%, el reto se presentó en la logística necesaria para su producción y distribución en gran escala para todos los países y sectores sociales prioritarios [3]. Los responsables de la salud pública en los diferentes países tienen el reto de atender planes de vacunación de dimensiones jamás conocidas; Se desarrollaron

las vacunas en tiempo récord, es tarea de ellos organizar la logística de vacunación también en tiempo ciertamente rápido [6].

Las vacunas contra el COVID-19 que se tienen registradas con estatus regulatorio en México son ocho, sin embargo, solo cinco han sido autorizadas hasta el 9 de febrero de 2021 para su aplicación y son: BNT162b2 de Pfizer, Inc. y BioNTech; AZD1222 de AstraZeneca y la Universidad de Oxford; Sputnik V del Instituto Gamaleya; Ad5-nCoV de CanSino Biologics Inc y CoronaVac de Sinovac Research and Development Co [7]. Finalmente, la pandemia de COVID-19 no solo ha exhibido la problemática en los sistemas de salud mundiales, sino a los retos que enfrentan países y farmacéuticas como la logística en el traslado de las dosis.

El presente estudio pretende controlar aquellas regiones que potencialmente contribuyen con la mayor diseminación del virus y agilizar la entrega de vacunas a estados aledaños. Por lo tanto, el principal objetivo de este trabajo es presentar una propuesta para la distribución de estas vacunas, mediante una descentralización del punto de llegada de estas a otros puntos en México para posteriormente hacer una entrega a zonas cercanas.

Esta propuesta considera que, aunado al Aeropuerto Internacional de la Ciudad de México, se identifiquen otros aeropuertos internacionales en México con características similares para la recepción de lotes de vacunas del exterior. Con esto se pretende hacer una entrega inmediata dentro de los estados donde se encuentran estos aeropuertos, y una entrega más ágil a los aeropuertos nacionales en los estados cercanos.

Con el objetivo de favorecer el control de la epidemia en México, se consideran como candidatos los aeropuertos internaciones en los estados en donde haya más números de contagios. En este documento se utilizó el modelo logístico de la p-Mediana para hacer la asignación más apropiada entre aeropuertos principales (candidatos) y los demás estados (vecinos). Considerando $i = 1, \ldots, I$ posibles centros de distribución, instalaciones o plantas (en este caso, los aeropuertos internacionales principales en México a donde llegará la vacuna desde el exterior), y $j = 1, \ldots, J$ destinos o clientes (en este caso, los aeropuertos nacionales a donde llegarán las vacunas desde los aeropuertos internacionales principales en México) [8].

El presente documento se estructura de la siguiente manera: En la sección 2 se describen los trabajos relacionados. En la sección 3 se presenta el resultado del análisis de datos. En la sección 4 se describe el modelo logístico de ubicación y asignación utilizado. En la sección 5 los resultados obtenidos. En la sección 6 las conclusiones y trabajos futuros y finaliza con el listado de referencias.

2. Trabajos relacionados

El tener cadenas de suministro eficientes ha sido vital para la correcta operación de los sectores productivos y de servicios en el Mundo. En la presencia de desastres naturales, esto también ha influido en la entrega oportuna de ayuda para la población en zonas afectadas.

La distribución de medicamentos a través de la cadena de suministro siempre será un proceso complicado. Las cadenas de suministro farmacéuticas y médicas han fallado durante la pandemia de COVID-19 a raíz de la implementación de cadenas de suministro globales ajustadas; por lo menos el 94% de las empresas de Fortune 1000

(las 1000 empresas estadounidenses más grandes clasificadas por su ingreso, de acuerdo con lo compilado por la revista de negocios estadounidense Fortune) han informado interrupciones en la cadena de suministro como resultado de la pandemia de COVID-19 [9].

Por lo tanto, bajo la situación actual, la adaptación y la planeación de las cadenas de suministro son necesarias para reducir el impacto económico y social de las restricciones causadas por las medidas de distanciamiento social establecidas alrededor del Mundo para reducir los contagios por COVID-19 [10].

En el caso particular del sector público en México, la distribución de la vacuna se realiza a través de las diferentes dependencias e instituciones del gobierno federal. En este contexto, se identifican como prioritarias las siguientes operaciones: planificación, adquisición, distribución, prescripción y dispensación. En cualquiera de estas etapas pueden existir fallas que puedan comprometer la operación eficiente de la cadena de suministro y, por consiguiente, la disponibilidad de medicamentos y vacunas para los diferentes sectores poblacionales.

En cuanto a la distribución de medicamentos, la cual se gestiona después de su adquisición, los esquemas más utilizados han sido la subcontratación y la distribución directa. La integración eficiente de estos agentes es importante para agilizar su operación bajo el presente escenario de emergencia sanitaria [9,10].

En la actualidad, la demanda global de escala sin precedentes de la vacuna COVID-19, imposible de satisfacer debido a la falta de preparación de los sistemas de producción y distribución. Disponibilidad que está limitada por factores externos a México. La Secretaría de Salud a través del Grupo Técnico Asesor de Vacunación Covid-19 (GTAVCovid-19) desarrolló recomendaciones sobre la estrategia de vacunación, tomando en cuenta la mejor evidencia disponible.

Consideró tomar en cuenta la distribución geográfica de la mortalidad; en la implementación de la campaña, se puede aprovechar la infraestructura del país, aumentando así el beneficio mediante la velocidad de cobertura y tener mayor efecto sobre la reducción de la mortalidad [13].

En México, el gobierno federal, a través de la Secretaría de Salud reportó al 17 de febrero de 2021, 2 millones 13 mil 563 casos confirmados con 177 mil 61 muertes por COVID-19 [14]. Con estos datos, para la primera etapa de vacunación se eligieron a la Ciudad de México y el Estado de Coahuila dado que se encuentran en el centro-norte de la República Mexicana y son los lugares en donde comenzó el fenómeno de rebrote de contagios.

En esta fase también se incluyeron a Querétaro, Nuevo León y el Estado de México. En cuanto a la llegada de las vacunas a México desde el exterior, fue hasta el 12 de enero de 2021 que se recibieron las primeras 546 mil 975 dosis, y se inició su aplicación en la Ciudad de México, Coahuila, Estado de México, Querétaro y Nuevo León [15].

El 20 de enero de 2021, en conferencia de prensa el secretario de la Defensa Nacional del Gobierno de México explicó que, a diferencia al plan anterior de distribución de vacunas, en esta ocasión se redujo el tiempo de entrega de las vacunas en las rutas del norte del país. La Defensa Nacional puso a disposición del Operativo Correcaminos 32 aeronaves; además, la Secretaría de Marina, un avión y dos helicópteros, y la Guardia Nacional, tres helicópteros [16]. Por lo anterior, se propone aumentar la distribución de las vacunas, a través de un esquema de entrega aérea en los estados con mayor tasa de crecimiento de casos por contagio. Con el fin de controlar aquellas regiones que

Tabla 1. Análisis de Pendientes de Crecimiento de Casos por Entidad Federativa en México.

Entidad Federativa	**Cuatrimestre 2020**			**Pendiente entre Cuatrimestres**			**Pendiente Total**
(Estado)	1Q	2Q	3Q	m_{12}	m_{23}	m_{13}	m_{total}
DISTRITO FEDERAL	0.10%	1.11%	2.70%	0.010	0.016	0.013	0.039
BAJA CALIFORNIA SUR	0.04%	0.94%	1.28%	0.009	0.003	0.006	0.019
QUERETARO	0.01%	0.29%	1.17%	0.003	0.009	0.006	0.017
DURANGO	0.00%	0.36%	1.02%	0.004	0.007	0.005	0.015
NUEVO LEON	0.01%	0.54%	0.96%	0.005	0.004	0.005	0.014
ZACATECAS	0.01%	0.32%	0.95%	0.003	0.006	0.005	0.014
COAHUILA	0.01%	0.69%	0.89%	0.007	0.002	0.004	0.013
GUANAJUATO	0.01%	0.54%	0.83%	0.005	0.003	0.004	0.012
SONORA	0.01%	1.01%	0.80%	0.010	-0.002	0.004	0.012
AGUASCALIENTES	0.02%	0.39%	0.80%	0.004	0.004	0.004	0.012
SAN LUIS POTOSI	0.01%	0.68%	0.77%	0.007	0.001	0.004	0.011
CHIHUAHUA	0.02%	0.26%	0.73%	0.002	0.005	0.004	0.011
TABASCO	0.05%	1.08%	0.62%	0.010	-0.005	0.003	0.009
COLIMA	0.00%	0.50%	0.52%	0.005	0.000	0.003	0.008
YUCATAN	0.03%	0.66%	0.51%	0.006	-0.001	0.002	0.007
HIDALGO	0.01%	0.34%	0.46%	0.003	0.001	0.002	0.007
BAJA CALIFORNIA	0.06%	0.43%	0.50%	0.004	0.001	0.002	0.007
TAMAULIPAS	0.02%	0.69%	0.43%	0.007	-0.003	0.002	0.006
MEXICO	0.03%	0.40%	0.44%	0.004	0.000	0.002	0.006
JALISCO	0.01%	0.25%	0.38%	0.002	0.001	0.002	0.006
MICHOACAN	0.01%	0.33%	0.37%	0.003	0.000	0.002	0.005
OAXACA	0.00%	0.36%	0.35%	0.004	0.000	0.002	0.005
TLAXCALA	0.02%	0.48%	0.35%	0.005	-0.001	0.002	0.005
PUEBLA	0.01%	0.42%	0.31%	0.004	-0.001	0.001	0.004
GUERRERO	0.01%	0.41%	0.31%	0.004	-0.001	0.001	0.004
SINALOA	0.04%	0.51%	0.31%	0.005	-0.002	0.001	0.004
QUINTANA ROO	0.06%	0.57%	0.31%	0.005	-0.003	0.001	0.004
MORELOS	0.03%	0.24%	0.26%	0.002	0.000	0.001	0.003
NAYARIT	0.01%	0.39%	0.22%	0.004	-0.002	0.001	0.003
VERACRUZ	0.01%	0.34%	0.17%	0.003	-0.002	0.001	0.002
CAMPECHE	0.01%	0.57%	0.15%	0.006	-0.004	0.001	0.002
CHIAPAS	0.00%	0.12%	0.03%	0.001	-0.001	0.000	0.000

potencialmente contribuyen con la mayor diseminación del virus, y agilizar la entrega de vacunas a los estados cercanos.

3. Análisis de datos

A partir de los datos proporcionados por el gobierno federal [17], se procedió a obtener el porcentaje de contagios para cada estado en los tres cuatrimestres del año 2020. Posteriormente, se estimó la tasa de crecimiento mediante el cálculo de la pendiente entre los tres cuatrimestres del año 2020. Finalmente, se obtuvo la pendiente total para determinar aquellos estados con las mayores tasas de crecimiento. Estos cálculos se presentan en la Tabla 1, destacando los primeros 12 estados con mayor tasa.

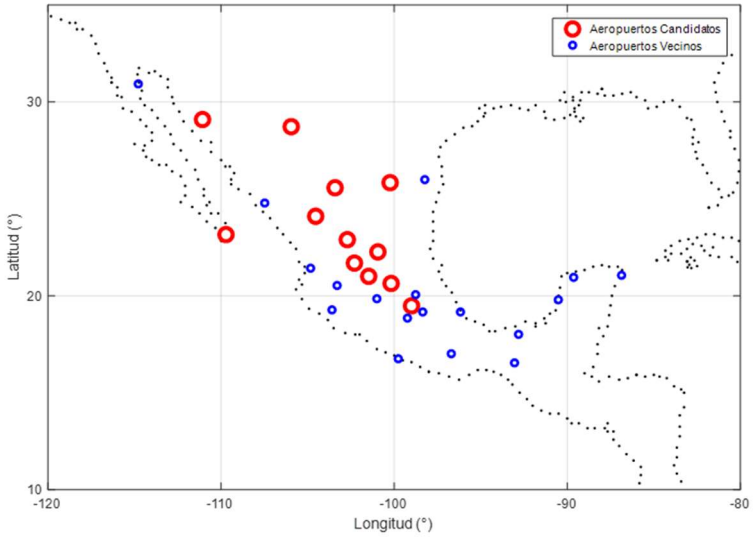


Fig. 1. Aeropuertos candidatos en estados con mayor tasa de crecimiento de casos COVID- 19.

Como se puede observar, la Ciudad de México es la entidad con mayor tasa de crecimiento de casos, en tanto, Chiapas es la entidad con la menor tasa. Aquí es importante señalar que una tasa alta no necesariamente se correlaciona con un número alto de personas contagiadas, sino con una alta velocidad de crecimiento de contagios, lo cual es importante para controlar la curva de contagio que representa a todas las personas que se enfermaron por COVID-19 (p.e., "aplanar la curva") lo que significa que las medidas de contención y mitigación han logrado reducir la saturación de los servicios de salud y el número total de casos de contagio [18]. También una alta tasa de crecimiento está correlacionada con una alta diseminación del virus a regiones aledañas [7].

Para propósitos de hacer más sencilla la planeación, se escogen de manera preliminar los 12 estados con mayor tasa como candidatos para recibir de manera prioritaria los cargamentos de vacunas desde el exterior para los mismos estados y para sus estados vecinos. En la Figura 1 se muestran las ubicaciones de los aeropuertos principales de estos estados (candidatos), incluyendo aquellos en los estados cercanos (vecinos).

4. Modelo logístico de ubicación y asignación

Para hacer la asignación más apropiada entre aeropuertos principales (candidatos) y los demás estados (vecinos), se hizo uso del modelo logístico de la p-Mediana. Considerando $i = 1, \ldots, I$ posibles centros de distribución, instalaciones o plantas (en este caso, los aeropuertos internacionales principales en México a donde llegará la vacuna desde el exterior), y $j = 1, \ldots, J$ destinos o clientes (en este caso, los aeropuertos nacionales a donde llegarán las vacunas desde los aeropuertos internacionales

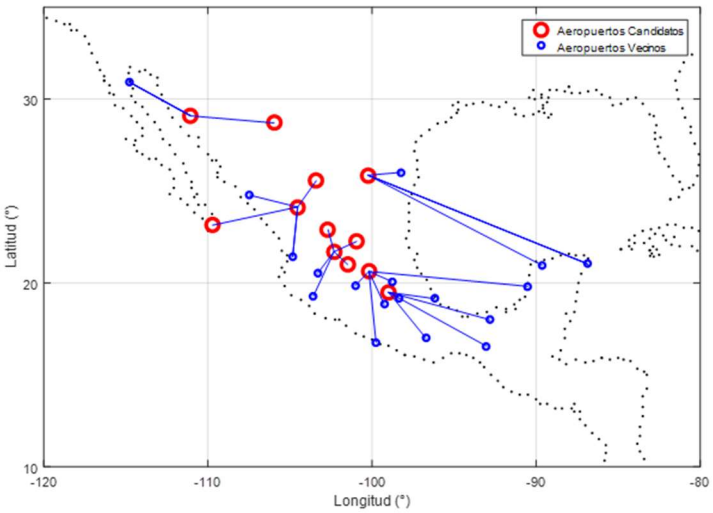


Fig. 2. Asignación de aeropuertos candidatos (internacionales) y vecinos (nacionales) en estados con mayor tasa de crecimiento de casos COVID-19.

principales en México), el modelo matemático de la p-mediana se define como [8, pp. 18–19]:

$$Minimizar\ \sum_{i\in I}\sum_{j\in J} d_j c_{ij} x_{ij}, \tag{1}$$

S.A.

$$\sum_{i\in I} x_{ij} = 1,\ \ \forall\, j \in J, \tag{2}$$

$$\sum_{i\in I} y_i = p, \tag{3}$$

$$x_{ij} - y_i \leq 0\ \ \forall\, i \in I;\ j \in J, \tag{4}$$

$$y_i \in \{0,1\}\ \ \forall i \in I, \tag{5}$$

$$x_{ij} \geq 0\ \ \forall i \in I;\ j \in J. \tag{6}$$

En donde (1) representa la función objetivo la cual minimiza el costo o distancia total ponderada de la asignación de aeropuertos candidatos i a los aeropuertos de los estados j (en este caso, c_{ij} representa el costo o distancia entre cada origen i y destino j en particular, y d_j la demanda del destino o estado j); (2) representa la restricción de que toda la demanda en el destino j debe satisfacerse; (3) representa la restricción que establece que exactamente se seleccionen p instalaciones (o aeropuertos candidatos finales) para hacer la distribución a los destinos j; (4) son restricciones que establecen que los destinos j solo se pueden asignar a instalaciones i que son abiertas, habilitadas o seleccionadas.

Finalmente, las restricciones (5) y (6) definen la naturaleza de las variables de decisión: x_{ij} es una variable binaria no-negativa, que es igual a "1" si se hace la

Tabla 2. Detalles de la asignación de aeropuertos candidatos (internacionales) y vecinos (nacionales) en estados con mayor tasa de crecimiento de casos COVID-19.

Entidad Federativa	(Estados) Vecinos																															
(Estados) Candidatos	DISTRITO FEDERAL	BAJA CALIFORNIA SUR	QUERETARO	DURANGO	NUEVO LEON	ZACATECAS	COAHUILA	GUANAJUATO	SONORA	AGUASCALIENTES	SAN LUIS POTOSI	CHIHUAHUA	TABASCO	COLIMA	YUCATAN	HIDALGO	BAJA CALIFORNIA	TAMAULIPAS	JALISCO	MICHOACAN	OAXACA	PUEBLA	GUERRERO	SINALOA	QUINTANA ROO	MORELOS	NAYARIT	VERACRUZ	CAMPECHE	CHIAPAS
DISTRITO FEDERAL	X												X								X	X						X		X
QUERETARO			X													X			X			X			X				X	
DURANGO		X		X			X																	X			X				
NUEVO LEON					X										X			X								X					
SONORA									X			X					X														
AGUASCALIENTES						X		X		X	X			X					X												

asignación entre el origen *i* y el destino *j*, y es igual a "0" en caso contrario. Para este problema en particular, c_{ij} se define como la longitud de arco geográfico entre los aeropuertos candidatos y los aeropuertos vecinos de todos los estados de la República Mexicana.

Esta longitud de arco se usa frecuentemente para determinar la distancia aérea recorrida por aviones. Considerando las coordenadas geográficas de longitud (λ) y latitud (ϕ) entre dos ubicaciones (p.e., *i* y *j*) en la Tierra esférica, la longitud de arco terrestre entre ambos se puede estimar como [19, p. 81]:

$$c_{ij} = R \times Arcos\big(sinf_i sinf_j + cosf_i cosf_j cos(l_i - l_j)\big), \quad (7)$$

en donde R es el radio de la Tierra y es igual a 6,371 km.

5. Resultados

Para propósitos de cálculo de c_{ij} y visualización de la solución fue utilizado el software GNU Octave 6.2.0 [20]. Para la resolución del problema de p-Mediana, fue utilizado el software de optimización Lingo 19.0 [21] considerando 6 aeropuertos principales a escoger de los 12 que fueron candidatos (véase Tabla 1). El resultado obtenido se presenta en la Figura 2.

La Tabla 2 presenta los detalles de la asignación mostrada en la Figura 2. Como se puede observar, aunado al Aeropuerto Internacional de la Ciudad de México (que coincide con el Estado de México y el Distrito Federal), se recomienda tener como aeropuertos de llegada de lotes de vacunas desde el exterior (Estados Unidos, China, Rusia, Inglaterra) a aquellos en los estados de Querétaro, Durango, Nuevo León, Sonora y Aguascalientes.

A partir de estos aeropuertos se puede hacer la entrega nacional a los demás estados de la República Mexicana. Por ejemplo, de Durango, se puede hacer de manera inmediata la entrega al mismo estado, y hacer el envío nacional a los aeropuertos de los estados de Baja California Sur, Coahuila, Sinaloa y Nayarit.

Nótese que para evitar la saturación de los 6 aeropuertos candidatos, a lo mucho, cinco estados vecinos son asignados a cada uno. Esta es la razón por la cual, algunos estados vecinos que pudieran estar más cerca de un aeropuerto candidato son asignados a otro aeropuerto, aunque esté un poco más alejado. Sin embargo, es importante recordar que la distancia está ponderada por el número de casos (véase Sección 4) por lo que un estado puede tener menor prioridad en la asignación si tiene pocos casos, aunque se encuentre más cerca.

6. Conclusiones y trabajo a futuro

Actualmente el aspecto de la entrega de vacunas a México y su posterior distribución en el territorio nacional son tareas de gran importancia para reactivar el sector económico y reducir la saturación del sector salud. En este trabajo se presenta una propuesta para agilizar la distribución mediante la llegada de vacunas del exterior a 6 aeropuertos internacionales en México a partir de los cuales se puede hacer la distribución a los demás estados. En particular, se hace énfasis en la recepción directa en los estados con mayor tasa de contagios.

La recepción y distribución a través de los aeropuertos internacionales y nacionales se propone para hacer más rápida y segura su llegada a los estados. Esta red aérea se puede conectar de manera eficiente con la logística de "última milla", definida como el último segmento de un proceso de entrega, que a menudo se considera el aspecto más caro y menos eficiente de una cadena de suministro [22, p. 309] que, involucra a los demás agentes implicados en su distribución como son la Guardia Nacional y las autoridades estatales.

Como trabajo futuro dentro de la misma línea de investigación se debe considerar el poder incorporar el análisis de asignación de aeropuertos vecinos, basado en diversas variables vinculadas con la tasa de contagio estatal, que permitan delimitar los criterios esenciales de distribución de recursos y suministros.

Además de comparar los resultados del modelo propuesto con el plan de distribución de vacunas en México publicado por la Defensa Nacional. Asimismo, realizar un mapeo para establecer centros de distribución en zonas estratégicas para el control de la vacuna de COVID-19 y el resto de los medicamentos del sector salud.

Referencias

1. Acosta, L. D.: Response capacity to the COVID-19 pandemic in Latin America and the Caribbean. Rev. Panam. Salud Publica/Pan Am. J. Public Heal., vol. 44, no. 1 (2020) doi: 10.26633/RPSP.2020.109
2. Grupo Técnico Asesor de Vacunación Covid: Priorización inicial y consecutiva para la vacunación contra SARS-CoV-2 en la población mexicana. Recomendaciones preliminares. Salud Publica de Mexico, vol. 63, no. 2, pp. 286–307 (2021)

3. Orgaización Pan Americana de la Salud: Orientaciones para la planificación de la introducción de la vacuna contra la COVID-19. (2020). Consultado: feb. 17, 2021. [En línea]. Disponible en: https://iris.paho. org/handle/10665.2/52533
4. Ortiz-Hernández, L., Pérez-Sastré, M. A.: Social inequalities in the progression of COVID-19 in the Mexican population. Rev. Panam. Salud Publica/Pan Am. J. Public Heal., vol. 44 (2020) doi: 10.26633/RPSP.2020.106
5. Gobierno de México: Acuerdo por el que el Consejo de Salubridad General reconoce la epidemia de enfermedad por el virus SARS-CoV2 (COVID-19) en México, como una enfermedad grave de atención prioritaria, así como se establecen las actividades de preparación y respuesta ante dicha pandemia. DOF (2020)
6. Salaices-Sánchez, M., Menendez-Ramos, J. C.: Y a final de año llegó la esperanza. At the nd of the year hope came, Anales de la Real Academia Nacional de Farmacia, vol. 86, pp. 225–230 (2020)
7. Gobierno de México: Vacunación COVID-19 –Coronavirus. Gobierno de México - Información Covid-19 México, 2021. https://coronavirus.gob.mx/vacunacion-covid/ (2021)
8. Diaz-Ramirez, N.: Modelo de centros de gravedad y asignación por clusters para puntos de distribución de líquidos de consumo humano en Bogota. Universidad Libre, Facultad De Ingeniería Industria (2019)
9. Golan, M. S., Jernegan, L. Linkov, H. I.: Trends and applications of resilience analytics in supply chain modeling: Systematic literature review in the context of the COVID-19 pandemic. Environment Systems and Decisions, vol. 40, no. 2. Springer, pp. 222–243 (2020) doi: 10.1007/s10669-020-09777-w
10. Queiroz, M. M., Ivanov, D., Dolgui, A., Fosso-Wamba, S.: Impacts of epidemic outbreaks on supply chains: Mapping a research agenda amid the COVID-19 pandemic through a structured literature review. Annals of Operations Research (2020) doi: 10.1007/s10479-020-03685-7
11. Lozano-Diez, J., Marmolejo-Saucedo, A. J. A., Rodriguez-Aguilar, R.: Designing a resilient supply chain: An approach to reduce drug shortages in epidemic outbreaks. vol. 6, no. 21 (2020) doi: 10.4108/eai.13-7-2018.164260
12. OMS, Panel de la OMS sobre la enfermedad por coronavirus (COVID-19)", Cuadro de mando de la OMS sobre la enfermedad por coronavirus (COVID-19), (2021) https://covid19.who.int/ (consultado feb. 18, 2021)
13. Barrientos-Gutiérrez, T., Alpuche-Aranda, C., Bautista-Arredondo, S.: Preguntas y respuestas sobre la estrategia de vacunación contra Covid-19 en México. Salud Pública de México (2021) https://www.medigraphic.com/cgi-bin/new/resumen.cgi?ID ARTICULO =98735
14. GobMx, COVID-19 Tablero México - CONACYT - CentroGeo - GeoInt - DataLab. Gobierno de México -Información Covid-19 México https://datos.covid-19.conacyt.mx/#DOView (consultado feb. 18, 2021) (2021)
15. Gobierno de México: Política nacional rectora de vacunación contra el SARS-CoV-2 para la prevención de la COVID-19 en México (2021)
16. Gobierno de México: Control de la epidemia y adquisición de vacunas son prioridades para el gobierno federal, afirma presidente – Presidente de México. Presidencia de la Republica - Gobierno de México https://presidente.gob.mx/control-de-la-epidemia-y-adquisicion-de-vacunas-son-prioridades-para-el-gobierno-federal-afirma-presidente/ (2021)
17. Martinez-Soria J. W., Vargas-Flores, A.: Covid-19: evolución y estimaciones de las curvas epidémicas. Consultado: mar. 23, 2021. [En línea]. Disponible en: http://bibliodigitalibd.senado.gob.mx/bitstream/handle/123456789/4877/Reporte TE 78 Curvas epidemicas F.pdf?sequence=1&isAllowed=y (2020)
18. Sánchez-Sierra, S. T., Caballero-Morales, S. O., Sánchez-Partida, D., Martínez-Flores, J. L.: Facility location model with inventory trasnportation and management costs. Actalogistica.EU, vol. 5, no. 3, pp. 79–86 (2018) doi: 10.22306/al.v5i3.98

19. Eaton, J. W.: GNU Octave is a high-level language, primarily intended for numerical computations. GNU Octave, https://www.gnu.org/software/octave/about (consultado mar. 15, 2021) (2021)
20. LINDO Systems Inc.: LINGO and optimization modeling https://www.lindo.com/index.php/products/lingo-and-optimization-modeling (consultado mar. 23, 2021) (2021)
21. Lim, S. F. J. W. T., Jin, X., Srai, J. S.: Consumer-driven e-commerce: A literature review, design framework, and research agenda on last-mile logistics models. International Journal of Physical Distribution and Logistics Management, Emerald Group Publishing Ltd, vol. 48, no. 3, pp. 308–332 (2018) doi: 10.1108/IJPDLM-02-2017-0081

ISSN 1870-4069

Un sistema experto para corrección de textos usando reglas ortográficas

Manuel Cristóbal López Michelone

Universidad Nacional Autónoma de México,
México

morsa@la-morsa.com

Resumen. Hoy en día los procesadores de palabras nos son familiares y cotidianos. Además de permitir a los usuarios crear documentos, estos pueden ser revisados y corregidos por software incluido que normalmente permite revisar si las palabras están correctamente escritas. El enfoque más usado es depender de un diccionario con cientos de miles de palabras para hacer la corrección ortográfica. Aquí presentamos un sistema experto que saca ventaja de las características del idioma español para hacer la corrección ortográfica sin necesidad de usar un diccionario. Además, el enfoque puede servir para aprender y entender el uso de muchas reglas comunes del idioma español.

Palabras clave: Corrección ortográfica, sistemas expertos, reglas ortográficas, algoritmos.

An Expert System for Proofreading Using Spelling Rules

Abstract. Word processors are very familiar to us. In addition to allowing users to create documents, a word processor can review and correct the documents trying to find spelling mistakes. The most widely used approach is to rely on a dictionary with hundreds of thousands of words to do the spell checking. Here we present an expert system that takes advantage of the characteristics of the Spanish language to do the spell checking without the need of a dictionary. In addition, this approach can serve to learn and understand the use of many common rules of the Spanish language.

Keywords: Text proofreading, expert systems, spelling rules, algorithms.

1. Introducción

Los correctores ortográficos son parte de muchos paquetes de software para la oficina, entre los que se cuentan los procesadores de palabras. LibreOffice, OpenOffice y Microsoft Word son tres ejemplos de programas que contienen un procesador de palabras con corrector ortográfico, el cual en su forma más general, utiliza un enorme

diccionario de términos en donde se busca que las palabras estén correctamente escritas. Este tipo de software hace búsquedas muy rápidas y en general trabaja adecuadamente.

Sin embargo, en el idioma español existe un número grande de reglas ortográficas las cuales definen, sin ambigüedad, cómo deben escribirse muchas palabras. Un buen corrector ortográfico, sin embargo, bien puede apoyarse de otras técnicas para hacer su tarea. Esto es, debe ser capaz de:

– Usar un diccionario con unos 500,000 términos.
– Usar diccionarios personalizados.
– Usar otras tecnologías para buscar errores, como patrones equivocados de letras.
– Usar un diccionario de verbos ya conjugados.
– Usar reglas ortográficas.

Un sistema híbrido sin duda es mejor que un sistema que sólo sigue una técnica en particular. Nosotros nos concentraremos en el sistema experto de corrección basado en reglas ortográficas, aunque mencionaremos al final del artículo la implementación de las otras posibilidades.

2. Sistemas expertos

Un sistema experto es "un programa de computadora inteligente que usa conocimiento y procedimientos de inferencia para resolver problemas que son los suficientemente difíciles para requerir la expertez significativa de un ser humano para su solución" [6]. Dicho de otra manera, un sistema experto es un sistema que **emula** las habilidades de decisión de un experto humano. Los sistemas expertos trabajan en dominios restringidos y en general contienen tres grandes apartados:

1. Interfaz con el usuario.
2. Base de conocimientos.
3. Sistema de inferencia.

En el caso de la corrección ortográfica, considerando el número de reglas sobre el uso de las diferentes palabras, se puede utilizar un sistema experto que, a través de la aplicación de las reglas ortográficas conocidas, pueda decidir (sin necesidad de consultar un diccionario de palabras necesariamente) si las palabras están bien escritas o no.

El idioma español consigna centenares de reglas [10], las cuales son la parte fundamental de la base de conocimientos del sistema experto. Por lo que se refiere al sistema de inferencia, se usará el que utiliza por definición Prolog, que es encadenamiento hacia atrás, backward chaining, el cual se usa en un número de sistemas expertos.

Finalmente, la interfaz del usuario se plantea como un proceso por lotes (batch), (en Prolog), en donde el usuario le indica al sistema que revise el texto escrito para ver si hay errores ortográficos, es decir, no se hace la corrección en tiempo real. Sin embargo, también se ataca el problema de hacer corrección después de que el usuario escribe alguna palabra en el procesador de palabras (tiempo real), incluso usando aplicaciones comerciales como Word de Microsoft.

3. Trabajo relacionado

La verificación y corrección de textos es parte cotidiana en el mundo digital. Por años se ha trabajado en el tema, particularmente en el idioma inglés. Se ha encontrado que los errores ortográficos son consecuencia de diferentes factores:

1. **Errores tipográficos**: Ocurren cuando el usuario escribe mal una palabra por error. Aquí no se puede adjudicar el error a un criterio lingüístico sino a errores en la escritura. En [3] se muestra que el 80 % de los errores tipográficos caen en alguna de estas categorías:
   a) Inserción extra de una o más letras.
   b) Borrado de una o más letras.
   c) Sustitución de una letra por otra.
   d) Transposición de dos letras adyacentes.
2. **Errores cognitivos**: Ocurren cuando el usuario no conoce la manera en cómo se escribe las palabras (simplemente ignorancia). En muchos casos esto tiene que ver la manera en cómo el usuario pronuncia las palabras [8].

Existen dos tipos de software: los verificadores ortográficos y los correctores ortográficos. Un verificador tiene una tarea relativamente sencilla: debe identificar, dado un archivo de texto (entrada), las palabras que están mal escritas. En cambio, un corrector ortográfico tiene que hacer dos cosas: detectar las palabras mal escritas y tratar de hallar cómo se escriben dichas palabras correctamente [12].

Existen muchas técnicas para verificar y corregir un texto. Una de las más usadas es la búsqueda de las palabras en un diccionario (dictionary lookup) [2], cuya única dificultad consiste en tener un diccionario de palabras lo suficientemente grande de forma que pueda abarcar gran parte del idioma de los usuarios escritores. Normalmente se usa un algoritmo de búsqueda binaria, por ser muy rápido y eficiente [13], aunque también hay esquemas como las búsquedas por hash [4].

Por otra parte, la idea de utilizar reglas ortográficas no es nueva, pero aparentemente sólo se ha aplicado a idiomas como el inglés [9] o incluso sueco [1, 5]. Esto ocurre igualmente en lo que se refiere a crear sistemas expertos para esquemas de corrección ortográfica [14, 11].

4. El sistema experto de correción ortográfica por reglas

Si consideramos que un sistema experto puede validar la idea de hacer corrección ortográfica usando reglas, entonces debemos describir los elementos de lo que consta el software en cuestión.

4.1. Descripción de las reglas en Prolog (base de conocimientos)

Tenemos tres posibilidades en las reglas ortográficas[1].

[1] https://www.urosario.edu.co/Centro-Multicultural-y-Multilingue-old/Centro-de-Lectura-y-Escritura-en-Espanol/Material-de-Apoyo/Material-de-Apoyo/Material-de-apoyo-Uso-de-las-letras-B-y-V/

Palabras que **contienen** una o varias letras específicas:

Se escriben con *B*, las palabras que lleven *rr* en su escritura.

Excepciones: ferroviario, corrosivo, verruga, correctivo, verrojo.
Ejemplos: barrer, arrabal, borrador, becerro, berrear, burro.

Palabras que **terminan** con una o varias letras específicas:

Se escriben con *B*, las terminaciones en *bundo, bunda, bilidad*.

Excepciones: movilidad, civilidad.
Ejemplos: vagabundo, nauseabunda, amabilidad, afabilidad, habilidad.

Palabras que **empiezan** con una o varias letras específicas:

Se escriben con *B*, las palabras que comienzan con *bur, bus, buz*.

Ejemplos: burla, buzo, buscar, buzón, burócrata, busto.

Esto nos hace pensar que en muchas ocasiones no hay necesidad de revisar cada palabra (candidata a una corrección), usando un diccionario, sino que bien podríamos aplicar la regla ortográfica para hallar si está bien o mal escrita. El español, como todo lenguaje humano es cambiante.

Palabras, expresiones y reglas que se usaban en el pasado son obsoletas ahora y considerando esto, se decidió mantener las reglas ortográficas en un archivo especial, el cual es consultado cada vez que el sistema se ejecuta. Este archivo, en el software creado, se denomina Reglas.DB, y contiene alrededor de 260 reglas ortográficas de uso común en el idioma español. Las reglas pueden ser editadas con cualquier procesador de palabras (o editor de textos) que utilice el formato ASCII sin caracteres de control o símbolos especiales.

Por ejemplo, el block de notas. Cada una de las reglas ocupa un renglón en el archivo Reglas.Db. Así entonces, escribir una regla nueva significa, finalmente, agregar una línea más al archivo ya mencionado. Las reglas ortográficas, para que puedan ser entendidas por el programa, requieren estar en un formato específico para poder ser leídas por el sistema. Las reglas ortográficas tienen tres posibles alternativas, las cuales pueden aplicar a:

– Prefijo (p) de la palabra analizada (parte inicial de la palabra en cuestión).
– Sufijo (s) de la palabra analizada (parte final de la palabra en cuestión).
– Subcadena (sb) de la palabra analizada (en cualquier parte de la palabra en cuestión).

Las letras “p”, “s”, y “sb” corresponden respectivamente a prefijo, sufijo y subcadena, y estas letras serán usadas para informarle al corrector en qué parte de la palabra se aplica la regla ortográfica.

La regla entonces sigue el siguiente formato:

letra palabra clave lista

Cuya descripción puede verse aquí:

- **letra**. Indica la letra a la cual se aplica la regla. Por ejemplo, si la regla ortográfica es sobre el uso de la v, éste es el parámetro que se utiliza. (Véanse los ejemplos más adelante).
- **palabra**. Indica la palabra que no cumple precisamente con la regla que está siendo definida, es decir, muestra el ejemplo de la contraposición a la regla ortográfica misma.
- **clave**. Es exactamente lo que indica el alcance de aplicación de la regla (prefijo, sufijo o subcadena). Utilícese solamente las siguientes palabras claves (entre doble comillas: p, s, sb).
- **lista**. Se refiere a la lista de palabras que son la excepción a la regla en cuestión. Tales palabras deben aparecer entrecomilladas y separadas por comas.

Algunos ejemplos ilustrativos. Considérese la siguiente regla:

Las palabras que empiezan con env se escriben siempre con v

La regla en el archivo Reglas.DB se escribirá entonces así:

v enb p

El software reconoce la regla de la siguiente manera: Las excepciones a las palabras que empiezan con *env* nada más pueden ser aquellas que comienzan con *enb*, que en este caso no hay tales excepciones a la regla y el rango de aplicación de la misma es al inicio (por eso le denominamos prefijo) de las palabras. Ahora considérese la siguiente regla:

Todas las palabras que terminan con ave se escriben con v.

Esta regla puede expresarse en el lenguaje descrito de la siguiente forma:

v abe s árabe jarabe

La cual puede ser descrita de la siguiente manera: Las excepciones a las palabras que terminan en *ave* nada más pueden ser aquellas que terminan con *abe*, las cuales son, *árabe, jarabe* (y nada más), y el rango de aplicación de la regla son los sufijos de las palabras. Por último, un ejemplo de una regla que se refiera a subcadenas:

Las palabras que tienen dentro de ellas (en cualquier parte de la misma) las letras ilv se escriben siempre con v.

La cual se traduce en el archivo de reglas de la siguiente forma:

v ilb sb bilbao

Y se interpreta de la siguiente manera: Las excepciones a las palabras que tienen como subpalabra *ilv* se escriben siempre con *v*, excepto la palabra *bilbao*. Un fragmento del archivo de reglas es este (ya con la sintaxis legal de Prolog):

```
ex(“v”,“enb”,[],“p”)
ex(“v”,“db”,[],“sb”)
ex(“v”,“nb”,[],“sb”)
ex(“v”,“lb”,[“elba”],“sb”)
ex(“v”,“ilb”,[“bilbao”],“sb”)
ex(“v”,“clab”,[],“sb”)
ex(“v”,“mob”,[“mobiliario”],“sb”)
ex(“v”,“seb”,[“sebo”,“sebastián”,“sebastopol”],“p”)
```

Las excepciones se encuentran como una lista de Prolog tradicional, entre paréntesis cuadrados.

4.2. Mecanismo de inferencia

En un sistema de reglas, como en el caso que nos ocupa, el mecanismo de inferencia determina cómo utilizar las mismas. El mecanismo natural en Prolog es el encadenamiento hacia atrás, el cual se basa en plantear una hipótesis (lo que queremos demostrar), para así llegar a la conclusión deseada.

Por ejemplo, en un sistema de diagnóstico médico, un síntoma podría ser que el paciente tenga congestión nasal, lo cual se toma como la hipótesis. A partir de esto el sistema podría llegar a la conclusión de que el paciente tiene gripa si se demuestran otros síntomas, es decir, si se satisfacen los hechos en las reglas [6].

Para el caso de las reglas ortográficas, al recibirse la palabra a corregir, podemos empezar por la hipótesis (la regla), de que dicha palabra está mal escrita a partir de la primera regla en el archivo Reglas.DB. Si no cumple con las características de esa regla particular, el esquema de inferencia la desecha y continúa con la siguiente regla. Prolog siempre hace una búsqueda exhaustiva con todas las reglas definidas.

4.3. Interfaz con el usuario

El sistema experto de reglas incluyó originalmente un editor con las características del antiguo programa WordStar[2], el cual era el estándar de facto y que, además, era un predicado que venía incluido en Turbo Prolog 2.0[3]. Este software no trabaja en tiempo real, sino por lotes (batch), de forma que las correcciones quedan grabadas todas en un archivo de texto, el cual contiene las palabras ofensoras (o equivocadas), y lo que debe hacerse para corregirlas. Sin embargo, el sistema contempla una opción para hacer una pausa cada vez que se genera una corrección.

[2] WordStar fue un procesador de textos, incluido en las computadoras Osborne 1. En particular, WordStar fue el último procesador de textos comercial para el sistema operativo CP/M y fue lanzado en septiembre de 1978.

[3] Lo que quiere decir que un predicado de Turbo Prolog genera todo el editor de textos, similar al que se usa en el IDE de este sistema.

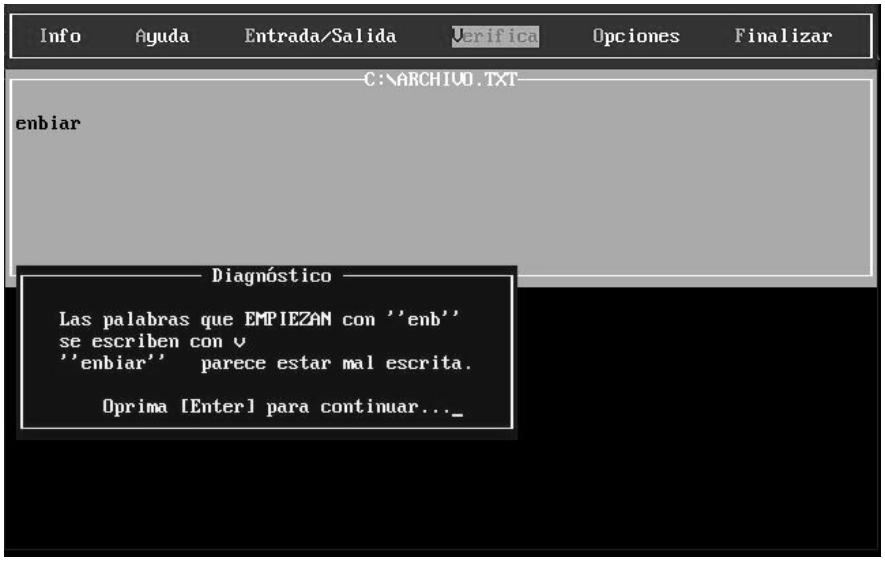


Fig. 1. La corrección que hace Prolog, indicando la regla usada.

Sin embargo, una vez que la prueba de concepto mostró el funcionamiento del corrector ortográfico usando reglas, se decidió implementar una versión que fuese capaz de trabajar con el estándar actual, Microsoft Word, a través de la comunicación entre procesos. Esto puede verse en la sección de resultados.

5. Resultados preliminares

Para las primeras pruebas, se escribió un programa en Turbo Prolog 2.0[4], el cual funciona bajo MsDOS (usando DOSBox)[5], pero que da algunas facilidades para depurar fácilmente el software. De hecho, todo el sistema experto de reglas ortográficas trabaja sobre archivos de texto ASCII sin caracteres de control. El software lee ese archivo de texto y lo separa, palabra por palabra, el cual se pasa como parámetro a la base de conocimiento de las reglas.

Entonces, mediante el encadenamiento hacia atrás, Prolog revisa la palabra y busca saber si está bien o mal escrita. En caso de hallar que la palabra esté mal escrita, el sistema mandará un mensaje que indica el error –de acuerdo a la regla– y cómo debe solucionarse. En caso de que la regla no se cumpla, a través del mecanismo del backtrack, se buscará si cumple con la siguiente regla y así sucesivamente.

Prolog revisa todos los nodos del árbol solución.

[4] Turbo Prolog 2.0 es un sistema completo de desarrollo de software que incluye un compilador y un entorno de desarrollo integrado (IDE) para el lenguaje de programación PROLOG, desarrollado por Borland.

[5] DOSBox es un emulador que recrea un entorno parecido al sistema DOS con el objetivo de poder ejecutar programas y videojuegos originalmente escritos para el sistema operativo MS-DOS de Microsoft en computadoras más modernas o en diferentes arquitecturas.

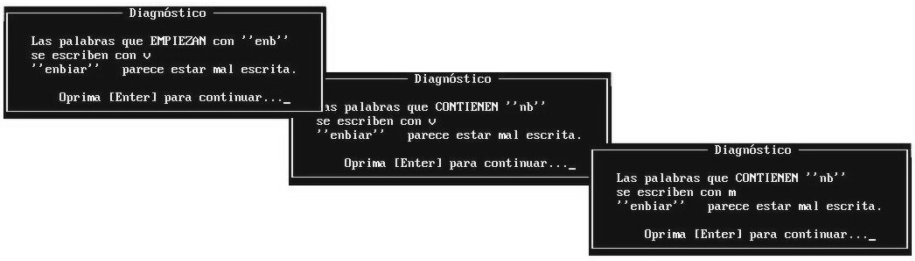


Fig. 2. Más de una regla se dispara con la palabra “enbiar”.

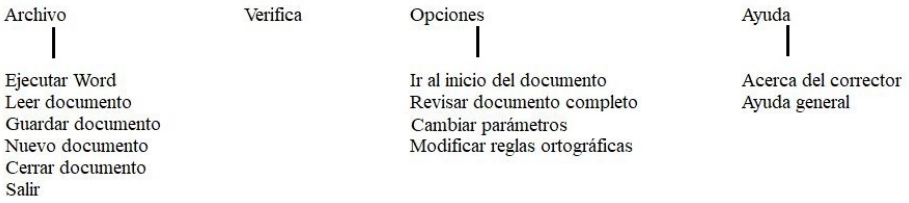


Fig. 3. La estructura del programa de corrección por reglas, en Windows.

La figura 1 muestra cómo el sistema hace la corrección ortográfica, indicando incluso el uso de la regla: Cabe señalar que más de una regla puede ser desplegada. Por ejemplo, en la figura 2 puede verse que la palabra “enbiar” consulta más de una regla.

5.1. Enlazando el corrector con MsWord en Windows

Escribir un procesador de palabras que contenga un corrector ortográfico es una labor que requiere de mucho trabajo. Además, es difícil que las personas cambien de software pues ya hay un apego y costumbre a usar lo que ya han aprendido. Hoy los usuarios procesan textos con algún programa de una suite de oficina.

El estándar es Microsoft y la mayoría de los usuarios usan Word para Windows. Considerando esto, la idea de probar el corrector ortográfico por reglas (bautizado como “Lapsus”), debiese hacerse enlazando éste a Word de alguna manera.

Afortunadamente Microsoft ofrece un mecanismo de comunicación entre su suite de oficina y aplicaciones de terceros. De esta manera, podemos pedirle a Word que mande las palabras a analizar al corrector ortográfico y este último le responde precisamente con los mensajes de corrección, palabra por palabra, en tiempo real si se desea.

Esto se realiza a través de la automatización OLE (Object Linking and Embedded), que es el mecanismo de comunicación entre dos procesos que corren bajo Windows. Para nuestro propósito, usamos OLE y Delphi[6]. Para ello nos basamos en los artículos de Ron Gray [7].

[6] Delphi es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual, basado en Object Pascal, ya con 26 años de existencia.

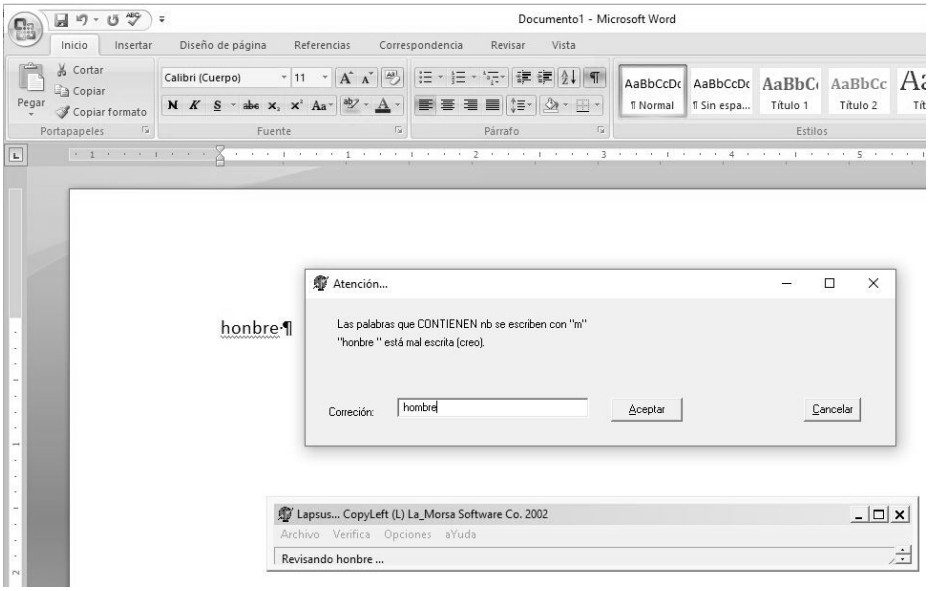


Fig. 4. Lapsus detectando el error en la palabra “honbre”.

En la figura 3, se describe la estructura del corrector Lapsus, el cual es un programa escrito en Delphi que se comunica con Word a través de la automatización OLE:

Hay que señalar que la implementación del sistema experto en Delphi requiere emular las características que tiene Prolog [15], de buscar exhaustivamente la solución, moviéndose por todos los nodos en el árbol de búsquedas, en este caso, en todas las reglas definidas en el sistema. Una vez implementada esta parte, se hicieron las pruebas correspondientes y en la figura 4 puede verse el corrector trabajando sobre la palabra “honbre”, evidentemente mal escrita.

6. Resultados, conclusiones y trabajo futuro

Se utilizó una computadora AMD Ryzen 5 PRO 1600 con seis procesadores, corriendo a 3.20 GHz y con 16 GB de RAM.

6.1. Resultados usando DosBox y Turbo Prolog 2.0

Para el primer lote de pruebas se usó DosBox (que emula MsDOS) y Turbo Prolog 2.0. Por ende, la memoria disponible no sobrepasa los 640 Kbytes y no hace uso de la memoria extendida. Se midieron tiempos de corrección en diferentes escenarios. Usando el texto:

“La noche estaba muy oscura. Tenía la ventana abierta y por ella me llegaba el resplandor de un anuncio de cerveza Superior. No se veían las estrellas, no había luna, nada más el rostro sonriente de la rubia de categoría, y fugaz intermitente, la luz roja

de una frase: 'La rubia que todos quieren es cerveza Superior'. La habitación, entre los ires y venires de la luz exterior, quedaba abandonada al humor pajizo de las menguadas veladoras. Del otro lado, en el cuarto contiguo, solamente se escuchaba el radio (ahora un locutor decía con voz trémula: 'Su programa favorito, Serenaataaa. Su estación, la B grande de México.'), de Sealtiel Alatriste[7], el sistema tardó 21.2 segundos, usando solamente un diccionario de 500 mil palabras, es decir sin utilizar las reglas ortográficas.

El sistema pudo revisar unas 5.09 palabras por segundo, teniendo el texto 107 palabras. Usando el mismo texto y solamente las reglas ortográficas, el sistema tardó 13.02 segundos, lo que representa 8.29 palabras revisadas por segundo.

Para probar la efectividad del corrector usando solamente reglas ortográficas, se encontró que el sistema pudo analizar –de un texto con 65 palabras mal escritas a propósito (todas con problemas en su construcción ortográfica)– en 12.03 segundos, logrando una efectividad de 5.40 palabras por segundo.

Sin embargo, haciendo el mismo proceso incluyendo el diccionario de 500 mil términos, el sistema tardó 27.62 segundos, logrando unas 2.35 palabras por segundo analizadas. Cabe decir que el total de reglas contempladas es de alrededor de 260, las cuales son aplicadas a cada palabra a corregir.

6.2. Resultados usando Windows 10, MsWord y Lapsus en Delphi

En el segundo lote de pruebas, usando el sistema escrito en Delphi (dentro de Windows 10), y conectado vía OLE con MsWord, se encontró que, para el texto de Alatriste –que consta de 107 palabras– el sistema pudo revisarlo en 5.04 segundos, usando solamente las reglas ortográficas. Esto muestra una eficiencia de 21.23 palabras analizadas por segundo.

Para el texto con 65 palabras mal escritas, es decir, que no cumplen con alguna de las reglas ortográficas, Lapsus tardó 2.77 segundos en hallar todos los problemas ortográficos usando solamente su conjunto de reglas. Se logró una eficiencia de unas 23.4 palabras por segundo en la versión para Windows.

Cabe señalar que en la versión de Lapsus para Windows, no se utiliza el diccionario de 500 mil términos, pues se usa el que viene integrado en Word. Se esperaba –como ocurrió– que este segundo lote de pruebas presentara mayor velocidad pues el sistema no está limitado por la memoria como en el caso de DosBox.

6.3. Conclusiones

La corrección de textos no es un asunto trivial, pues los idiomas contienen no solamente muchísimas palabras, sino que también hay giros idiomáticos, palabras que pueden considerarse erróneas (por ejemplo, "revolver" y "revólver", en donde dependiendo del contexto son palabras correctamente escritas), acentos diacríticos, etcétera. No obstante, lo que queda claro es que la corrección ortográfica basada solamente en un diccionario con cientos de miles de palabras no es suficiente.

[7] http://www.materialdelectura.unam.mx/index.php/cuento-contemporaneo/13-cuento-contemporaneo-cat/241-111-sealtiel-alatriste?start=4

Este es un primer enfoque que soluciona dificultades pero que requiere entender que se puede sacar ventaja de la estructura del propio idioma para hacer un corrector mucho más robusto. La corrección con reglas ortográficas para el idioma español es un primer paso a futuros correctores mucho más inteligentes.

6.4. Trabajo futuro

Un esquema híbrido parece ser mucho mejor idea: combinar un diccionario de muchísimos términos, además de diccionarios especializados (por ejemplo, de medicina, de términos legales), además de un esquema de verificación basándose en sílabas válidas en el español e incluso, el uso de diccionarios con palabras que no existen en el español (por ejemplo, nombres de empresas), pueden hacer más robusto el sistema.

Igualmente los futuros correctores deben analizar no solamente las palabras, sino frases enteras y si es necesario, sugerir o indicar errores. Es probable que un análisis estadístico de cómo se usan las palabras y los contextos en donde aparecen, puede ser un camino nuevo en la corrección de textos. La corrección ortográfica es sin duda un tema abierto en cómputo.

Referencias

1. Arppe, A.: Developing a grammar checker for swedish. In: Proceedings of NODALIDA, pp. 13–27 (1999)
2. Avarbuch, L. M.: Lookup: Interactive spell checker for single words. vol. 15, no. 1, pp. 33–34 (1991) doi: 10.1108/eb024363
3. Damerau, F. J.: A technique for computer detection and correction of spelling error. Communication ACM, vol. 7, no. 3, pp 171–176 (1964) doi: 10.1145/363958.363994
4. Damgard, I. B.: A design principle for hash functions. In: Brassard, G. (eds) Advances in Cryptology - CRYPTO' 89 Proceedings, CRYPTO 1989, Lecture Notes in Computer Science, vol 435, pp. 416–427 (1990) doi: 10.1007/0-387-34805-0_39
5. Domeij, R., Knutsson, O., Carlberger, J., Kann, V.: Granska: An efficient hybrid system for Swedish grammar checking. In: Proceedings of the 12th Nordic Conference of Computational Linguistics (1999), pp. 49–56 (2000)
6. Gary, R., Giarratano, J.: Expert Systems, principles and programming. PWS Publishing Company (1998)
7. Gray, R.: Word control. Delphi Informant Magazine, vol. 6, no. 9 (2000)
8. Kumar, R., Bala, M., Sourabh, K.: A study of spell checking techniques for Indian languages. JK Research Journal in Mathematics and Computer Sciences, vol. 1, no. 1 (2018)
9. Naber, D.: A rule-based style and grammar checker. Diplomarbeit Technische Fakultät, Universität Bielefeld (2003)
10. Ortografía Lengua Española. Reglas y Ejercicios, Larousse (2001)
11. QasemiZadeh, B., Ilkhani, A., Ganjeii, A.: Adaptive language independent spell checking using intelligent traverse on a tree. In: IEEE Conference on Cybernetics and Intelligent Systems, pp. 1–6 (2006) doi:10.1109/ICCIS.2006.252325
12. Peterson, J. L.: Computer programs for detecting and correcting spelling errors. Communications of the ACM, vol. 23, no. 12, pp 676–687 (1980) doi: 10.1145/359038.359041

13. R. Nowak: Generalized binary search. In: 46th Annual Allerton Conference on Communication, Control, and Computing, pp. 568–574 (2008) doi: 10.1109/ALLERTON.2008.4797609
14. Sokele, M., Dembitz, Š., Knežević, P.: Developing a spell checker as an expert system. Journal of Computing and Information Technology, vol. 11, no. 4 (2003) doi: 10.2498/cit.2003.04.03
15. Sawyer, B., Foster, D.: Programming expert systems in Pascal. Wiley (1986)

ISSN 1870-4069

Sistema CADx para la clasificación de cáncer de mama basado en Técnicas de Transfer Learning y Pseudocolor

Oscar García-Ávila, José A. Almaraz-Damián, Volodymyr Ponomaryov, Rogelio Reyes-Reyes, Clara Cruz-Ramos

Instituto Politécnico Nacional,
ESIME Culhuacán,
México

ogarciaa2104tmp@alumnoguinda.mx,
jalmarazd1401@alumno.ipn.mx,
{vponomar, rreyesre, ccruzra}@ipn.mx

Resumen. La detección temprana del cáncer de mama es crucial en el tratamiento de esta enfermedad, como primer instrumento de diagnóstico se tiene la mamografía, la cual es uno de los estudios por imagen más utilizados por el especialista debido a su forma no invasiva. En este artículo se presenta un Sistema de Detección Asistida por Computadora (CADx) para el análisis de imágenes de mamografía digitales, el cual, clasificara si la imagen presenta una lesión de tipo maligna o benigna. Los métodos que se utilizan en el sistema CADx propuesto son el uso de la Transferencia de Aprendizaje, Maquinas de Soporte Vectorial y reducción de características a través del Análisis de Componentes Principales (PCA). El sistema obtuvo resultados favorables en comparación de los métodos del estado del arte utilizando las métricas de calidad tales como Exactitud, Especificidad, Sensibilidad, Medida-F1.

Palabras clave: Mamografía, pseudocolor, CADx, CNN, PCA, SVM, transferencia de aprendizaje.

Breast Cancer Classification Via CADx System Based on Transfer Learning and Pseudocolor

Abstract. Early detection of breast cancer is crucial in the treatment of this decease, as the principal diagnoses tool mammography is employed, which is one of the most used medical studies due to no invasive form. In this paper, a Computer-Aided Detection System (CADx) is presented for the analysis of digital mammogram images. The methods used in the proposed CAD system are Transfer Learning, Support Vector Machine Classifier, and Feature Reduction based on Principal Component Analysis. The system has demonstrated improved performance in comparison with state-of-the-art methods in terms of quality metrics such as Accuracy, Specificity, Sensibility, and F1-Score.

Keywords: Mammograph, pseudocolor, CADx, CNN, PCA, SVM, transfer learning.

1. Introducción

Cáncer es el nombre que se le da a diversas enfermedades las cuales están relacionadas al crecimiento desproporcionado de células, el cual se debe a un daño en el ciclo de reproducción de las células, este puede ser ocasionado por diferentes factores como estilo de vida, ambientales o genéticos. Esto causa la formación de una masa referida como tumor o neoplasia, normalmente estas masas pueden ser de tipo maligno o benigno [14, 20]. Los tumores benignos crecen a un ritmo no acelerado, además de que no se diseminan o infiltran a partes vecinas y comúnmente al extirpase no reaparecen.

Los tumores malignos se pueden expandir a tejidos cercanos o infiltrarse a través del sistema linfático y circulatorio, creando nuevos tumores en distintas partes del cuerpo usualmente al extirparse estas tumoraciones pueden reaparecen en el mismo sitio. El cáncer de mama es una de las principales enfermedades mortales que afectan a las mujeres de México y el mundo. De acuerdo con la Organización Mundial de la Salud (OMS) [33] cada año se detectan 1.38 millones de nuevos casos y fallecen 458000 por este padecimiento.

En México, representa la primera causa de muerte en mujeres y la tasa de mortalidad es de 17 defunciones por cada 100000 mujeres entre 20 años o más [9]. La manera eficaz de erradicar esta enfermedad es la detección temprana [13]. Los estudios de imagenología que nos permiten realizar un diagnóstico temprano del cáncer de mama es la mamografía (MG), la cual es considerada en primer lugar debido a que es el método más práctico para obtener una representación visual del área sospechosa y es accesible a la población en general debido a su bajo costo.

De ser necesario que especialista requiera el análisis con mayor detalle del área sospechosa se realizan estudios complementarios de imagenología como la Ecografía (US), Imagen por Tomografía Computarizada (CT), Imagen por Resonancia Magnética (MRI y, por último, en caso de ser necesario se procede a realizar una biopsia, en la cual se extrae una porción de tejido mamario el cual se examina a través de un estudio histopatológico para determinar su malignidad [4].

El sistema Breast Imaging Reporting and Data System (BI-RADS) es un esquema el cual nos permite clasificar de manera estandarizada los hallazgos obtenidos en una imagen MG. Este permite determinar si en el tejido mamario contiene alguna masa y si esta puede ser de tipo maligno o benigno, además de categorizar el grado de sospecha de malignidad o benignidad de la lesión con el fin de brindar un diagnóstico de forma no invasiva y por consiguiente el proceso que se debe seguir el especialista para su tratamiento [2, 27], la categorización del sistema BI-RADS se detalla a en la siguiente tabla 1:

Los sistemas de Diagnostico Asistido por Computadora (CAD) son sistemas que son capaces de realizar el procesamiento de datos con el fin de ayudar a los especialistas médicos. Estos sistemas se consideran inteligentes debido a que usan la retroalimentación para adquirir nuevos conocimientos y así mejorar su rendimiento. Se consideran 2 tipos de estos sistemas, los dedicados a la detección de alguna anomalía normalmente llamados Detección Asistido por Computadora (CADe) o los dedicados a determinar un diagnostico denominados Diagnostico Asistido por Computadora (CADx) [35]. Un sistema CADx consta de básicamente 4 etapas:

Tabla 1. Resumen de sistema BI-RADS.

Categoria BI-RADS	Definición	Accción
0	Incompleto	Se necesita una evaluacion adicional.
1	Negativo	No hay sospecha para reportar.
2	Benigno	Se describe como un nodulo benigno.
3	Lesión probablemente benigna	Sugiere un nuevo estudio a corto plazo.
4	Anormalidad Sospechosa	Consideracion de toma de muestra por Biopsia.
5	Alta sospecha de Malignidad	Debe considerarse intervencion por el especialista.
6	Malignidad confirmada por biopsia	

1. **Preprocesamiento**: En esta etapa se hacen mejoras a la imagen como ajustes del contraste, disminución del ruido, entre otros.
2. **Segmentación**: En esta etapa de la imagen del estudio se obtiene el área donde se encuentra la anomalía o lesión que se analizara, es decir se obtiene la región de interés (ROI).
3. **Extracción de características**: Se obtienen algunas características de la región de interés para realizar su posterior clasificación. Algunas características que se obtienen son forma, color, textura, entre otras.
4. **Clasificación y evaluación**: A partir de la extracción de las características establecidas, en esta el sistema deberá ser capaz de determinar las características sobresalientes por cada clase y así poder discriminar entre una clase u otra.

En este trabajo se propone un Sistema CADx para el análisis de imágenes de mamografía, el cual, clasificara si la imagen presenta una lesión de tipo maligna o benigna. Los métodos que se utilizan en el sistema CADx propuesto son el uso de la Transferencia de Aprendizaje, Maquinas de Soporte Vectorial (SVM) y la reducción de características a través del Análisis de Componentes Principales (PCA). El sistema obtuvo resultados favorables en comparación de los métodos del estado del arte utilizando las métricas de calidad como Exactitud, Especificidad, Sensibilidad y Medida-F.

2. Trabajos relacionados

Tsochatzidis et al. [32] utilizan dos esquemas para la clasificación binaria (benigno y maligno) de imágenes mamográficas. A partir de la extracción manual de la Región de interés (ROI) de las imágenes obtenidas de la base de datos DDSM-400, posteriormente se aplica un redimensionamiento de las imágenes como extractor de características utilizan arquitecturas basadas en Redes Neuronales Convolucionales (CNN), las arquitecturas utilizadas son AlexNet [16], VGG-16/19 [26], ResNet [12], GoogLeNet [29] e Inception-BN v2 [30]. En el primer esquema el clasificador es remplazado y entrenado desde 0 y en el segundo esquema el clasificador es afinado es decir se inicializan con los pesos establecidos y en cada imagen se actualizan.

Los mejores resultados fueron obtenidos en el segundo esquema con la red ResNet-101 con $\mathrm{AUC} = 0.859$ y $\mathrm{Exactitud} = 0.785$. Ragab, et al. [23] proponen un sistema CADx usando la arquitectura de Red Neuronal Convolucional Alexnet para extraer las características de la base de datos DDSM y CBIS-DDSM para la clasificación binaria de lesiones contenidas en la mama, extrayendo ROI de manera manual y por medio de una técnica de umbralización basada en el valor de los pixeles de la imagen.

En la etapa de clasificación reemplazan el clasificador original de la arquitectura Alexnet por un clasificador de tipo Totalmente Conectado y finalmente conectado a una Maquina de Soporte Vectorial. Los mejores resultados obtenidos son: $\mathrm{Exactitud} = 0.872$, $\mathrm{AUC} = 0.94$, $\mathrm{Sensibilidad} = 0.862$, $\mathrm{Especificidad} = 0.877$, $\mathrm{Precision} = 0.88$, $\mathrm{Valor\text{-}F1} = 0.871$.

Arora, et al. [5] proponen una metodología utilizando la base de datos CBIS-DDSM para clasificar masas como malignas y benignas, analizando 1318 imágenes de ROI contenidas en esta base. Implementan las arquitecturas CNN: Alexnet, VGG16, GoogLeNet, Resnet18, InceptionResNet [28] como extractores de características posteriormente son concatenadas.

Finalmente utilizan una Red Neuronal Artificial obteniendo una $\mathrm{Exactitud} = 0.88$, $\mathrm{AUC} = 0.88$, $\mathrm{Presicion} = 0.85$, $\mathrm{Exhaustividad} = 0.91$. Lévy et al. [17] proponen un sistema CADx basado en la red AlexNet y GoogLeNet, utilizando la base de datos DDSM. Al utilizar la Trasferencia de Aprendizaje y la técnica de Aumento de Datos alcanzan una $\mathrm{Exactitud} = 0.929$ $\mathrm{Precision} = 0.924$ y $\mathrm{Exhaustividad} = 0.934$.

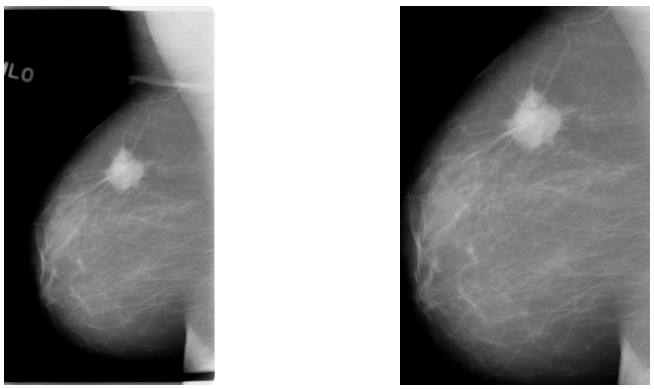
Comparando los sistemas anteriormente mencionados, la mayoría utiliza características extraídas por Redes Neuronales Convolucionales con la ayuda de la técnica de Transferencia de Aprendizaje basadas en arquitecturas del estado del arte [16, 26, 12, 29, 30, 28] para la clasificación de Imagenet [25]. Sin embargo, la desventaja de dichos métodos es la extracción manual de la Región de Interés esto perjudica a la evaluación de los sistemas debido a que se requiere el conocimiento de un especialista para determinar esta región.

3. Metodología

3.1. Desarollo CADx

El sistema CADx propuesto es ilustrado en la Fig.1. En primera instancia la imagen mamográfica es segmentada de forma manual eliminando artefactos como etiquetas, entre otros. Posteriormente se realiza una Pseudocoloración con el fin de resaltar las partes de mayor interés contenidas en la imagen mamográficas. Subsecuentemente se obtienen las características basadas en una arquitectura de Red Neuronal Convolucional, la cual fue entrenada para la tarea de clasificación. Finalmente, estas características son reducidas a partir del Análisis de Componentes Principales y Clasificadas por una Maquina de Soporte Vectorial. Los detalles de cada etapa del sistema propuesto se describen a continuación.

Fig. 1. Diagrama a bloques del sistema CADx implementado.



(a) (b)

Fig. 2. Resultados de la etapa de segmentación: a) Imagen Original b) Región de interés.

3.2. Segmentación

Al obtener una imagen de mamografía comúnmente son afectadas por artefactos como etiquetas, instrumentación médica o en el caso específico la aparición del musculo oblicuo pectoral. Para eliminar dichos artefactos se procedió a realizar un recorte manual con el fin de preservar la mayor cantidad posible del área del seno y eliminar los artefactos antes mencionados (Fig.3), este proceso se realizó para todas las imágenes utilizadas en este trabajo.

3.3. Pseudocolor

La pseudocoloración de imágenes en escala de grises es un proceso el cual se utiliza para complementar información visual de varias aplicaciones de las imágenes de Rayos-X, mejorando la detección de características, estructuras o patrones. El propósito principal de la pseudocoloración es aprovechar las capacidades perceptuales del Sistema Visual Humano [11, 18]. En este trabajo se aplica la pseudocoloración con el fin de mejorar la imagen de la lesión $I_m(x, y)$ la cual fue segmentada forma manual.

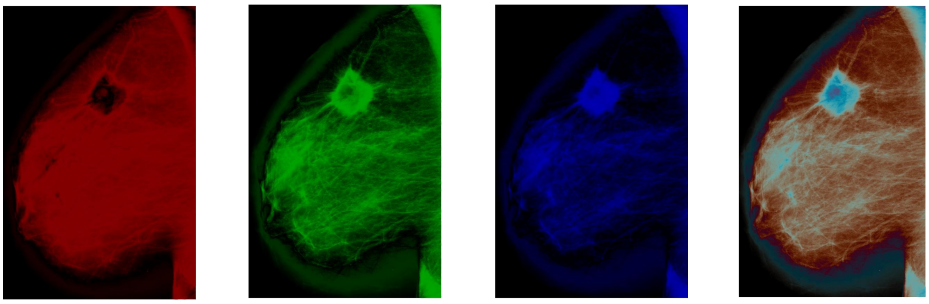


Fig. 3. Resultados de la etapa de segmentación, de izquierda a derecha: a) Imagen generada a partir de la ec.1 b) Imagen generada a partir de la ec.2 c) Imagen generada a partir de la ec.3 d) Imagen generada a partir de la ec.4.

La pseudocoloración de la imagen de la lesión $I_m(x, y)$ está basada por las ecuaciones:

$$R(x,y) = \left|\sin\left(2\pi \times \left(\frac{I_m}{255} + \frac{\pi}{2}\right)\right)\right|, \tag{1}$$

$$G(x,y) = \left|\sin\left(2\pi \times \left(\frac{I_m}{255} + \frac{\pi}{4}\right)\right)\right|, \tag{2}$$

$$B(x,y) = \left|\sin\left(2\pi \times \left(\frac{I_m}{255} + \frac{\pi}{6}\right)\right)\right|. \tag{3}$$

Finalmente, se concatenan las imágenes anteriores para generar la imagen final $I_p(x, y)$ a partir de:

$$I_p(x,y) = [R(x,y), G(x,y), B(x,y)]. \tag{4}$$

3.4. Extracción de características

Red neuronal convolucional (CNN) es un método del Aprendizaje Profundo que consiste en 2 bloques genéricos, el bloque convolucional en donde se encuentran las capas convolucionales, las capas de agrupación y las funciones de activación propuestas seguidas del bloque de clasificación. Normalmente las primeras capas de las CNN pueden detectar características básicas como círculos, líneas o bordes y las capas más profundas detectara patrones complejos y específicos para cada clase [3].

Transferencia de aprendizaje es un método del aprendizaje automático en donde un modelo utilizado para resolver una tarea dada es reusado para resolver una tarea específica [36, 21]. Comúnmente esta técnica es usada en métodos basados en Aprendizaje Profundo debido a la gran cantidad de datos que se requieren para entrenar estos métodos desde 0, lo cual es el problema principal al implementar dichos métodos. Existen dos estrategias que deben ser consideradas al utilizar este método a partir de un modelo preentrenado: El ajuste afinado y el uso del modelo como extractor de características.

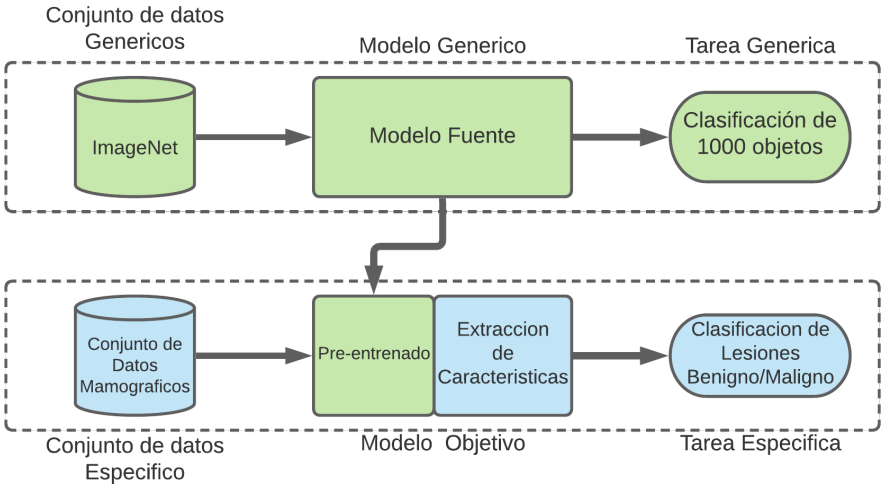


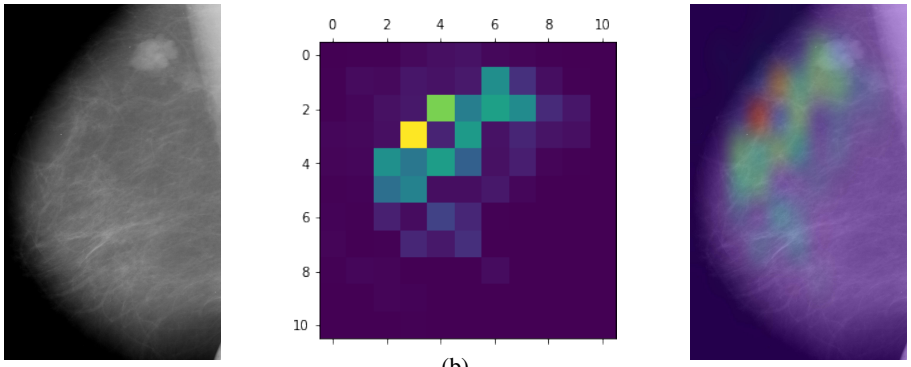
Fig. 4. Vista conceptual de la Transferencia de Aprendizaje.

El Ajuste Afinado de un modelo implica reentrenarlo desde una capa convolucional específica y reemplazar su clasificador por uno el cual, se ajuste a nuestra tarea específica, con ello este método ajusta al modelo a la nueva información que se le provee. Al usar el modelo como extractor de características, involucra remover el bloque de clasificación y obtener los valores de la última capa convolucional del modelo, con el fin de obtener las características que el modelo detecta antes de ser clasificadas, estas características se deben considerar como genéricas debido a que el modelo las detectara basadas en su conocimiento en una tarea similar.

Finalmente, para clasificar estas características se debe utilizar un clasificador adecuado para nuestra tarea específica (ver Fig.4). NASNet es una arquitectura de Red Neuronal Convolucional propuesta por Zoph et al. [37], para la base de datos CIFAR-10 y posteriormente modificada para ILSVRC [25] en el cual obtuvo el $82.7\,\%$ de Precisión. Esta arquitectura contiene 2 tipos de células llamadas células normales y de reducción.

Las células normales contienen operaciones convolucionales, que devuelven un mapa de características de una misma dimensión, mientras que las células de reducción el mapa de características es reducido por un factor 2, con el fin de reducir el ancho y el alto del mapa de características obtenido. En este trabajo se utiliza la arquitectura $\mathrm{NASNet\text{-}A}\ 4@64$ donde el primer número (4) es el número de células que se repiten y el segundo (64) indica el número de filtros que se encuentran en la penúltima capa de la red obteniendo finalmente 4032 características.

En este trabajo, se utiliza el método de Extracción de Características basado en la arquitectura NasNet [37] la cual fue entrenada en la tarea de clasificación ILSVRC [25]. Grad-CAM propuesto por Selvaraju et al. [6] este algoritmo permite visualizar las características más relevantes para la Red Neuronal Convolucional al realizar la predicción de una imagen utilizando el gradiente la última capa convolucional de la arquitectura utilizada. En este trabajo se utiliza para garantizar que la arquitectura detecte los patrones extraídos dentro de la Región de interés.



(a) (b) (c)

Fig. 5. Resultados de la etapa de visualización por Grad-CAM: a) Imagen Original, b) Mapa de Activación obtenido por Grad-CAM, c) Interpolación de la imagen a) y b).

3.5. Reducción de características

Posteriormente de extraer las características de tipo profundas, es necesario reducir el número de ellas ya que una gran cantidad incrementa el tiempo de cómputo para realizar una predicción. Para resolver este problema el Análisis de Componentes Principales (PCA) es empleado.

El Análisis de Componentes Principales este método estadístico se utiliza para transformar un conjunto de datos de p-dimensiones en otro conjunto de datos de q-dimensiones llamados componentes, proyectando el vector original en básicamente una menor dimensión [10].

PCA es usado para obtener los atributos óptimos para la etapa de clasificación donde las ventajas de usar PCA son evitar el sobreajuste y mejora la precisión en la predicción [22]. Para este trabajo se encontro que el resultado óptimo corresponde a obtener 404 componentes.

3.6. Clasificador

El clasificador utilizado en este trabajo es la máquina de soporte vectorial la cual pertenece a los algoritmos de aprendizaje automático supervisado. Está basado en el concepto de planos de decisión los cuales, son capaces de separar clases que se representan como conjuntos de puntos a partir de un hiperplano el cual, es generado por un subconjunto de elementos de las dos clases llamados vectores de soporte [8].

En muchos casos el conjunto de datos no puede separarse con precisión por un hiperplano, por lo que se utiliza una función llamada núcleo. Algunos de los núcleos comúnmente utilizados son el Lineal, Polinomial, Función de Base Radial (RBF) este último, asigna los datos originales a un nuevo espacio de identidades en el que se puede encontrar separabilidad entre las clases. En el presente trabajo se implementa un SVM con núcleo RBF con parámetros $C = 1$ y $\gamma = \frac{1}{\#\,\text{elementos}}$.

4. Resultados experimentales

El método descrito fue implementado en la plataforma Google®Collaboratory en la que se asignó una plataforma Linux con 12 GB de RAM y una GPU Nvidia®Tesla K80 con 12 GB VRAM, Python 3.x y las librerías Keras, Sk-learn y TensorFlow.

4.1. Bases de datos

En este trabajo se utilizaron 2 conjuntos de datos públicos los cuales serán descritos a continuación:

Conjunto de Datos CBIS-DDSM: (Curated Breast Imaging Subset of DDSM) contiene 2620 imagenes de mamografía. Contiene casos normales, benignos y malignos con información patológica verificada por especialistas. Las imágenes se encuentran en el formato DICOM. Para este trabajo se utilizaron las imágenes que contienen lesiones de masas. La cantidad de imágenes utilizadas de la base de datos CBIS-DDSM son: 370 para entrenamiento y 121 para prueba de la clase **Benigna** y 121 para entrenamiento y 80 para prueba de la clase **Maligna**.

Conjunto de Datos UIC contiene 286 imágenes, las cuales se obtuvieron en el marco de un Protocolo aprobado por la Junta de Revisión de la Universidad de Centro Médico de Chicago. Estas imágenes se encuentran en escala de grises y formato PNG [31]. La cantidad de imágenes utilizadas de esta base de datos son: 111 de la clase **Benigna** y 175 de la clase **Maligna**.

4.2. Criterios de calidad

Para evaluar el rendimiento de nuestro sistema utilizamos las siguientes métricas de calidad. Exactitud (ACC): es el número total de predicciones correctas entre el número total de muestras, está dada por:

$$\text{Exactitud} = \frac{vp + vn}{vp + vn + fp + fn}. \tag{5}$$

Tasa de verdaderos positivos o Sensibilidad (SEN): es el número de casos positivos que se predijeron correctamente como positivos con respecto a todos los casos positivos se calcula como:

$$\text{Sensibilidad} = \frac{vp}{vp + fn}. \tag{6}$$

Tasa de verdaderos negativos o Especificidad (SPC): es el número de casos negativos que se predijeron correctamente como negativos con respecto a todos los casos negativos se calcula como:

$$\text{Especificidad} = \frac{vn}{vn + fp}. \tag{7}$$

Precisión (PRE): es el número de resultados positivos correctos entre el número de resultados positivos predecidos por el clasificador.

$$\text{Precision} = \frac{vp}{vp + fp}. \tag{8}$$

Medida-F1 (F1): mide la media armónica entre precisión y sensibilidad.

$$\mathrm{F1} = \frac{2vp}{2vp + fp + fn}, \tag{9}$$

donde vp son los verdaderos positivos, vn son los verdaderos negativos, fp son los falsos positivos y fn son los falsos negativos. Para la validación de resultados se empleó la técnica de K-fold la cual, es una técnica común para evaluar el sistema en donde el conjunto de datos se divide en k-divisiones y se entrena un clasificador usando $K - 1$ divisiones, y un valor de error se calcula probando el clasificador en el conjunto restante, para este trabajo se propuso $k = 5$ obteniendo resultados comparativos con los encontrados en [32, 23].

Desde nuestro punto de vista esto se debe a 3 factores principales: el uso de la pseudocoloración para explotar las cualidades de detección las Redes Neuronales Convolucionales, segundo, el uso de la red NASNet como extractor de características y su correcta detección en el área de interés (región mamaria), y finalmente la reducción de características a partir del análisis de componentes principales.

5. Conclusiones

En este trabajo se presentó el diseño de un sistema CADx para la clasificación de lesiones benignas o malignas en imágenes de mamografía, en el cual se emplea el método de Pseudocolor para complementar la información perceptual tanto para el especialista como del método de Extracción de características. Al evaluar el método propuesto, se obtuvieron aproximaciones a los resultados publicados en [32, 23], es por ello que se debe mejorar la etapa de selección de características.

Como a trabajo a futuro se diseñará un método para la clasificación de estas lesiones, a partir de una Red Neuronal Convolucional desde entrenada desde 0 la cual, nos permita determinar la etapa BI-RADS perteneciente de la lesión. Además, se implementara un método que sea capaz de realizar la segmentación automática de la ROI, con técnicas de tipo artesanales o técnicas basadas en aprendizaje profundo.

Referencias

1. Abidi, B. R., Zheng, Y., Gribok, A. V., Abidi, M. A.: Improving weapon detection in single energy X-ray images through pseudocoloring. In: Proceedings of the IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 36, no. 6, pp. 784–796 (2006)
2. Aibar, L., Santalla, A., Criado, M. L., González–Pérez, I., Calderón, M., Gallo, J., Parra, J. : Clasificación radiológica y manejo de las lesiones mamarias. Clínica e Investigación en Ginecología y Obstetricia, vol. 38, no. 4, pp. 141–149 (2011) doi:10.1016/j.gine.2010.10.016
3. Albawi, S., Mohammed, T. A., Al-Zawi, S.: Understanding of a convolutional neural network. In: International Conference on Engineering and Technology (ICET), pp. 1–6 (2017) doi: 10.1109/ICEngTechnol.2017.8308186
4. American Cáncer Society: Cuando se comunican con usted después del mamograma (2021)

5. Arora, R., Rai, P. K., Raman, B.: Deep feature–based automatic classification of mammograms. International Federation for Medical and Biological Engineering and Computing, vol. 58, no. 6, pp. 1199–1211 (2020) doi: 10.1007/s1 1517-020-02150-8
6. Chattopadhay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N.: Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In: Proceedings of IEEE Winter Conference on Applications of Computer Vision (2018) doi:10.1109/wacv.2018.00097
7. Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., Tarbox, L., Prior, F.:, The Cancer Imaging Archive (TCIA): maintaining and operating a public information repository. Journal of Digital Imaging, vol. 26, no. 6, pp. 1045–1057 (2013) doi: 10.1007/s10278-013-9622-7
8. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning, vol. 20, no. 3, pp. 273–297 (1995) doi: 10.1007/BF00994018
9. Dale la mano a la prevención del cáncer de mama.
10. De Oliveira, J. E., Machado, A. M., Chavez, G. C., Lopes, A. P., Deserno, T. M., Araújo, A. de A.: MammoSys: A content-based image retrieval system using breast density patterns. Computer Methods and Programs in Biomedicine (2010) doi: 10.1016/j.cmpb.2010.01.005
11. Haindl, M., Remeš, V.: Pseudocolor enhancement of mammogram texture abnormalities. Machine Vision and Applications, vol. 30, pp. 785–794 (2019) doi: 10.1007/s00138-019-01028-6
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Computing Research Repository (2015) doi: 10.48550/arXiv.1512.03385
13. INEGI: Estadísticas a propósito del día mundial de la lucha contra el cáncer de mama (2020)
14. Instituto Nacional de Cancerología (INCan), Secretaria de Salud Gobierno de México: ¿Qué es el cáncer?
15. Ippolito, P.: Feature extraction technics. Towardsdatascience (2019)
16. Krizhevsky, A., Sutskever, I., Hinton, G. E.: ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, vol. 25, no. 2 (2012) doi: 10.1145/3065386
17. Lévy, D., Jain, A.: Breast mass classification from mammograms using deep convolutional neural networks. Computing Research Repository (2016)
18. Mery, D., Saavedra, D., Prasad, M.: X-ray baggage inspection with computer vision: A survey. IEEE Access, vol. 8, pp. 145620–145633 (2020) doi: 10.1109/ access.2020.3015014
19. Mishra, A.: Metrics to evaluate your machine learning algorithm (2018)
20. National Cáncer Institute, USA.gov, ¿Qué es el cáncer?
21. Pan, S. J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359 (2010) doi: 10.11 09/tkde.2009.191
22. Ragab, D. A., Sharkas, M., Marshall, S., Ren, J.: Breast cancer detection using deep convolutional neural networks and support vector machines. PeerJ Hubs (2019) doi: 10.7717/peerj.6201
23. Ragab, D. A., Sharkas, M., Marshall, S., Ren, J.: Breast cancer detection using deep convolutional neural networks and support vector machines. PeerJ Hubs (2019) doi:10.7717/peerj.6201
24. Rodriguez, J. D., Perez, A., Lozano, J. A.: Sensitivity analysis of k-fold cross validation in prediction error estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 3, pp. 569–575 (2010) doi: 10.1109/TPAMI.2009.187
25. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252 (2015) doi: 10.48550/ARXIV.1409.0575

26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Computer Vision and Pattern Recognition (2014)
27. Spak, D. A., Plaxco, J. S., Santiago, L., Dryden, M. J., Dogan, B. E.: BI-RADS ® fifth edition: A summary of changes. pp. 135 (2017) doi:10.1016/j.diii.2017.01.001
28. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, Inception-ResNet and the impact of residual connections on learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1 (2016) doi: 10.48550/arXiv.1602.07261
29. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2014) doi:10.48550/arXiv.1409.4842
30. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2818–2826 (2015)
31. The University of Illinois at Chicago, UIC dataset (2020)
32. Tsochatzidis, L., Costaridou, L., Pratikakis, I.: Deep learning for breast cancer diagnosis from Mammograms-A comparative study. Journal of Imaging, vol. 5, no. 3, pp. 37 (2019) doi: 10.3390/jimaging5030037
33. World Health Organization. OMS — Cáncer de mama: Prevención y control. World Health Organization.
34. Yanase, J., Triantaphyllou, E.: A systematic survey of computer-aided diagnosis in medicine: Past and present developments. Expert Systems with Applications, vol. 138 (2019)
35. Yanase, J., Triantaphyllou, E.: A systematic survey of computer-aided diagnosis in medicine: Past and present developments. Expert Systems with Applications, vol. 138, no. 112821 (2019) doi: 10.1016/j.eswa.2019.112821
36. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks?. In: Advances in Neural Information Processing Systems, vol. 27, pp. 3320–3328 (2014) doi: 10.48550/arXiv.1411.1792
37. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V.: Learning Transferable Architectures for Scalable Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8697–8710 (2017) doi: 10.48550/arXiv.1707.07012

ISSN 1870-4069

Reconocimiento automático de marcha antiálgica a partir de la medición de la cantidad de actividad empleando el giroscopio de un teléfono inteligente

Juan-Carlos González-Islas[1,2], Omar-Arturo Domínguez-Ramírez[1], Omar López-Ortega[1], René-Daniel Paredes-Bautista[3], David Diaz Girón-Aguilar[2]

[1] Universidad Autónoma del Estado de Hidalgo, México

[2] Universidad Tecnológica de Tulancingo, México

[3] Centro de Rehabilitación Integral de Hidalgo, México

(juan_gonzalez7024,omar,lopezo)@uaeh.edu.mx,
1718110586@utectulancingo.edu.mx,
drrenedanielparedes@gmail.com

Resumen. La estadística de la marcha antiálgica, como consecuencia de diversas enfermedades desarrolladas por el ser humano, sostiene un crecimiento exponencial. Los métodos clásicos de diagnóstico advierten subjetividad en el resultado. El empleo del reconocimiento automático es una alternativa que garantiza certeza y reducción de tiempo en el diagnóstico. En este artículo, se presenta un marco de trabajo para el reconocimiento automático de la marcha para clasificar las marchas antiálgica y no antiálgica, basado en la medición de la cantidad de actividad. Para ello es empleado el giroscopio embebido en un teléfono inteligente. El muestreo, reducción, extracción y selección de características son elementos del marco propuesto que preservan la esencia de los datos brutos. En el estudio comparativo de algoritmos de clasificación para seleccionar el método idóneo para el caso de estudio, se evalúan los algoritmos de: i) Análisis Discriminante Lineal (LDA), ii) k-Vecinos más Cercanos (kNN), iii) Máquinas de Soporte Vectorial (SVM), iv) Naive Bayes (NB), y v) Árboles de Decisión (DT). El desempeño de los algoritmos fue determinado empleando las métricas de tasa de clasificación correcta (CCR), sensibilidad (R), especificidad (SP) y precisión. Siendo SVM el mejor con $\mathrm{CCR} = \mathrm{CR} = \mathrm{R} = \mathrm{SP} = \mathrm{P} = 100\,\%$. Los resultados obtenidos permiten determinar la viabilidad de emplear el marco de trabajo para diagnóstico objetivo y soporte de toma decisiones en los tratamientos asociados a la marcha antiálgica en escenarios médicos reales.

Palabras clave: Análisis de la marcha, aprendizaje automático, marcha antiálgica, giroscopio.

Juan-Carlos González-Islas, Omar-Arturo Domínguez-Ramírez, Omar López-Ortega, et al.

Automatic Recognition of Antalgic Gait from the Measurement of the Amount of Activity Using the Gyroscope of a Smartphone

Abstract. The statistics of the antalgic gait, as a consequence of various diseases developed by the human being, sustains an exponential growth. Classic diagnostic methods warn of subjectivity in the result. The use of automatic recognition is an alternative that guarantees certainty and reduction of time in the diagnosis. In this article, an automatic gait recognition framework is presented to classify antalgic and non-antalgic gaits, based on the measurement of the amount of activity. For this, the gyroscope embedded in a smartphone is used. Sampling, reduction, extraction, and feature selection are elements of the proposed framework that preserve the essence of the raw data. In the comparative study of classification algorithms to select the ideal method for the case study, the algorithms of: i) Linear Discriminant Analysis (LDA), ii) k-Nearest Neighbors (kNN), iii) Support Vector Machines (SVM), iv) Naive Bayes (NB), and v) Decision Trees (DT). The performance of the algorithms was determined using the correct classification rate (CCR), sensitivity (R), specificity (SP), and precision metrics. Being SVM the best with $\mathrm{CCR} = \mathrm{CR} = \mathrm{R} = \mathrm{SP} = \mathrm{P} = 100\%$. The results obtained allow us to determine the feasibility of using the framework for objective diagnosis and decision-making support in treatments associated with analgesic gait in real medical scenarios.

Keywords: Gait analysis, machine learning, antalgic gait, gyroscope.

1. Introducción

La marcha humana es la forma más importante de locomoción que el ser humano tiene para desplazarse de manera autónoma usando sus extremidades inferiores [34]. El análisis cuantitativo de este proceso, permite evaluar numéricamente la marcha mediante la medición de los datos clínicos y biomecánicos de ésta. El análisis de la marcha ha sido muy útil en áreas como robótica, biomecánica, deportes, seguridad, rehabilitación y diagnóstico clínico [17, 28, 34].

En términos clínicos, esta herramienta ha sido útil para diagnosticar padecimientos asociados a la marcha antiálgica como: i) osteoartritis de rodilla, ii) artritis reumatoide, iii) derrame cerebral , iv) mal de Parkinson, v) parálisis cerebral, entre otras [16]. La marcha antiálgica es una de las más comunes dentro de las marchas anómalas, la mayoria de alteraciones en huesos, músculos, articulaciones y tejidos blandos derivadas en este tipo de marcha no son tan evidentes, por lo que recientemente ha sido de especial interés el desarrollo de herramientas asistenciales y de diagnóstico para este padecimiento.

Una de sus característica es la presencia de cojera en el patrón de marcha, que provoca que se acorte la fase apoyo con respecto a la de oscilación [3]. Por ello, la necesidad de determinar objetivamente la fase del ciclo de marcha con anomalía y su relación con los elementos del sistema músculo esquelético con mayor deterioro.

Derivado de la gran cantidad de datos y naturaleza multivariable, multidimensional e incertidumbre, el problema de análisis de la marcha se puede resolver desde un enfoque basado en aprendizaje automático que se conoce como reconocimiento automático de la marcha, el cual se centra en la evaluación y comparación de patrones de marcha de diferentes sujetos, que permite distinguir la forma de caminar de cada persona [29].

Por lo que es necesario el desarrollo de algoritmos que identifiquen las formas de caminar de las personas como lo es la marcha antiálgica [30]. La aceleración y velocidad angular de las articulaciones del sistema músculo-esquelético de una persona durante el ciclo de marcha son diferentes respecto a otra. Lo que ha posibilitado que este tipo de datos sean utilizados para identificar individuos.

Comúnmente, los sensores se colocan en las partes de interés a evaluar del cuerpo [23], como la rodilla por ejemplo. Hoy en día, los desarrollos en términos de sensores vestibles y electrónica portable, ha permitido el desarrollo de sistemas sofisticados, con alta precisión, bajo nivel de integración y bajo costo, embebidos en los teléfonos inteligentes [33]. En este trabajo se considera el problema de clasificación de personas con un patrón de marcha antiálgica y marcha no antiálgica.

Para lo cual, se describe el desarrollo de un marco de trabajo de reconocimiento automático para clasificar entre ambos tipos de marcha, basado en aprendizaje automático empleando el giroscopio embenbido de un teléfono inteligente como dispositivo de adquisición de datos. El cual aduce su aplicación en escenarios médicos para el apoyo en el diagnóstico objetivo y soporte de toma de decisiones en tratamientos en padecimientos relacionados a la marcha antiálgica.

1.1. Trabajo relacionado

Recientemente, el reconocimiento automático de la marcha ha sido usado para determinar anomalías en el patrón de marcha [7]. En términos de la marcha antiálgica, se han reportado trabajos para el diagnóstico de: i) osteoartritis de rodilla [19, 35], ii) artritis reumatoide [22], iii) mal de Parkinson [21], iv) derrame cerebral [8, 14] y v) esclerosis múltiple [1].

Hoy en día, existen múltiples plataformas de adquisición de datos para el análisis automático de la marcha, por ejemplo, la instrumentación de la fuerza de interacción entre la fascia plantar y la superficie de caminata, se ha utilizado para determinar parámetros asociados a condiciones normales o anormales de marcha [5, 36].

De igual manera, la visión artificial se ha empleado para este propósito, en [15] se ha reportado la clasificación de la marcha normal y 5 tipos de marcha anómala entre las que se encuentra la antiálgica, basado en una unidad recurrente cerrada (GRU) con una exactitud del 90.13 %. Por su parte, en [20] se reporta hasta el 88.68 % de tasa de reconocimiento usando redes Bayesianas. También se han usado otros algoritmos de clasficación como redes neuronales artificiales (ANN), redes neuronales convolucionales (CNN) y k-vecinos más cercanos (kNN)[30].

Sin embargo, hay factores asociados tanto al sensor de piso como a los sistemas de visión que limitan el rendimiento de la plataforma de caracterización, además de no permitir detectar de manera explícita el nivel de actividad humana como parte complementaria al tipo de padecimiento [16, 29].

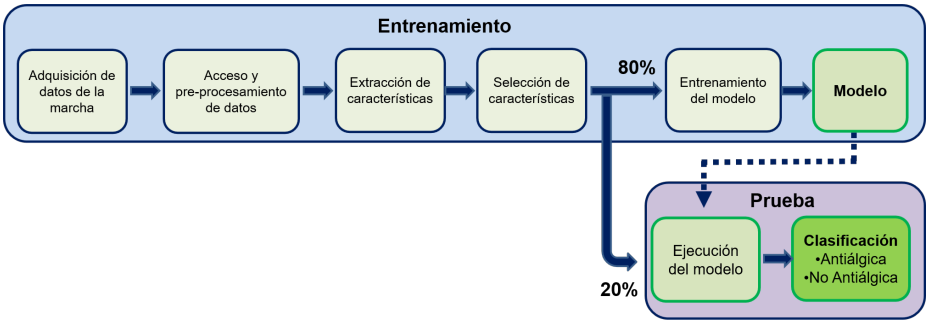


Fig. 1. Marco de trabajo de aprendizaje automático supervisado para la clasificación de marchas antiálgica y no antiálgica.

En este trabajo se reporta el uso de sensores de velocidad angular, como una alternativa para la detección de marcha antiálgica basada en el nivel de actividad humana. Aunado a lo anterior, se han desarrollado otros sistemas para el reconocimiento de la marcha con un rendimiento de 94.4 % usando kNN [27]. Por su parte, Hoang [13] usando el acelerómetro embebido en un teléfono y máquinas de soporte vectorial (SVM) ha reportado una exactitud del 91 %. Gafurov et al., [12] usan acelerómetros para el mismo propósito, obteniendo una tasa de reconocimiento de 83.3 % mediante métodos estadísticos.

Los sensores embebidos en un teléfono inteligente resuelven parcialmente el problema de la instrumentación y acondicionamiento de señales inherente a este tipo de sensores, en [2] se ha empleado este enfoque para reconocimiento de actividades humanas como la caminata con una precisión del 95.6 % implementando SVM y en [9] se reporta el reconocimiento de la marcha con una tasa de reconocimiento de 87.6 % y 86.7 % usando LibSVM y modelo logístico de árboles, respectivamente.

El uso de varias unidades inerciales, aunque aumenta la complejidad en el procesamiento, mejora la capacidad del sistema para reconocer la marcha, en [11] se describe el uso de dos acelerómetros. Mientras que Ngo et al., [23] han generado una de las bases de datos más grandes (OUISIR) usando 4 unidades inerciales para identificar si una persona camina sobre una superficie plana o con pendientes positiva o negativa.

Debido a que la marcha es un proceso periódico, muchos de los trabajos que usan la aceleración y la velocidad angular para reconocerla se basan en la detección del periodo para construir patrones de marcha. También es posible emplear características en el dominio de la frecuencia, como la intensidad del espectro o la transformada de Fourier [18].

De acuerdo a la revisión de las aportaciones en la literatura, y como parte complementaria a lo reportado en [4], se establece el siguiente planteamiento:

Si bien existen muchos trabajos sobre el reconocimiento automático de la marcha, pocos están desarrollados para identificar objetivamente marcha antiálgica y su relación con los elementos del sistema músculo-esquelético, menos aun aprovechando las prestaciones de una unidad inercial embebida en un teléfono inteligente para identificar anomalías con base en la cantidad de actividad humana detectada. Adicionalmente, las

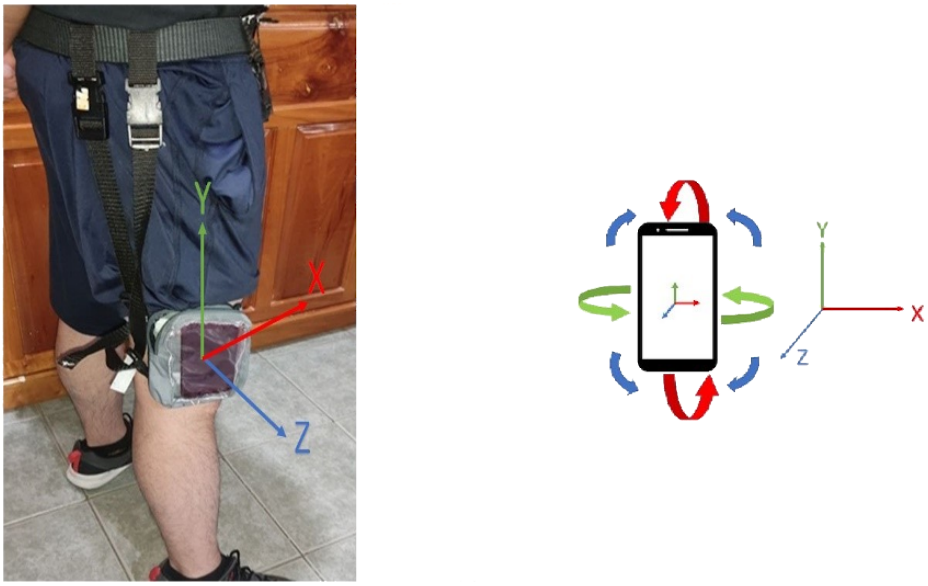


Fig. 2. Disposición del teléfono inteligente y correspondencia entre los ejes anatómicos del sujeto y del giroscopio del teléfono.

condiciones de eficiencia en la clasificación y validación requeridas por especialistas médicos del área, respecto al diagnóstico de padecimientos no han sido resueltas.

Este artículo está organizado de la siguiente manera: Sección 1, se presentan la introducción y las aportaciones en la literatura sobre el reconocimiento automático de marcha antiálgica, empleando el giroscopio de un teléfono inteligente. El marco de trabajo de reconocimiento automático y cada una de sus secciones son descritas en la sección de materiales y métodos. Los resultados experimentales y una discusión sobre los mismos se proporcionan en la sección 3. Finalmente, se presentan las conclusiones y el trabajo futuro.

2. Marco de trabajo de reconocimiento automático de la marcha

Para el desarrollo de la tarea de clasificación de marchas antiálgica y no antiálgica basado en aprendizaje automático empleando un telefono inteligente como dispositivo de adquisición de datos planteada en este trabajo de investigación, se requiere una serie de tareas secuenciales e iterativas que procesan el conjunto de datos, la cual se conoce como el marco de trabajo [32, 33].

En la Figura 1 se ilustra la metodología de dicho marco. El marco de trabajo consiste en 2 fases las cuales son entrenamiento y prueba [19]. Para ambas fases el punto de inicio es la adquisición de datos de la marcha, posteriormente se realizan las etapas de acceso y preprocesamiento de datos, extracción y selección de características.

La quinta etapa de la primera fase consiste en el entrenamiento del modelo de clasificación, para lo cual, derivado de la poca cantidad de secuencias se emplea el 80 %

del conjunto de datos, mientras que el otro 20 % se usa en la fase de prueba. Cuando se tiene un conjunto de datos con muchas instancias usualmente se emplea el 70 % para entrenamiento, 20 % para prueba y 10 % de instancias no conocidas (nuevos ejemplos).

La plataforma de trabajo empleada para implementar este trabajo es el toolbox de aprendizaje automático y estadística de MATLAB®. A continuación cada etapa es descrita a detalle. El conjunto de datos asociados a la velocidad angular adquiridos, consiste en las señales de los 3 ejes del giroscopio modelo bmi160 de BOSCH® (embebido en un teléfono inteligente) con una precisión de 0.0001 rad/s y un intervalo máximo de 34.9 rad/s.

El sistema operativo del celular es android 10.0, memoria RAM de 4.00 GB y procesador de 8 núcleos a 2.32 GHz; el cual fue colocado en la parte lateral de la rodilla derecha de los sujetos de estudio como se muestra en la Figura 2, debido a que en este trabajo es la articulación de interes. Sin embargo, puede ser colocado de manera libre en otra parte del cuerpo. El eje anatómico anterior coincide con el eje x, el eje superior del cuerpo con el eje y, mientras que el eje derecho con el eje z del giroscopio, respectivamente.

Los datos brutos sin procesar, capturados con una frecuencia de muestreo de 406 Hz contienen ventanas variables entre 19 y 12 segundos en promedio de marchas antiálgica y no antiálgica, respectivamente. El experimento realizado fue el de la caminata de 10 metros [26], la cual es una medida de rendimiento utilizada para evaluar la velocidad de caminata en metros por segundo en una distancia corta y que se emplea para diagnosticar padecimientos como los citados en la introducción.

La prueba fue realizada con 30 varones entre 18 y 49 años [34], de las cuales se obtuvieron 18 secuencias de una marcha no antiálgica y 12 de una marcha antiálgica sintética, es decir la marcha antiálgica fue emulada para validar los componentes en un ambiente de laboratorio con base en las especificaciones de un especialista clínico (desarrollo tecnológico de acuerdo a la metodología TRL [24]).

Derivado a que no existe un patrón de marcha diferente para hombres o mujeres, solo cambia la mágnitud de los parámetros evaluados, en este trabajo como primera etapa se usó solo una muestra masculina. El primer paso en cualquier proyecto de aprendizaje automático es el acceso y exploración de los datos mediante la inspección de algunos ejemplos creando visualizaciones [25].

La Figura 3 muestra las señales de los 3 ejes del giroscopio durante una secuencia de la prueba de caminata de 10 metros de una secuencia de marcha antiálgica y no antiálgica entre la ventana de los 3 y 6 segundos. Las tareas de preprocesamiento de datos, para resolver problemas de ruido, incompletez, inconsistencia y cantidad en las bases de datos son: limpieza, integración, reducción, y transformación [6].

En este trabajo no se consideran datos incompletos, sin embargo, si se aplica un filtro digital pasabajas con una frecuencia de corte de 0.5 Hz. En la etapa de integración, los datos de las señales del giroscopio del eje x (Gx), eje y (Gy) y eje z (Gz) de las secuencias de marchas antiálgica y no antiálgica se incorporan en un solo conjunto de datos.

Posteriormente, se realizó una reducción de dimensionalidad de todas las señales originales mediante la extracción de los datos de la ventana entre (3 - 6) segundos y una frecuencia de muestreo de 100 Hz, resultando secuencias de 305 datos con la

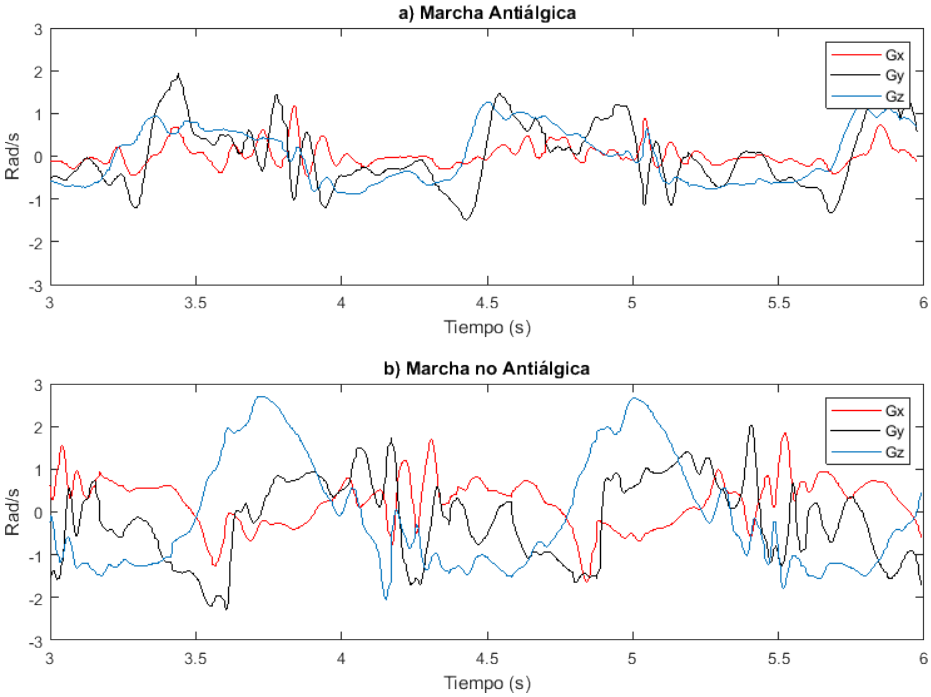


Fig. 3. Visualización de las señales del giroscopio (Gx, Gy, y Gz) sin preprocesar para una secuencia de: a) Marcha antiálgica y b) Marcha no antiálgica.

Tabla 1. Valores promedio del conjunto de datos para cada uno de los atributos de las marchas antiálgica $\overline{A}$ y no antiágica $\overline{NA}$.

$Gx(m)$	$Gy(m)$	$Gz(m)$	$Gx(s)$	$Gy(s)$	$Gz(s)$	$Gx(r)$	$Gy(r)$	$Gz(r)$	$Gx(p)$	$Gy(p)$	$Gz(p)$	**Clase**
0.032	-0.002	-0.009	0.280	0.724	0.670	0.283	0.728	0.672	0.517	-0.480	-0.948	$\overline{A}$
0.119	-0.040	0.038	0.700	1.058	1.280	0.715	1.062	1.285	-0.344	0.320	0.632	$\overline{NA}$

misma representatividad que los originales. En la etapa de extracción de características los datos brutos se convierten en información útil para los algoritmos de clasificación, eliminando redundancia y facilitando la generalización [32].

Existe una gran variedad de características que se pueden extraer tanto en los dominios del tiempo como de la frecuencia. Por simplicidad, en este caso se calculó la media (m), la desviación estandar (s), el valor de la raiz cuadrática media (RMS) (r) y los valores del análisis de componentes principales (PCA) (p) de cada señal Gx, Gy y Gz, respectivamente [33].

El conjunto de datos consiste de 30 instancias con 12 atributos y la clase (antiálgica y no antiálgica) cada una como se ejemplifica en la Tabla 1. Por el tamaño del conjunto de datos, en la tabla solo se presenta el valor promedio de cada uno de los atributos para cada clase, respectivamente.

Los valores obtenidos permiten determinar que ambos tipos de marcha se pueden diferenciar significativamente y a partir de esas métricas clasificarlas. Una vez que se

han extraido las características, se hace una selección de éstas para escoger la menor cantidad, para no tender al sobreajuste y mantener la escencia de los datos [25].

Mediante el método de fuerza bruta empleado sobre las tuplas $\{Gx(m), Gx(s), Gx(r), Gx(p)\}$; $\{Gy(m), Gy(s), Gy(r), Gy(p)\}$ y $\{Gz(m), Gz(s), Gz(r), Gz(p)\}$, se determinó que la media, desviación estandar y el valor RMS representan con mayor precisión el conjunto de datos de cada instancia con su respectiva clase. Para validar dicho proceso se implementó el algoritmo de agrupamiento particional basado en prototipos k-means [10], agrupando el 100 % de los ejemplos en su respectiva clase.

A través de la inclusión de las características de la marcha y las clases, de manera iterativa en la fase de entrenamiento, se generan los modelos de clasificación. Los algoritmos reportados en la literatura más empleados para este propósito y que son implementados en este trabajo son: i) máquinas de soporte vectorial (SVM), ii) análisis discriminante lineal (LDA), iii) árboles de decisión (DT), iv) K-vecinos más cercanos (kNN) y v) Naive-Bayes (NB) [29, 33].

De los 30 sujetos de estudio (instancias), de manera aleatoria y derivado de que son pocas instancias, el 80 % se utilizó para entrenamiento y el 20 % para prueba. Las métricas empleadas para evaluar el desempeño de cada algoritmo de clasificación para n experimentos son: la tasa de clasificación correcta (CCR), sensibilidad o recall (R), especificidad (SP) y precisión (P) [29]. En este trabajo, para ampliar el análisis de desempeño, además de las métricas anteriores se usan la tasa clasificación errónea (ECR), tasa clasificación (CR) y predominio (Pr) [31]. A continuación, se presentan las ecuaciones para calcular las métricas mencionadas.

– Tasa de clasificación correcta (CCR):

$$\overline{\mathrm{CCR}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{VP}_i + \mathrm{VN}_i}{\mathrm{M}_i}, \tag{1}$$

– Tasa clasificación errónea (ECR):

$$\overline{\mathrm{ECR}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{FP}_i + \mathrm{FN}_i}{\mathrm{M}_i}, \tag{2}$$

– Tasa clasificación (CR):

$$\overline{\mathrm{CR}} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{VP}_i + \mathrm{VN}_i + \mathrm{FP}_i + \mathrm{VN}_i), \tag{3}$$

– Sensibilidad o Recall (R):

$$\overline{\mathrm{R}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{VP}_i}{\mathrm{VP}_i + \mathrm{FN}_i}, \tag{4}$$

– Especificidad (SP):

$$\overline{\mathrm{SP}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{VN}_i}{\mathrm{VN}_i + \mathrm{FP}_i}, \tag{5}$$

Tabla 2. Desempeño de los algoritmos empleados para la clasificación de marcha antiálgica y no antiálgica.

Métrica / Algoritmo	**LDA %**	**kNN %**	**SVM %**	**NB %**	**DT %**
CCR	98.33	99.17	100	94.17	87.50
ECR	1.67	00.83	0	5.83	12.50
CR	100	100	100	100	100
R	100	100	100	90	90.50
SP	97.75	98.75	100	96.50	90
P	95.83	98.33	100	94.17	82.50
Pr	35	40	48.33	39.17	33.33

– Precisión (P):

$$\overline{\mathrm{P}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{VP}_i}{\mathrm{VP}_i + \mathrm{FP}_i}, \tag{6}$$

– Predominio (Pr):

$$\overline{\mathrm{CR}} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{VP}_i, \tag{7}$$

donde:

– Verdaderos positivos (VP): Instancias de marcha antiálgica clasificadas correctamente.
– Verdaderos negativos (VN): Instancias de marcha no antiálgica clasificados correctamente.
– Falsos positivos (FP): Instancias clasificadas como antiálgica y no lo son.
– Falsos negativos (FN): Instancias clasificados como no antiálgicos y si lo son.
– N: Número total de instancias del conjunto de prueba.
– M: Número total de instancias de prueba clasificadas.

3. Análisis y discusión de resultados

El rendimiento del marco de trabajo para clasificar entre marchas antiálgica y no antiálgica propuesto en este trabajo, fue evaluado mediante la ejecución de 20 experimentos por cada algoritmo sobre el conjunto de prueba aleatorio. En la Tabla 2 se resume el resultado que define el desempeño de los algoritmos de clasificación, a partir de la evaluación de las métricas (Ecuaciones: 1, 2, 3, 4, 5, 6 y 7). La tasa de clasificación (CR) en todos los casos fue igual a 100 %, lo que implica que todas las instancias fueron clasificadas por todos los algoritmos.

La tasa de clasificación correcta (CCR), se relaciona a la probabilidad de clasificar acertadamente el tipo de marcha dado un nuevo ejemplo, ya sea antiálgica o no antiálgica, en este caso SVM y DT, tienen las tasas más alta (100 %) y más baja (87.50 %), respectivamente. De igual manera que la CCR, tanto en la Tabla 2 como en la Figura 4a se presenta la tasa de clasificación errónea (ECR), la cual se refiere a la probabilidad de clasificar de manera incorrecta un nuevo ejemplo.

La Figura 4b y la Tabla 2 presentan los resultados de la sensibilidad (R) de los algoritmos implementados, la cual se relaciona a la probabilidad de clasificar correctamente una instancia de marcha antiálgica, siendo LDA, kNN y SVM los métodos con la tasa más alta (100 %) y NB la más baja (90 %). Mientras que la especificidad (SP) (Figura 4c), corresponde a la tasa de clasificación de sujetos con marcha no antiálgica.

En ambos casos, dichas métricas representan gran utilidad para la toma de decisiones en el diagnóstico y tratamiento de los padecimientos relacionados a la marcha antiálgica. En ese sentido, SVM presenta la mayor especificidad (100 %) y NB la menor, con un 90 %. La precisión (P) en este caso, implica la probabilidad de determinar a los sujetos con marcha antiálgica sobre todos los sujetos clasificados correctamente, siendo nuevamente SVM el algóritmo con mejor resultado (100 %) y DT el peor (82.5 %) como se presenta en la Tabla 2 y la Figura 4d.

La utilidad médica de la precisión se centra en la determinación de sujetos con marcha antiálgica. Finalmente, el predominio provee información de la tasa de detección de los sujetos con una marcha antiálgica con respecto a todas las instancias, teniendo un desempeño similar de $\approx 55\,\%$ en todos lo casos. Como puede observarse en la Tabla 2 y la Figura 4 el algoritmo de clasificación con mejor desempeño para el problema de clasificación de las marchas antiálgica y no antiálgica tratado en este trabajo es SVM.

4. Conclusiones

Ante la posibilidad de emplear a los sensores embebidos de un teléfono inteligente (giroscopio y acelerómetro), es factible la integración de plataformas de instrumentación biomédica de bajo costo, particularmente para la medición de la actividad humana. El ciclo de marcha bípeda, a pesar de que representa una conducta espacial y no temporal (porcentaje en fase de apoyo y oscilación por zancada), puede ser evaluado no sólo a través de la cadencia o cantidad de zancadas por unidad de tiempo, también a partir del nivel de actividad (asociado a la energía total y rendimiento).

Es aquí en el que la señal del giroscopio tiene sentido en la aplicación (diagnóstico de la marcha antiálgica y exclusión de la no antiálgica). Debido a las características y factor de forma de las señales emitidas por el giroscopio, fue indispensable realizar una poda de datos muestreados (preparación o preprocesamiento basado en técnicas de sampling y reducción) que resultan no ser indispensables, sin pérdida de la integridad de los datos brutos significativos.

A partir del volumen de datos resultante; se realizó la extracción de características basado en RMS, media aritmética, desviación estándar y análisis de componentes principales (PCA), siendo excluida ésta última en la etapa de selección desarrollada a partir del algoritmo de agrupamiento (K-Means).

La evaluación de 5 algoritmos de clasificación (LDA, kNN, SVM, NB y DT), enfocados al estudio y clasificación del tipo de marcha (antiálgica o no antiálgica), resultó satisfactoria con la validación de 20 experimentos, para los cuales, derivado del bajo número de instancias, se ocupó el 80 % en la etapa de entrenamiento y definición del modelo, y el 20 % restante para la etapa de prueba.

Para ello, fue propuesta una generalización de las métricas basado en la media aritmética de la evaluación sobre los sujetos de prueba ($\overline{\mathrm{CCR}}$, $\overline{\mathrm{ECR}}$, $\overline{\mathrm{CR}}$, $\overline{\mathrm{R}}$, $\overline{\mathrm{SP}}$, $\overline{\mathrm{P}}$

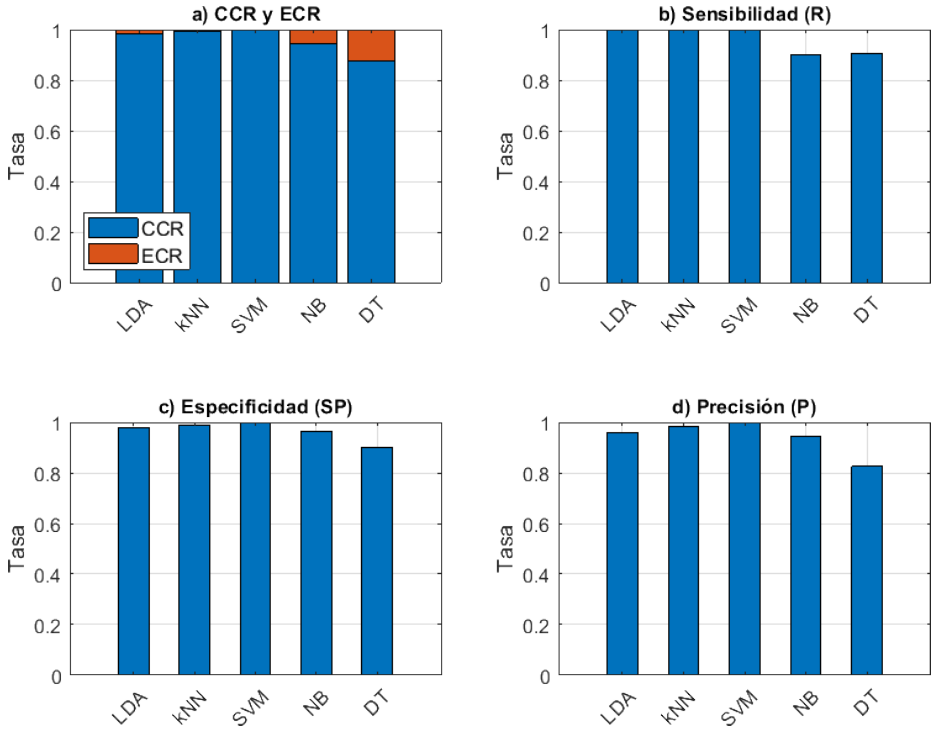


Fig. 4. Métricas (CCR, ECR, R, SP, P) de los algoritmos de clasificación implementados.

y $\overline{\text{Pr}}$), que dependen de manera explícita de los elementos de la matriz de confusión generada por cada algoritmo de clasificación. Como se mencionó anteriormente, la tabla de resultados y las gráficas de desempeño a partir del estadístico de las métricas de evaluación, fue evidente que el algoritmo SVM (Máquinas de Soporte Vectorial) resulta idóneo en la clasificación de marcha antiálgica y no antiálgica.

Derivado del desempeño obtenido en la clasificación es posible extender el marco de trabajo para el diagnóstico de otras enfermedades reflejadas en la marcha o para otro tipo de padecimientos, que derive en el diagnóstico objetivo y soporte de toma de decisiones en tratamiento de la enfermedades de estudio. Además, es posible mejorar el desempeño del marco de trabajo en la etapa de clasificación incrementando el número de instancias tanto en la etapa de entrenamiento como en la etapa de prueba, así como la integración de más datos clínicos y biomédicos que robustezcan el diagnóstico.

Referencias

1. Alaqtash, M., Sarkodie-Gyan, T., Yu, H., Fuentes, O., Brower, R., Abdelgawad, A.: Automatic classification of pathological gait patterns using ground reaction forces and machine learning algorithms. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 453–457 (2011)

2. Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J. L.: Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: Proceedings of the International workshop on ambient assisted living, Springer, pp. 216–223 (2012)
3. Auerbach, N., Tadi, P.: Antalgic gait in adults. StatPearls, (2020)
4. Brahim, A., Jennane, R., Riad, R., Janvier, T., Khedher, L., Toumi, H., Lespessailles, E.: A decision support tool for early detection of knee osteoarthritis using x-ray imaging and machine learning: Data from the osteoarthritis initiative. Computerized Medical Imaging and Graphics, vol. 73, pp. 11–18 (2019), doi: 10.1016/j.compmedimag.2019.01.007
5. Brenton-Rule, A., Mattock, J., Carroll, M., Dalbeth, N., Bassett, S., Menz, H. B., Rome, K.: Reliability of the tekscan matscan® system for the measurement of postural stability in older people with rheumatoid arthritis. Journal of Foot and Ankle Research, vol. 5, no. 1, pp. 21 (2012), doi: 10.1186/1757-1146-5-21
6. Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., Matera, M.: Series in data management systems: Designing data-intensive Web applications. Morgan Kaufmann (2003), doi: 10.1016/B978-155860843-6/50000-1
7. Connor, P., Ross, A.: Biometric recognition by gait: A survey of modalities and features. Computer Vision and Image Understanding, vol. 167, pp. 1–27 (2018), doi: 10.1016/j.cviu.2018.01.007
8. Cui, C., Bian, G. B., Hou, Z. G., Zhao, J., Su, G., Zhou, H., Peng, L., Wang, W.: Simultaneous recognition and assessment of post-stroke hemiparetic gait by fusing kinematic, kinetic, and electrophysiological data. IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 4, pp. 856–864 (2018), doi: 10.1109/tnsre.2018.2811415
9. Derawi, M., Bours, P.: Gait and activity recognition using commercial phones. Computers and Security, vol. 39, pp. 137–144 (2013), doi: 10.1016/j.cose.2013.07.004
10. Frigui, H.: Clustering: Algorithms and applications. In: Proceedings of the First Workshops on Image Processing Theory, Tools and Applications, IEEE, pp. 1–11 (2008), doi: 10.1109/ipta.2008.4743793
11. Gafurov, D., Helkala, K., Søndrol, T.: Gait recognition using acceleration from mems. In: Proceeding of the First International Conference on Availability, Reliability and Security, IEEE (2006)
12. Gafurov, D., Snekkenes, E., Bours, P.: Gait authentication and identification using wearable accelerometer sensor. In: IEEE workshop on automatic identification advanced technologies, pp. 220–225 (2007), doi: 10.1109/autoid.2007.380623
13. Hoang, T., Nguyen, T., Luong, C., Do, S., Choi, D.: Adaptive cross-device gait recognition using a mobile accelerometer. Journal of Information Processing Systems, vol. 9, no. 2, pp. 333–348 (2013), doi: 10.3745/JIPS.2013.9.2.333
14. Ihlen, E. A. F., Støen, R., Boswell, L., de Regnier, R.-A., Fjørtoft, T., Gaebler-Spira, D., Labori, C., Loennecken, M. C., Msall, M. E., Möinichen, U. I., Peyton, C., Schreiber, M. D., Silberg, I. E., Songstad, N. T., Vågen, R. T., Øberg, G. K., Adde, L.: Machine learning of infant spontaneous movements for the early prediction of cerebral palsy: A multi-site cohort study. Journal of Clinical Medicine, vol. 9, no. 1, pp. 5 (2019), doi: 10.3390/jcm9010005
15. Jun, K., Lee, Y., Lee, S., Lee, D.-W., Kim, M. S.: Pathological gait classification using kinect v2 and gated recurrent neural networks. vol. 8, pp. 139881–139891 (2020), doi: 10.1109/access.2020.3013029
16. Khera, P., Kumar, N.: Role of machine learning in gait analysis: A review. Journal of Medical Engineering and Technology, vol. 44, no. 8, pp. 441–467 (2020), doi: 10.1080/03091902.2020.1822940
17. Kitade, I., Nakajima, H., Takahashi, A., Matsumura, M., Shimada, S., Kokubo, Y., Matsumine, A.: Kinematic, kinetic, and musculoskeletal modeling analysis of gait in patients with cervical myelopathy using a severity classification. The Spine Journal, vol. 20, no. 7, pp. 1096–1105 (2020), doi: 10.1016/j.spinee.2020.01.014

18. Kobayashi, T., Hasida, K., Otsu, N.: Rotation invariant feature extraction from 3-d acceleration signals. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3684–3687 (2011), doi: 10.1109/icassp.2011.5947150
19. Kokkotis, C., Moustakidis, S., Papageorgiou, E., Giakas, G., Tsaopoulos, D.: Machine learning in knee osteoarthritis: A review. Osteoarthritis and Cartilage Open, vol. 2, no. 3 (2020)
20. Kozlow, P., Abid, N., Yanushkevich, S.: Gait type analysis using dynamic bayesian networks. Sensors, vol. 18, no. 10 (2018), doi: 10.3390/s18103329
21. Leightley, D., McPhee, J. S., Yap, M. H.: Automated analysis and quantification of human mobility using a depth sensor. IEEE Journal of Biomedical and Health Informatics, vol. 21, no. 4, pp. 939–948 (2016)
22. Nair, S. S., French, R. M., Laroche, D., Thomas, E.: The application of machine learning algorithms to the analysis of electromyographic patterns from arthritic patients. IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 18, no. 2, pp. 174–184 (2009), doi: 10.1109/tnsre.2009.2032638
23. Ngo, T. T., Makihara, Y., Nagahara, H., Mukaigawa, Y., Yagi, Y.: The largest inertial sensor-based gait database and performance evaluation of gait-based personal authentication. Pattern Recognition, vol. 47, no. 1, pp. 228–237 (2014), doi: 10.1016/j.patcog.2013.06.028
24. Olechowski, A. L., Eppinger, S. D., Joglekar, N., Tomaschek, K.: Technology readiness levels: Shortcomings and improvement opportunities. Systems Engineering, vol. 23, no. 4, pp. 395–408 (2020), doi: 10.1002/sys.21533
25. Paluszek, M., Thomas, S.: MATLAB machine learning. Apress (2016)
26. Physiopedia: 10 metre walk test (2022)
27. Rong, L., Jianzhong, Z., Ming, L., Xiangfeng, H.: A wearable acceleration sensor system for gait recognition. In: Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications, pp. 2654–2659 (2007), doi: 10.1109/iciea.2007.4318894
28. Sharif Bidabadi, S., Tan, T., Murray, I., Lee, G.: Tracking foot drop recovery following lumbar-spine surgery, applying multiclass gait classification using machine learning techniques. Sensors, vol. 19, no. 11, pp. 2542 (2019), doi: 10.3390/s19112542
29. Singh, J. P., Jain, S., Arora, S., Singh, U. P.: Vision-based gait recognition: A survey. Institute of Electrical and Electronics Engineers Access, vol. 6, pp. 70497–70527 (2018)
30. Sithi Shameem, F., Wahida, B.: Abnormal walk identification for systems using gait patterns. pp. 112–117 (2016)
31. The MathWorks, Inc: Classperformance properties (2021)
32. The MathWorks, Inc: Heart sound classifier (2021)
33. Wan, C., Wang, L., Phoha, V. V.: A survey on gait recognition. Association for Computing Machinery Computing Surveys, vol. 51, no. 5, pp. 1–35 (2019), doi: 10.1145/3230633
34. Whittle, M. W.: Gait analysis: An introduction. Butterworth-Heinemann (2014)
35. Yoo, T. K., Kim, S. K., Choi, S. B., Kim, D. Y., Kim, D. W.: Interpretation of movement during stair ascent for predicting severity and prognosis of knee osteoarthritis in elderly women using support vector machine. In: Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 192–196 (2013), doi: 10.1109/embc.2013.6609470
36. Zheng, S., Huang, K., Tan, T.: Evaluation framework on translation-invariant representation for cumulative foot pressure image. In: Proceedings of the 18th IEEE International Conference on Image Processing, pp. 201–204 (2011), doi: 10.1109/icip.2011.6115874

ISSN 1870-4069

Traductor automático neuronal ayuuk-español

Delfino Zacarías Márquez[1], Iván Vladimir Meza Ruiz[2]

[1] Universidad Nacional Autónoma de México,
Facultad de Estudios Superiores Acatlán,
México

[2] Universidad Nacional Autónoma de México,
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
México

delfino.zacarias@comunidad.unam.mx,
ivanvladimir@turing.iimas.unam.mx

Resumen. Este artículo presenta el primer sistema de traducción automática neuronal para la lengua ayuuk. En nuestros experimentos traducimos de ayuuk al español y de español a ayuuk. La lengua ayuuk es hablada en el estado de Oaxaca en México por los Ayuukjä'äy (en español comúnmente referidos como Mixes). Se usan diferentes fuentes escritas para crear en un corpus paralelo, esencial para hacer traducción automática, de más de $6,000$ frases que se considera como de bajos recursos. Para algunos de estas fuentes usamos la metodología de la alineación automática. El sistema propuesto se basa en la arquitectura neuronal Transformer y utiliza la tokenización a nivel de subpalabras como entrada. Mostramos el desempeño actual dado los recursos que hemos recolectado para la variante del municipio de San Juan Güichicovi, los resultados son prometedores, hasta 7 en BLEU. Cabe destacar que nuestro desarrollo parte del proyecto Masakhane para lenguas africanas.

Palabras clave: Lengua ayuuk, corpus, traductor automático, subpalabras, transformers, BLEU.

Ayuuk-Spanish Neural Automatic Translator

Abstract. This article presents the first neural machine translation system for the Ayuuk language. In our experiments we translate from Ayuuk to Spanish and from Spanish to Ayuuk. The Ayuuk language is spoken in the state of Oaxaca in Mexico by the Ayuukjä'äy (in Spanish commonly referred to as Mixes). Different written sources are used to create a parallel corpus, essential for automatic translation, of more than $6,000$ phrases that are considered low-resource. For some of these fonts we use the automatic alignment methodology. The proposed system is based on the Transformer neural architecture and uses tokenization at the subword level as input. We show the current performance given the resources we have collected for the variant of the municipality of San Juan Güichicovi, the results are promising, up to 7 in BLEU. It should be noted that our development is part of the Masakhane project for African languages.

Keywords: Ayuuk language, corpus, automatic translator, subwords, transformers, BLEU.

1. Introducción

En los últimos años se han incrementado los esfuerzos para preservar y promover la creación de herramientas de PLN para las lenguas indígenas de las Américas, en particular abordando los desafíos que este esfuerzo requiere [7]. La traducción automática (MT) se ha convertido en uno de los principales metas a perseguir, ya que a largo plazo puede ofrecer beneficios a las comunidades que hablan dichas lenguas.

Por ejemplo, MT podría brindar acceso al conocimiento en alguna lengua nativa y podría facilitar el acceso a servicios como asistencia legal, médica y financiera. En este trabajo trabajamos con la lengua *ayuuk* para la variante del municipio de San Juan Güichicovi, principalmente porque uno de los autores es un hablante nativo de esta variante.

Hasta donde sabemos, no ha habido una construcción de tal sistema para el ayuuk aunque existen recursos para otras variantes[3] por ejemplo en el corpus JW300 [1].Este trabajo se basó en múltiples esfuerzos previos. En el núcleo de nuestra propuesta, seguimos los pasos del proyecto Masakhane[4] que se centra en las lenguas africanas [9]. También contamos con las siguientes bibliotecas:

– Para la alineación automática de nuestros recursos utilizamos el alineador YASA[5] [5].

– Para la tokenización usamos la biblioteca subword-nmt[6] [12].

– Para el entrenamiento de nuestros modelos utilizamos JoeyNMT[7] [4].

Con estas herramientas desarrollamos nuestro código base que se puede consultar en línea junto con la parte del corpus que está disponible con licencia libre[8].

2. Ayuuk de San Juan Güichicovi

Ayuukjä'äy se puede traducir como gente de la lengua de las montañas, la mayoría de los herederos de esta cultura se concentra en 24 municipios del estado de Oaxaca. Ellos son hablantes nativos de la lengua ayuuk aproximadamente $139,760$ hablantes en México. La lengua ayuuk pertenece a la familia lingüística mixe-zoqueana.

Esta familia lingüística está compuesta por las subfamilias Mixe y Zoque[9]. En particular, la subfamilia Mixe incluye las lenguas Mixe de Oaxaca, Sayula Popoluca y Oluta Popoluca. Para el ayuuk hay seis variantes principales de la lengua, entre ellas el Mixe bajo a la que pertenece la variante de San Juan Güichicovi, con un código ISO 639-3 mir.

[3] Coatlán Mixe (ISO 639-3 mco), ayuuk de la región de Coatlán.

[4] https://www.masakhane.io/ (Última visita en marzo de 2021)

[5] https://github.com/anoidgit/yasa (Última visita en marzo de 2021)

[6] https://github.com/rsennrich/subword-nmt (Última visita en marzo de 2021).

[7] https://github.com/joeynmt/joeynmt (Última visita en marzo de 2021)

[8] https://github.com/DelfinoAyuuk/corpora_ayuuk-spanish_nmt

[9] Para obtener más información, visite acerca de la familia mixe-zoqueana https://glottolog.org/resourc/languoid/id/mixe1284

Tabla 1. Fuente de datos recopilados.

Recursos	**es**	**mir**
La Biblia	Abierto	No abierto
Cantos y poemas	No abierto	No abierto
Constitución Política de los Estados Unidos Mexicanos	Abierto	No abierto
Colección personal de Albino Pedro Juan	No abierto	No abierto
Fabulas de Esopo	Abierto	No abierto
Archivo Nacional de lenguas indígenas[10]	No abierto	Abierto
Redes sociales[11]	Abierto	Abierto
The dragon and the rabbit[11]	Abierto	Abierto[11]
Frases traducidos por el autor[11]	Abierto[12]	Abierto

En este municipio se puede estimar que hay aproximadamente $18,298$ hablantes ayuuk. Es importante notar que se estima que solo $3,205$ son monolinguës. La variante ayuuk de San Juan Güichicovi no tiene una ortografía normalizada, hay esfuerzos para acordar las convenciones ortográficas, sin embargo, hay posiciones encontradas referente al número de consonantes.

Una de estas posiciones, se conoce como "bodegeros" que propone 20 consonantes (ver 1b.a) [2] y otra se conoce como "petakeros" que propone una reducción a 14 (ver 1b.b) [10]. En términos de vocales, la variante de San Juan Güichicovi tiene seis (ver 2) que contrastan con las otras variantes del ayuuk que pueden tener hasta nueve vocales.

(1) a. b ch d ds g j k l m n ñ p r s t ts w x y ’
b. p t k x ts m n w y j l r s ’

(2) a e ë i o u .

Las siguientes frases son ejemplos de *ayuuk* de San Juan Güichicovi, estos fueron tomados de cuentos recogidos y escritos por Albino Pedro Juan, hablante nativo y promotor de la lengua.

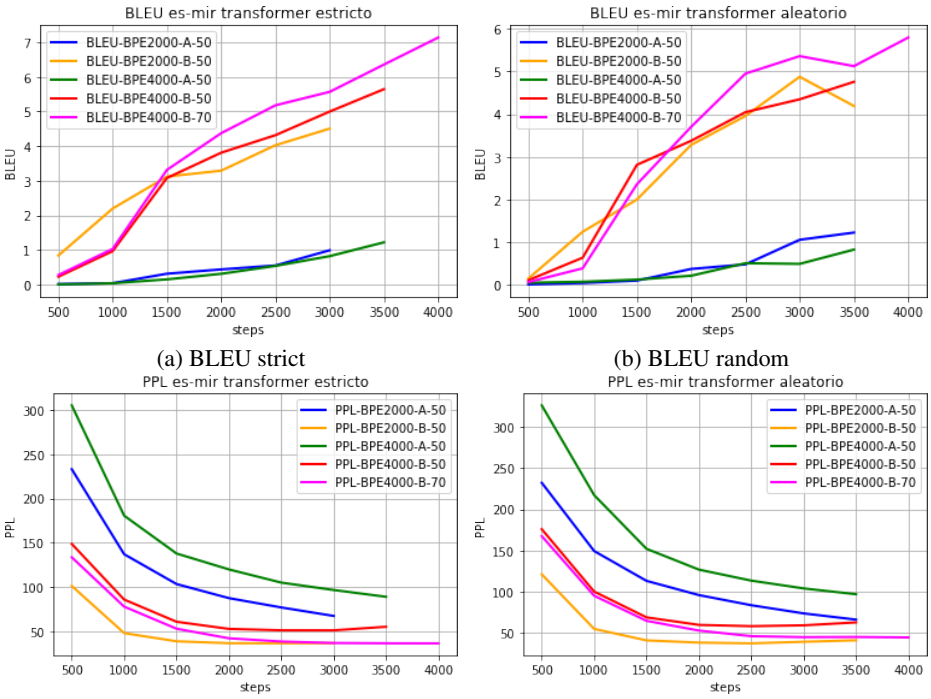
(1) Jantim xyondaak ja koy jadu’un.
El conejo se puso muy feliz.

(2) Kabëk je’e ti y’ok ëjy y’ok nójnë.
Cuando todo se quedó en silencio.

2.1. Español

En el caso del español, nuestro sistema produce traducciones al español mexicano que pertenece a la variante del español de América[13], identificamos la lengua por el código *es* del ISO-639-1.

[10] https://github.com/DelfinoAyuuk/corpora_ayuuk-spanish_nmt
[11] https://mexico.sil.org/es/resources/archives/55868
[12] https://www.manythings.org/anki/
[13] https://glottolog.org/resource/languoid/id/amer1254 (última visita en marzo de 2021)



(a) BLEU strict (b) BLEU random

(c) Perplexity strict (d) Perplexity random

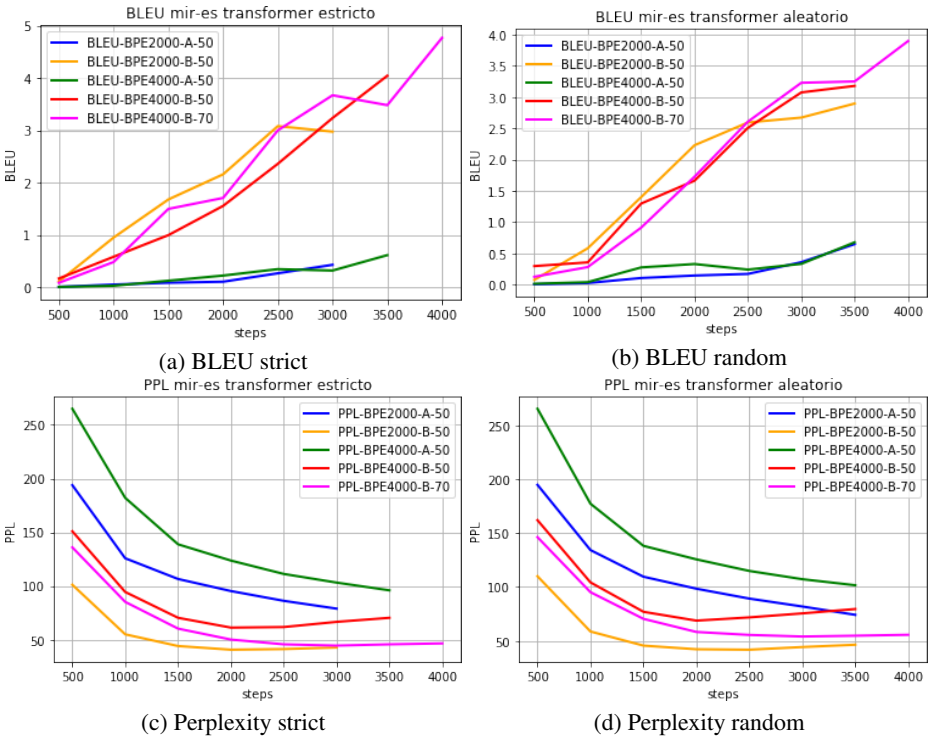
Fig. 1. Perplejidad y BLEU del entrenamiento con dirección es-mir en el conjunto de desarrollo.

3. Sobre el corpus paralelo

Para la creación del corpus paralelo, recolectamos textos de diferentes fuentes para las cuales había una traducción disponible entre ayuuk y español, ver Tabla 1. Dado que tenemos una diversidad de fuentes lingüísticas fue necesario normalizar la ortografía y algunas palabras. Para ello seguimos la propuesta derivada de la investigación de [11] quien ha seguido la unificación de la lengua ayuuk evitando tomar partido en la polémica sobre el número de consonantes. Principalmente hicimos dos reemplazos: ñ/ny y ch/tsy. Algunas de las obras ya estaban alineadas, otras no.

Para aquellos que no se estaban alineadas, creamos alineaciones automáticas usando la herramienta YASA [5]. Descartamos todas las alineaciones vacías y dobles. Finalmente, dividimos aleatoriamente las oraciones en conjuntos de entrenamiento, desarrollo y prueba. Para nuestro experimento, creamos dos versiones divididas, una estricta y otra aleatoria.

En la versión estricta usamos todas las frases del Archivo Nacional de lenguas indígenas [6] como test. Dado que estas oraciones están motivadas lingüísticamente y tienen como objetivo mostrar aspectos lingüísticos de la lengua, tienden a ser más difíciles de traducir. Esta división resultó en $5,847$ / 700 / 912 (train/dev/test). En la división aleatoria muestreamos oraciones al azar de nuestras fuentes, la división final resultó en $5,941$ / 700 / 912 (train/dev/test).



(a) BLEU strict (b) BLEU random

(c) Perplexity strict (d) Perplexity random

Fig. 2. Perplejidad y BLUE del entrenamiento con dirección mir-es en el conjunto de desarrollo.

Observe que la cantidad de frases entre las dos versiones cambia, esto se debe a que después de separar las frases de prueba (i.e., test) eliminamos frases repetidas o similares para los conjuntos train/dev.

Nuestra intuición era tener un entrenamiento/validación más uniforme, frases únicas, para la división aleatoria mientras que los ejemplos de test siguieran la distribución de las fuentes originales. Se siguió la misma metodología para la creación de la versión estricta.

4. Arquitectura neuronal

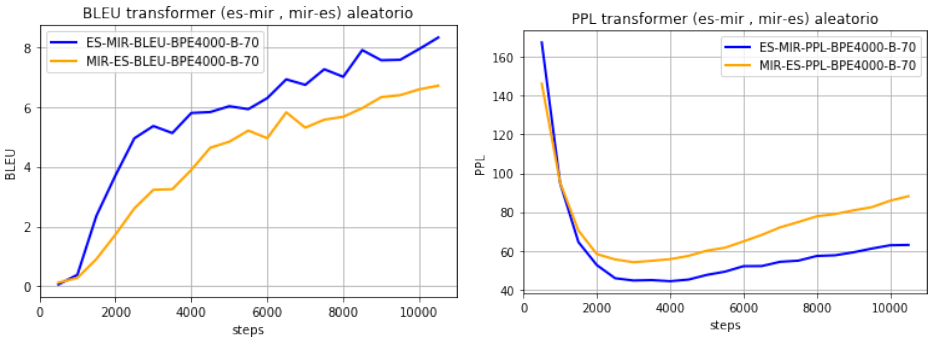
Nuestro modelo de traducción se basa en la arquitectura Transformer [13]. Usamos una configuración codificador-decodificador. Para nuestros experimentos tenemos dos configuraciones para el codificador y decodificador:

a) Número de capas: 3, número de cabezales: 4, dimensión de embedding de entrada: 64, dimensión embedding: 64, tamaño de lote: 128.

b) Número de capas: 6, número de cabezales: 4, dimensión de embedding de entrada: 256, dimensión de embedding: 256, tamaño de lote: 128.

Tabla 2. Puntuaciones BLEU de los entrenamientos con dirección es-mir y mir-es.

Configuracion A - 100 épocas	**Estricto es-mir**		**Aleatorio es-mir**		**Estricto mir-es**		**Aleatorio mir-es**	
BLEU	**dev**	**test**	**dev**	**test**	**dev**	**test**	**dev**	**test**
Longitud máxima 50 BPE 2000	1.72	0.05	1.66	1.71	0.64	0.10	0.91	0.66
Longitud máxima 50 BPE 4000	2.03	0.10	1.21	1.24	1.02	0.16	0.93	0.83
Configuración B - 100 épocas	**Estricto es-mir**		**Aleatorio es-mir**		**Estricto mir-es**		**Aleatorio mir-es**	
BLEU	**dev**	**test**	**dev**	**test**	**dev**	**test**	**dev**	**test**
Longitud máxima 50 BPE 2000	3.91	0.10	3.59	3.70	2.21	0.41	2.49	2.72
Longitud máxima 50 BPE 4000	5.02	0.13	4.17	4.20	2.33	0.28	2.13	2.23
Longitud máxima 70 BPE 4000	7.58	0.10	5.83	5.56	4.03	0.27	3.64	3.52
Configuración B - 250 épocas	**Aleatorio es-mir**				**Aleatorio mir-es**			
BLEU	**dev**		**test**		**dev**		**test**	
Longitud máxima 70 BPE 4000	5.83		5.56		3.64		3.52	



(a) BLEU strict

(b) BLEU random

Fig. 3. Perplejidad y BLEU del entrenamiento es-mir y mir-es con 250 epocas.

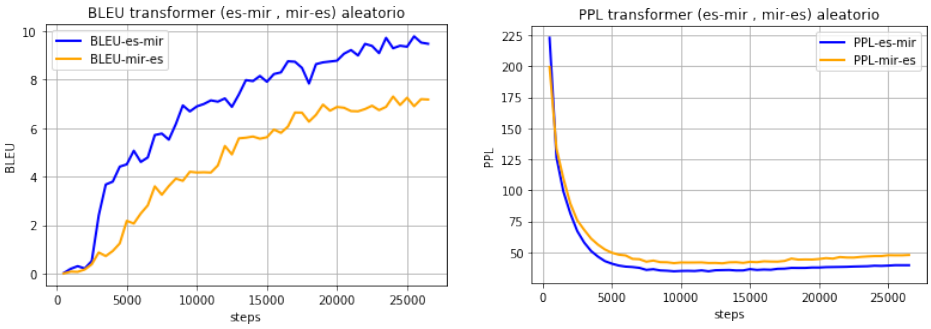
Estos modelos se entrenaron en un servidor con dos GPU Tesla V100. Para obtener el resultado de un modelo usualmente nos tomó alrededor de $2h$ por una cantidad de 100 épocas. También pudimos reproducir los experimentos en la plataforma Colaboratory.

5. Experimentos y resultados

Como se describió en la sección anterior, tenemos dos versiones diferentes de nuestras divisiones, estricto y aleatorio. Por cada división realizamos cinco experimentos, dos para la configuración con menos capas de la red Transformer (A) y tres para la configuración con más capas (B). También modificamos:

a) La longitud máxima de la frase (50 o 70).

b) El vocabulario del algoritmo de subpalabras BPE (probamos 2000 o 4000).



(a) BLEU strict (b) BLEU random

Fig. 4. Perplejidad y BLEU del entrenamiento es-mir y mir-es con 150 épocas.

Tabla 3. Puntajes BLEU de entrenamientos con la configuración con 150 épocas en dirección es-mir y mir-es.

Configuración C - 150 épocas	**Aleatorio es-mir**		**Aleatorio mir-es**	
BLEU	**dev**	**test**	**dev**	**test**
Longitud máxima 100 BPE 4000	7.34	7.29	6.18	5.82

La Figura 1 muestra la perplejidad y la puntuación BLEU en el conjunto de desarrollo durante el entrenamiento para la dirección del español (es) a ayuuk (mir). La primera parte de la Tabla 2, las columnas dos al cinco, presentan los resultados de los conjuntos de desarrollo y prueba. La Figura 2 muestra la curva de aprendizaje en la dirección de traducción ayuuk (mir) al español (*es*). La segunda parte de la Tabla 2, las columnas del seis al nueve, presentan los resultados sobre el desarrollo y la prueba para esta dirección de traducción. Como podemos apreciar, estos conjuntos de experimentos muestran que la traducción es posible.

Tenemos algunas ganancias en el modelo con más capas (B), esto no es trivial ya que tenemos una pequeña cantidad de datos de entrenamiento. Por otro lado, la división estricta como se esperaba muestra que es muy difícil de traducir, las puntuaciones BLEU son mínimas. Sin embargo, con las divisiones aleatorias, las puntuaciones BLEU son más prometedoras. También observamos que en la configuración actual es más "fácil" traducir del español al ayuuk que en la otra dirección. Dado los resultados prometedores de la configuración *B* en la división aleatoria, se realizó un experimento más grande con 250 épocas, siguiendo la intuición de que aún no se ha alcanzado el rendimiento correcto con 100 épocas.

En la Figura 3 se muestra la curva de aprendizaje en el conjunto de desarrollo del entrenamiento en ambas direcciones, la parte inferior de la Tabla 2 muestra los resultados finales. De acuerdo a los resultados, el entrenamiento con 250 épocas tiene el mismo puntaje BLEU que con 100 épocas, lo que destaca es una elevación de perplejidad en el step 4,000. Finalmente, realizamos un experimento con 150 épocas en ambas direcciones con la división aleatoria modificando la configuración *B*, donde reduce a 32 el número de lotes, se mantiene la BPE en $4,000$ y se aumenta a 100 la longitud máxima de frases.

Tabla 4. Traducciones candidatos generados por el traductor automático neuronal.

Traducción español - ayuuk variante de San Juan Güichicovi	
Origen	todos decían que soy malo
Objetivo	age nëm ajxy myana'any ko ëëtsy n'ëxëëgya'ayë
Candidato	a nëje'e ajxy ënajty myënaambë te'emjyëdu'un kyë'ëxë'ëky
Origen	ustedes me vieron ayer en el mercado
Objetivo	mijts axëëy xyijx jim ma too'ktaaktën
Candidato	xyijxëtsy mijts axëëy ma too'ktaaktën
Origen	le dijeron señor, ven y ve
Objetivo	nëm ajxy nyëmaay mëj windsën, jam ukte'emy'ix
Candidato	nëm ajxy y'adsooy mëj windsën, weenëtsy n'ijxë'ëky
Traduccion ayuuk variante de San Juan Güichicovi - español	
Origen	nëm ajxy nyëmaay mëj windsën, jam ukte'emy'ix
Objetivo	le dijeron señor, ven y ve
Candidato	y ellos le dijeron señor,
Origen	jim jaa koy y'aame'naay
Objetivo	el conejo estaba escondido
Candidato	estaba el conejo
Origen	nëm ja ya'ay ajxy y'adsooy ku ëdaa ya'eay ëxyëp kya'aku'uwandëyjëya'ayë, kap ëjts miitsy ëxyëp të nyajkë'ëdëgë'ëy
Objetivo	respondieron y le dijeron si éste no fuera malhechor, no te lo habríamos entregado
Candidato	ellos le respondieron si fuere necesario que no hagas

Dando como resultado final una disminución de perplejidad y un aumento en el puntaje BLEU, tanto para la dirección del español (es) al ayuuk (mir) como del sentido contrario, en la Figura 4 y en la Tabla 3 se muestran los resultados. Los puntajes BLEU y perplejidad dan una idea de cómo pueden ser las traducciones candidatas que proporciona el mejor modelo de traducción generado hasta el momento.

En la Tabla 4 se muestran algunos resultados de traducción que generan los modelos que han sido entrenados en este trabajo. Las frases de entrada se escogieron de manera aleatoria dentro del corpus test. Asimismo, en el tabla de resultados la fila origen corresponde a la lengua que se quiere traducir, la fila objetivo contiene la traducción correcta y la fila candidato es la traducción generada por el traductor automático neuronal.

6. Conclusiones y trabajos futuros

Las experiencias anteriores en MT basadas en la arquitectura de aprendizaje profundo, particularmente en la configuración de seq2seq, para las lenguas nativas de las Américas no habían sido prometedoras [8]. En particular, porque hay pocos o ningún dato de entrenamiento.

Sin embargo, nuestro trabajo muestra que un modelo estándar basado en la arquitectura Transformer y con una configuración de recursos extremadamente baja puede producir algunos resultados prometedores. Todavía son bajos para los estándares normales del campo de MT[14], sin embargo, son prometedores para un futuro donde hay más datos. Para mejorar el rendimiento del sistema, el trabajo futuro se centrará en:

1. Recopilar más datos, especialmente teniendo en cuenta las diferentes variantes del la lengua ayuuk. Hasta ahora en este trabajo abordamos una variante específica, pero existen múltiples variantes que también carecen de una ortografía estandarizada.
2. Aunque la configuración estricta penaliza fuertemente al sistema, creemos que las frases motivadas lingüísticamente podrían establecer una buena referencia para evaluar el progreso y el rendimiento de nuestro sistema de traducción automática. En esta dirección, seguiremos evaluando bajo esta configuración.
3. En este momento nos basamos en subpalabras, sin embargo, nuestro enfoque podría beneficiarse de un análisis morfológico más profundo [3].
4. Nuestra normalización seguirá respetando las posiciones de los "petakeros" y "bodegeros", y para otras variantes también incorporamos posiciones en cuanto al número de vocales.

Agradecimientos. Agradecemos a CONACYT por los recursos proporcionados a través de la Plataforma de Aprendizaje Profundo del Laboratorio de Supercomputación del INAOE para Tecnologías del Lenguaje. También agradecemos el proyecto "Traducción automática para lenguas indígenas de México" PAPIIT-IA104420, UNAM.

Referencias

1. Agić, Ž., Vulić, I.: JW300: A wide-coverage parallel corpus for low-resource languages. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 3204–3210 (2019) doi: 10.18653/v1/P19-1310
2. Graham, S., Hillman, V., Williams, J., Willett, T. L., Becerra-Bautista, M., Pérez-Luría, M., Eberle-Cruz, V., Araiza-Riquer, K., Dieterman, J., McCarty-Jr, J. M., Castañón-López, V., Castañón-Eugenio, M. D.: Breve diccionario del mixe del Istmo Mogoñé Viejo, Oaxaca. Instituto Linguistico de Verano (2018)
3. Kann, K., Mager Hois, J. M., Meza-Ruiz, I. V., Schütze, H.: Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, pp. 47–57 (2018) doi: 10.18653/v1/N18-1005
4. Kreutzer, J., Bastings, J., Riezler, S.: Joey NMT: A minimalist NMT toolkit for novices. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language

[14] Puntuaciones BLEU de entrenamientos de lenguas africanas https://github.com/masakhane-io/masakhane-mt/tree/master/benchmarks

Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations, Association for Computational Linguistics, pp. 109–114 (2019) doi: 10.18653/v1/D19-3019
5. Lamraoui, F., Langlais, P.: Yet another fast, robust and open source sentence aligner. Time to reconsider sentence alignment, XIV Machine Translation Summit, pp. 77–84 (2013)
6. Lyon, D. D.: Mixe de Tlahuitoltepec, Oaxaca, Archivo de Lenguas Indígenas de México. Colegio de México (1980)
7. Mager, M., Gutierrez-Vasques, X., Sierra, G., Meza-Ruiz, I.: Challenges of language technologies for the indigenous languages of the Americas. In: Proceedings of the 27th International Conference on Computational Linguistics, Association for Computational Linguistics, pp. 55–69 (2018)
8. Mager, M., Meza, I.: Hacia la traducción automática de las lenguas indıgenas de méxico. Proceedings of the DH, (2018)
9. Nekoto, W., Marivate, V., Matsila, T., Fasubaa, T., Fagbohungbe, T., Akinola, S. O., Muhammad, S., Kabenamualu, S. K., Osei, S., Sackey, F., Niyongabo, R. A., Macharm, R., Ogayo, P., Ahia, O., Berhe, M. M., Adeyemi, M., Mokgesi-Selinga, M., Okegbemi, L., Martinus, L., Tajudeen, K., et al.: Participatory research for low-resourced machine translation: A case study in African languages. In: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, pp. 2144–2160 (2020) doi: 10.18653/v1/2020.findings-emnlp.195
10. Reyes-Gómez, J. C.: Aportes al proceso de enseñanza aprendizaje de la lectura y la escritura de la lengua ayuuk. Centro de Estudios Ayuuk–Universidad Indígena Intercultural Ayuuk (2005)
11. Sagi-Vela González, A.: El mixe escrit i el miratge del bon alfabet. Revista de Llengua i Dret, no. 71, pp. 146–157 (2019) doi: 10.2436/rld.i71.2019.3256
12. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, vol. 1, pp. 1715–1725 (2016) doi: 10.18653/v1/P16-1162
13. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, Curran Associates, Inc., vol. 30 (2017) doi: 10.48550/ARXIV.1706.03762

ISSN 1870-4069

Sistema inteligente para el desarrollo de la instrumentación didáctica de asignaturas de educación superior tecnológica

César Rose-Gómez[1], Daniel Hernández-Carrasco[1], Abelardo Mancinas-González[1], Samuel González-López[2], Mirey García-Mora[1]

[1] Tecnológico Nacional de México
Instituto Tecnológico de Hermosillo,
México

[2] Universidad Tecnológica de Nogales,
México

{cesar.roseg, amancinas}@hermosillo.tecnm.mx,
{carrascodanielh samuelgonzalezlopez,
mrocio.garciam}@gmail.com

Resumen. En el sistema nacional de educación de México, desde hace varios años se usa el Modelo Educativo Basado en Competencias (EBC) en todos los niveles educativos. Por tal motivo, desde su incorporación del modelo en el sistema educativo nacional, la capacitación de los maestros se convirtió en una necesidad que conllevó a la realización de cursos y diplomados con mayor frecuencia para mantenerlos actualizados. Entre las actividades que se tienen en este modelo por parte del profesor para alcanzar el objetivo del modelo educativo basado en competencias, es el realizar una planeación de sus cursos mediante una instrumentación didáctica (ID). En esta planeación se deben describir las actividades de enseñanza-aprendizaje en concordancia con las evidencias de aprendizaje que muestren el nivel de competencia alcanzado por el alumno, así como los instrumentos con los cuales se pretende evaluar el nivel de logro antes mencionado. En este artículo se describe un sistema que incluye una plataforma Web para que el docente aprenda y comprenda la metodología para el desarrollo de una ID de acuerdo con el modelo EBC, realizando una mejor estructuración de la planeación de sus cursos. A su vez, el sistema ayudará a tener una retroalimentación inmediata del análisis de la ID, algo que en la actualidad requiere de días para obtenerse por parte del departamento capacitado para la revisión de la ID. El sistema se ha desarrollado con el uso de una ontología como modelo de conocimiento, el procesamiento de lenguaje natural para el análisis de la ID y un módulo de recomendación para la retroalimentación al docente.

Palabras clave: Modelo basado en competencias, instrumentación didáctica, ontología, procesamiento de lenguaje natural, recomendador.

Intelligent System for the Development of Didactic Instrumentation of Technological Higher Education Subjects

Abstract. In Mexico´s national education system, the Competency-Based Educational Model (CBE) has been used for several years at all educational levels. For this reason, since its incorporation of the model into the national

educational system, teacher training became a necessity that led to the completion of courses and diplomas more frequently to keep them updated. Among the activities, that the teacher has to do in this model is the planning of their courses. To achieve the objective of the educational model based on competencies is to carry out a planning of their courses through a didactic instrumentation (DI). In this planning, the teaching-learning activities must be described in accordance with the learning evidences that show the level of competence achieved by the student, as well as the instruments with which it is intended to evaluate the aforementioned level of achievement. This article describes a system that includes a Web platform for the teacher to learn and understand the methodology for the development of a DI according to the CBE model, making a better structuring of the planning of their courses. Moreover, the system will help to have immediate feedback on the DI analysis, something that currently requires days to obtain from the department trained for the DI review. The system has developed with the use of an ontology as a knowledge model, natural language processing for the analysis of DI and a recommendation module for feedback to the teacher.

Keywords: Competency-Based Educational Model, didactic instrumentation, ontology, natural language processing, recommender.

1. Introducción

Desde el año 2008 [1] fue aprobada la reforma educativa promovida por la Secretaría de Educación Pública (SEP) de México y por consiguiente un nuevo paradigma educativo que recibió el nombre de Modelo Educativo Basado en Competencias (EBC) [2, 3]. Este modelo ha sido usado en el sistema nacional de educación en todos los niveles educativos. Uno de estos niveles, es el de educación superior tecnológica [4], al cual pertenece el Tecnológico Nacional de México, institución, que atiende a una población escolar de más de 600 mil estudiantes en licenciatura y posgrado en todo el territorio nacional, incluida la Ciudad de México.

Entre las actividades que se tienen en este modelo por parte del docente para alcanzar el objetivo del modelo educativo basado en competencias, es el realizar una planeación de sus cursos mediante una instrumentación didáctica (ID), donde se planteen las actividades de enseñanza-aprendizaje en conjunto con las evidencias de aprendizaje que se esperan obtener como retroalimentación por parte del alumno y los instrumentos con los cuales se pretenden evaluar [5, 6], teniendo una visión clara sobre la metodología a seguir y no una relativa sobre el procedimiento para una redacción ideal [7].

Sin embargo, aún con la capacitación recibida por parte de los profesores del modelo, se siguen teniendo algunos problemas, en particular, con respecto a la construcción de la instrumentación didáctica. Por un lado, debido a que actualmente en México, para ser un docente del nivel de educación superior no es un requisito tener una formación pedagógica para laborar. Además, la mayoría de los que trabajan en este nivel son profesionistas egresados de alguna licenciatura o posgrado acorde a la materia a impartir, algunos tienen dificultades (en especial en los primeros años en la labor) para redactar la ID de sus cursos al desconocer la metodología que se debe de seguir para su elaboración.

Aunado a lo anterior, cada institución tiene un departamento encargado de la evaluación de las ID, pero la carga de trabajo impide una rápida retroalimentación al docente, quien tiene que esperar días (inclusive semanas) para recibir la aprobación o la lista de ajustes, lo que implica un nuevo proceso para cumplir con los requerimientos del departamento.

Por tal motivo, se ha propuesto un sistema que permita al docente desarrollar su ID a través de un proceso de capacitación. Este sistema incluye una plataforma Web para que el docente aprenda y comprenda la metodología para el desarrollo de una ID de acuerdo con el Modelo EBC, realizando una mejor estructuración de la planeación de sus cursos.

A su vez, la plataforma ayudará a tener una retroalimentación inmediata de la evaluación, algo que en la actualidad requiere de días para obtenerse por parte del departamento capacitado para la revisión al cual se le disminuirá la carga de trabajo, haciéndoles partícipe solo de la evaluación de la versión final del documento después de haber aprobado las recomendaciones de la plataforma.

El sistema se ha desarrollado con el uso de una ontología como modelo de conocimiento, el procesamiento de lenguaje natural para el análisis de la ID y un módulo de recomendación para la retroalimentación al docente, lo cual se describe en las secciones posteriores.

2. Trabajos relacionados

El uso de la tecnología computacional en el ámbito educativo data de los años setenta del siglo pasado [8]. De alguna manera los avances tecnológicos de la computación se han ido utilizando para apoyar el proceso de enseñanza-aprendizaje, sin embargo, en los últimos años las tecnologías de la Web Semántica se han aplicado en los entornos educativos para diversos fines.

Una de estas aplicaciones es la representación de conocimiento. En la Web Semántica la tecnología básica para la representación de conocimiento es la ontología, en [9] se presentan dos experiencias que muestran el uso de ontologías como apoyo a los procesos de evaluación y otras aplicaciones para diferentes necesidades de los usuarios en el entorno educativo.

El autor en [10] propone y plantea la construcción de un modelo de representación de conocimiento para las competencias educativas de enseñanza superior en el marco de estudios de grado mediante la construcción de una red de ontologías, con lo cual espera el desarrollo de aplicaciones que permita la búsqueda de información oportuna por parte de los estudiantes, personal académico y administrativo.

En [11] se describe una revisión sistemática para buscar modelos de diseño curricular, de competencias o de dominios específicos de formación, en los cuales se estuviera usando ontologías u otra representación de conocimiento relacionados con el dominio del diseño curricular basado en competencias. En este artículo [12] los autores presentan el diseño y construcción de una ontología para apoyar el diseño de secuencias didácticas con un enfoque de competencias en educación media superior.

La combinación de ontologías y el procesamiento de lenguaje natural (PLN) para representar competencias y su gestión es un campo de investigación que está propiamente en sus inicios. En [13] presenta un esquema para la actualización de

competencias profesionales y académicos desde perfiles obtenidos de la Web, a la vez que se usan ontologías para modelar las competencias y el PLN para encontrar patrones lingüísticos mediante tesauros, con lo cual permite poblar las ontologías.

3. Antecedentes

Si bien, el Modelo EBC no tiene al docente como el núcleo central sino al estudiante, su éxito depende en gran medida de la capacidad del primero para desempeñarse en el aula donde la formación pedagógica juega un papel crucial para guiar el conocimiento [1]. Por ello, es importante que se conozca y se desarrollen los temas de la asignatura de una manera crítica y formativa haciendo énfasis en su comprensión, con el fin de aplicarlo a problemas reales de interés para el estudiante en función del perfil de egreso deseado [1, 6].

En este punto, la capacitación y formación son importantes para poder diseñar una ID adecuada con la que se puedan cumplir las competencias planteadas [1, 6]. Sin embargo, la comprensión del término competencia por sí mismo es complejo [7], debido a que cada docente lo entiende a su manera en base a las experiencias de redacción de años anteriores, acumulando errores de comprensión sin percatarse de su existencia.

El desconocimiento de una metodología clara que permita la realización de una ID que relacione las actividades de enseñanza y aprendizaje para posteriormente hacer una evaluación del desempeño del estudiante haciendo uso de un instrumento adecuado, han retrasado significativamente los resultados del Modelo EBC [6, 14]. Como base, una ID tiene 4 secciones importantes para definir las actividades que se utilizarán a lo largo del curso y el cómo estas serán evaluadas [6, 12]:

- Actividades de Aprendizaje: procedimientos o actividades realizadas por los alumnos para participar en el proceso de formación con el fin de reforzar los conocimientos proporcionados en clase o, de forma autodidacta, adquirir nuevos.
- Actividades de Enseñanza: actividades, técnicas, métodos o procedimientos que el docente utiliza para conducir el proceso de enseñanza-aprendizaje. También se le conoce como estrategias instruccionales.
- Evidencias de Aprendizaje: producto de forma individual y/o grupal que demuestran los resultados del proceso de aprendizaje por parte del alumno.
- Instrumentos de Evaluación: herramientas utilizadas para evaluar las evidencias del desempeño del estudiante entregadas durante el proceso de enseñanza- aprendizaje.

En la secuencia que se utiliza para llegar desde las actividades de enseñanza hasta los instrumentos de evaluación, es vital que se tenga una congruencia entre estas etapas con el fin de dar coherencia a las competencias a desarrollar por el estudiante en la asignatura [7].

Por lo tanto, es importante que el docente comprenda la importancia de estas relaciones y, a la vez, que cuente con una profesionalización pedagógica que le permita hacer uso de la metodología para una correcta redacción de su ID acorde al paradigma del Modelo EBC.

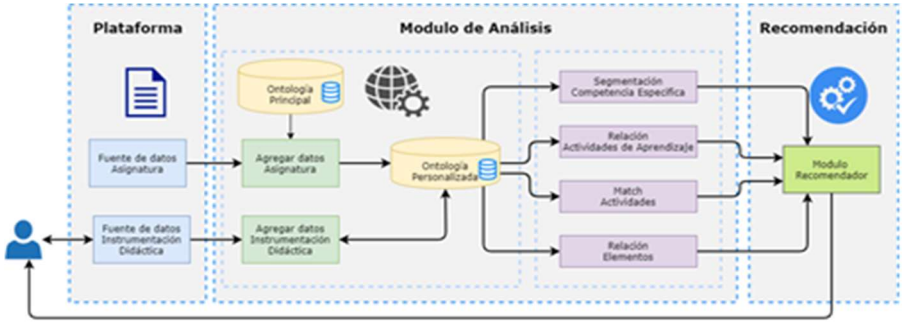


Fig. 1. Arquitectura general del sistema.

Como apoyo a esta labor del docente, se han realizado distintas clasificaciones sobre objetivos de aprendizaje para el diseño de estrategias de enseñanza, siendo una de las más utilizadas la Taxonomía de Bloom [15].

4. Estructura general del sistema

Como se puede apreciar en la Fig. 1, el sistema está constituido de tres componentes principales: la plataforma Web, el módulo de Análisis de la Instrumentación Didáctica (ID) y el módulo de Recomendación. Cada uno de esto componentes se describen a detalle posteriormente.

La plataforma Web permite al docente desarrollar la ID correspondiente a los cursos a impartir en el semestre. Se tiene interacción con el módulo de Análisis de la ID, el cual es un algoritmo inteligente que evalúa la ID de asignaturas de nivel superior tecnológica utilizando técnicas de PLN y un modelo de conocimiento.

Asimismo, dependiendo del resultado del análisis, el módulo de Recomendación permite que el docente reciba una retroalimentación al ir desarrollando su ID y tenga el apoyo por parte del sistema y por el instructor que esté realizando la capacitación correspondiente, teniendo como objetivo el obtener una ID correcta.

4.1. Plataforma Web

El objetivo de la plataforma Web es tener las funcionalidades necesarias para realizar el desarrollo de la ID, de tal manera, que se deben tener todos los servicios para el 'back-end' (servicios transparentes para el usuario), así como, las diferentes vistas del 'front-end' (interfaz gráfico de la página web) y sus correspondientes modelos de datos y conocimiento. El back-end, Fig. 2, procesa las interacciones del usuario (realizadas en el front-end) con los datos, y realiza todos los procesos con estos últimos.

Esta parte es la que proporciona realmente de funcionalidad al sistema, aunque sin el front-end, no funcionaría pues no podríamos interactuar con los datos y sus procesos. Es en esta parte donde subyacen todos los algoritmos que realizan el trabajo requerido del sistema, el acceso a los datos, su manipulación, al tiempo que desarrollan sus funcionalidades.

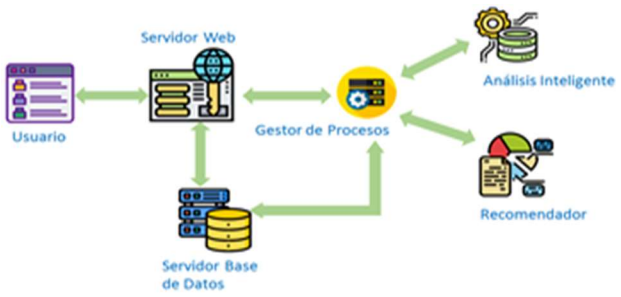


Fig. 2. Arquitectura Cliente Servidor de la plataforma.

4.2. Diseño e implementación de la ontología de la instrumentación didáctica

El diseño y construcción de una ontología requiere del seguimiento de una serie de actividades que permitan modelar el dominio a nivel conceptual a partir de la adquisición de conocimiento, para posteriormente ser transformado a un modelo formal en algún lenguaje ontológico como puede ser OWL o RDF. Una de las metodologías que permite su diseño y construcción, partiendo desde un nivel de conocimiento es la metodología Methontology compuesta por 5 pasos esenciales [16]:

– Especificación: limita el dominio a un área específica de conocimiento.

– Conceptualización: mediante la adquisición de conocimiento, permite la creación de glosarios que contienen los conceptos, atributos, taxonomías y sus relaciones.

– Formalización: admite convertir el modelo de conceptualización en un modelo formal utilizando lenguajes ontológicos.

– Implementación: haciendo uso de herramientas tales como Protégé, se convierte el modelo formal en computable.

– Mantenimiento: permite realizar actualizaciones.

El dominio de la presente ontología contiene información referente a la terminología y relaciones encontradas en una ID de asignaturas de nivel superior tecnológica, por lo que su dominio es específico de esta temática.

Debido a que esta metodología es muy minuciosa y es imposible describirla totalmente en este artículo, sólo se presenta algunos aspectos de la ontología. Una parte crucial de la ontología es la que abarca la Taxonomía de Bloom que está relacionada, mediante los verbos de dominio, a las clases correspondientes a las actividades de aprendizaje, actividades de enseñanza, instrumentos de evaluación y a las evidencias de aprendizaje, tal y como se muestra en la Fig. 3.

Además, la subdivisión de la Taxonomía en sus 6 clases principales (análisis, aplicación, conocimiento, síntesis, comprensión y evaluación) adjunto a la correspondencia con las actividades, implica una extensa lista de relaciones que pueden inferirse. Por lo que el uso de la ontología para almacenar la información de la taxonomía y la deducción del motor de inferencia para crear nuevas relaciones a partir

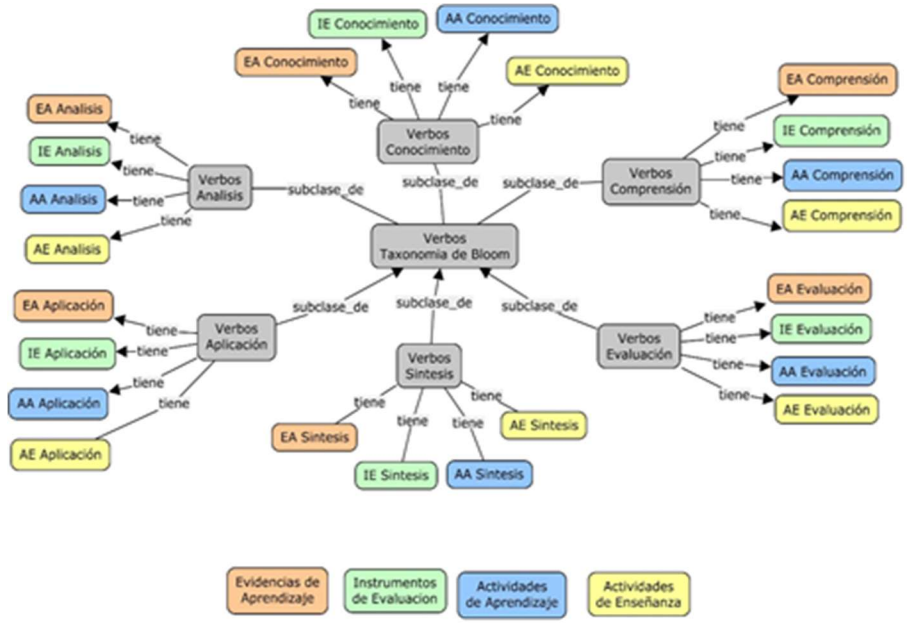


Fig. 3. Vista de la Taxonomía de Bloom en la ontología.

de las ya existentes, son la principal justificación para su utilización en la presente investigación.

Primeramente, se jerarquizaron las clases y subclases formando la taxonomía de la ontología, donde se obtuvieron 21 clases principales mostradas en la Fig. 4 como una vista parcial de los 83 conceptos. También, se puede observar a las actividades de aprendizaje separadas en predefinidas y seleccionadas.

En la primera categoría se tienen subclases correspondientes a los dominios de la taxonomía de Bloom, separación que facilita su recuperación específica. La misma lógica se sigue para las actividades de enseñanza, las evidencias, los instrumentos y los verbos de la competencia.

En Protégé [17], las relaciones entre los conceptos reciben el nombre de propiedades de objetos y permiten hacer inferencia de conocimiento mejorando la recuperación de la información al aportar por sí mismas conocimiento. En este modelo se cuenta con 49 propiedades de objeto diferentes. Las instancias son representaciones de objetos en el dominio sobre el cual se está trabajando y su relación con las clases generalmente está dada del tipo “esUn”.

La taxonomía de Bloom establece una lista predeterminada de verbos y actividades, por lo que es necesario hacer su declaración como instancias y así estos puedan ser consultados por el módulo de análisis. En total, se cuenta con 486 individuos que, en su mayoría pertenecen a los verbos de la taxonomía y a las actividades predeterminadas.

En cuanto a la validación de la ontología, se realizó con dos herramientas, ¡la primera de ellas es OOPS! (Ontology Pitfall Scanner) la cual permite en línea realizar un análisis de la ontología para encontrar hasta 26 tipos de problemas en la construcción de la ontología. La segunda evaluación se realizó dentro de Protégé con OntoDebug, la cual permite encontrar inconsistencias o incongruencias.

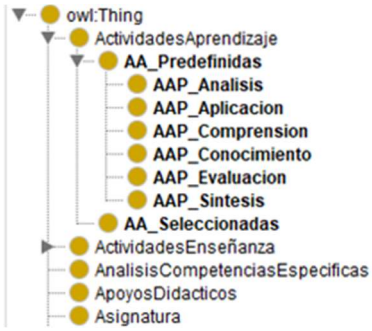


Fig. 4. Vista parcial de los conceptos de la ontología.

4.3. Análisis inteligente de la competencia especifica

El proceso de análisis inteligente de la ID tiene dos subprocesos: el primero corresponde a la población de la ontología con los datos recibidos de la plataforma de capacitación; mientras que el segundo es el proceso de análisis utilizando técnicas de Procesamiento de Lenguaje Natural (PLN) y consultas al modelo de conocimiento con el lenguaje SPARQL para la recuperación de información.

Para una mejor visualización, en la Fig. 5 se muestra el diagrama DFD del proceso de análisis y a continuación, se describen algunos de los procesos:

Proceso 1 Autómata: Segmentación de la competencia específica utilizando un análisis morfológico para la detección de las partes esenciales (verbo, actividad y finalidad o condición de ejecución), así como la determinación del dominio al cual pertenece basándose en la Taxonomía de Bloom.

En México, las competencias específicas están redactadas en base a la Taxonomía de Bloom clasificada en seis niveles que contienen verbos claves para su identificación.

En ella, los niveles van aumentando gradualmente de complejidad y, generalmente suelen ser representados como una pirámide, jerarquía que ayuda a clasificar tanto el verbo como a los objetos de aprendizaje.

Partiendo de esa información y del conocimiento de los expertos, se puede concluir que una competencia está compuesta por tres secciones:

competencia = verbo + objeto + (finalidad y/o condición de ejecución)

Siguiendo ese patrón en su redacción, la detección de las entidades que componen la competencia se realiza mediante un análisis morfológico para determinar el papel gramatical que desempeña cada una de las palabras dentro de la oración. La Fig. 6 contiene el flujo del análisis partiendo desde el texto redactado por el profesor hasta la obtención de la lista de actividades en base al dominio de la taxonomía.

Para el análisis morfológico, se emplea la librería de Freeling[1] que otorga el etiquetado de la categoría gramatical de las palabras. El autómata utiliza para la segmentación aquellas palabras cuya etiqueta gramatical[2] indique que es un verbo, conjunción, sustantivo, pronombre o inclusive signos de puntuación, cuyas variantes

[1] http://nlp.lsi.upc.edu/freeling/node/1

[2] https://www.cs.upc.edu/~nlp/tools/parole-sp.html

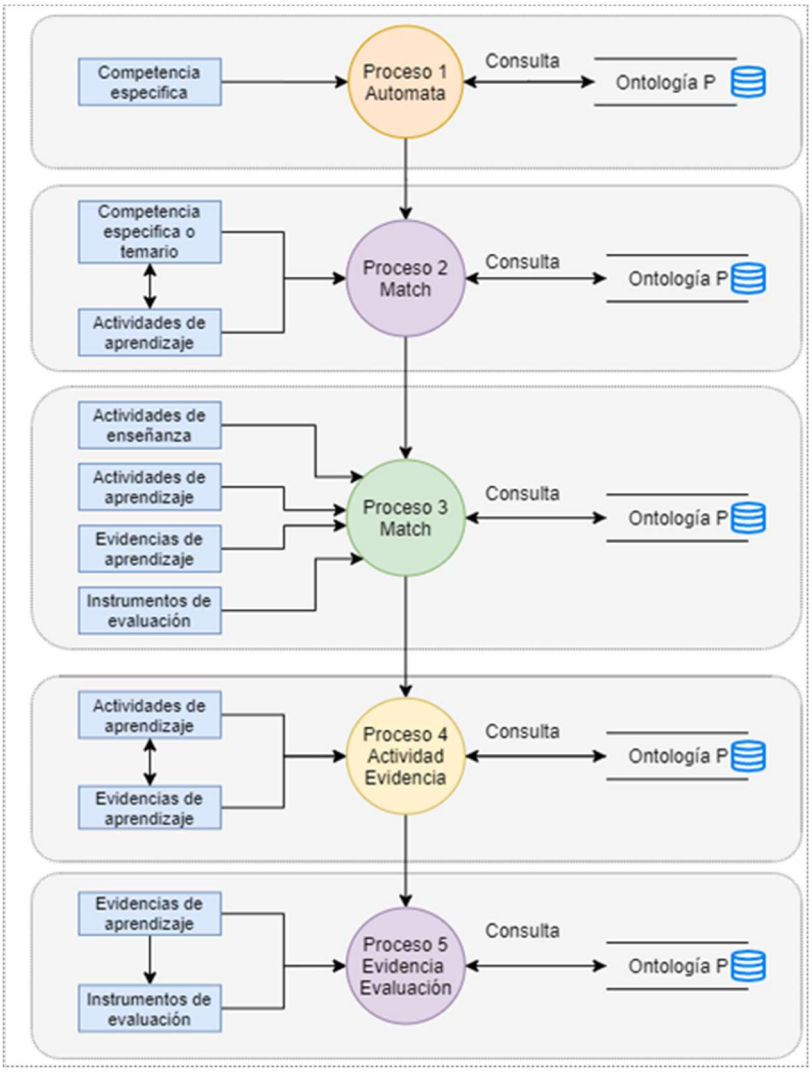


Fig. 5. Diagrama del proceso de análisis de la competencia especifica.

permiten realizar el cambio entre los estados del autómata. Secuencia, esta última, que se puede observar en la Fig. 7 donde se muestra un estado inicial (S0) y tres posibles estados finales (S0, S3, S5). En cuanto a su alfabeto es el siguiente:

V: El token es un verbo.

NP: Sustantivo propio.

CC: Conjunción coordinada.

PR: Pronombre de tipo relativo.

CS: Conjunción subordinada.

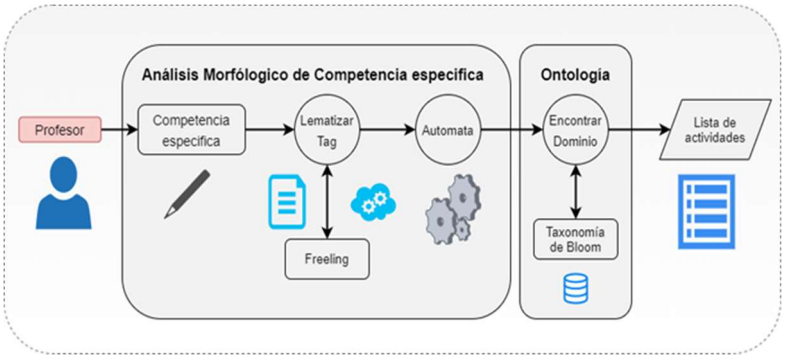


Fig. 6. Secuencia de análisis de la competencia especifica.

FC: Signo de puntuación de tipo coma.

X-Y: Cualquier etiqueta gramatical que sea excepción de las anteriores.

E1-E2-E3: Representan excepciones a la regla.

Para determinar el dominio de una competencia, esta debe de comenzar con un verbo o un sustantivo propio, de no ser así no es posible determinar su dominio.

Además, se toma en consideración que se puede tener más de un verbo de dominio en la competencia, siendo la transición del estado S1 al S2 en el cual se determina esta acción.

En cuanto a las excepciones de la regla, son situaciones en las cuáles la redacción contiene elementos únicos que provocan que no se pueda hacer la segmentación de acuerdo con el esquema convencional, siendo estas excepciones quienes permiten realizar la separación y obtener la finalidad y/o condición de ejecución.

Por ejemplo, si se analiza la competencia "Comprende y aplica estructuras de datos lineales para solución de problemas", se tiene el siguiente resultado:

Verbo: Comprende y Aplica	*Objeto: estructura de datos*	*Finalidad: para solución de problemas*

Con la detección de los verbos de la competencia, se determina su dominio haciendo una consulta a la ontología para recuperar las actividades y verbos del dominio utilizados en el proceso de match. Además, teniendo el tercer elemento identificado se busca su relación con las actividades de aprendizaje.

Proceso 2 Match de actividades de aprendizaje: La condición de ejecución de la competencia debe de estar relacionada con al menos una de las actividades de aprendizaje que han sido redactadas. A la vez, estas deben de estar enfocadas en alguno de los temas de la unidad.

Debido a que el Modelo EBC centra al alumno como el actor principal en el proceso educativo, las actividades de aprendizaje que se han elegido son un punto de inflexión al estar relacionadas directamente con la competencia de la unidad y el temario.

Primeramente, el tercer elemento que compone a la competencia debe de estar relacionado directamente con al menos una de las actividades redactadas, de las que también se espera que esté complementada con algún subtema.

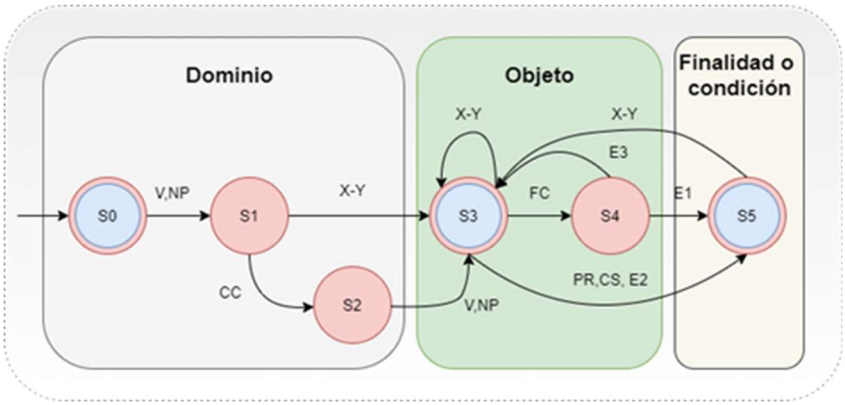


Fig. 7. Autómata para segmentar la competencia especifica.

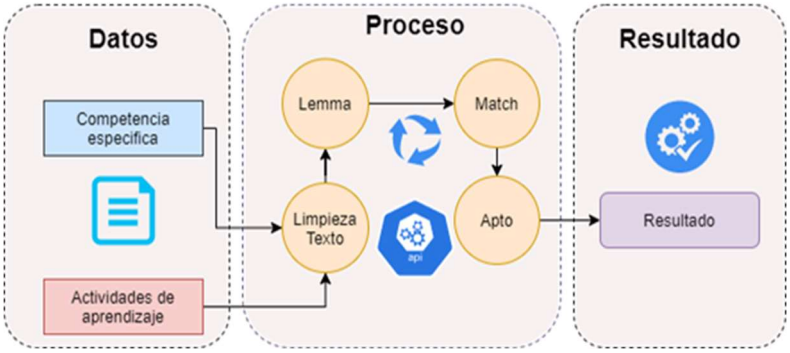


Fig. 8. Proceso de match de actividades de aprendizaje.

La Fig. 8 contiene un esquema base del proceso que se sigue para la búsqueda del match, considerando el pre-procesamiento del texto y el ciclo repetitivo que se rompe cuando se ha encontrado una relación entre la competencia y al menos una de las actividades.

4.4. Módulo de recomendación

El módulo de recomendación emite trece tipos de recomendaciones en base al análisis inteligente de la competencia específica, debido a que se dan dos tipos de recomendación para cada uno de los elementos: verbo, verbo en infinitivo, relación entre competencia y actividades de aprendizaje, relación entre actividades de aprendizaje y evidencias de aprendizaje y, relación entre evidencias de aprendizaje e instrumentos de evaluación

La primera vez que se encuentra un error, se muestra una lista de objetos de aprendizaje recomendados para cada caso en específico, esto se muestra en la Fig. 9. En el caso de las relaciones también se muestran las actividades, evidencias e instrumentos de evaluación en los que se haya errado. Por otro lado, a partir del segundo

Tabla 1. Resultados evaluación del análisis de la ID.

Actividad	Total	Igual	Diferente	Bien	Mal	Porcentaje
Competencia	19	19	0	13	6	100
Actividades de aprendizaje	116	104	12	74	30	89.65
Actividades de enseñanza	91	86	5	55	31	94.50
Evidencias de aprendizaje	65	61	4	40	21	93.84
Instrumentos de evaluación	65	65	0	63	2	100
Match Competencia-Aprendizaje	114	85	29	53	32	74.56
Match Aprendizaje-Evidencia	116	112	4	82	30	96.55
Match Evidencia-Aprendizaje	65	62	3	58	4	95.38
Match Evidencia-Instrumentos	65	65	0	59	6	100
Total	715	658	57	496	162	92.02

intento tanto en la creación de la competencia como en las relaciones, se recomiendan verbos, actividades, evidencias o instrumentos específicos según sea el caso.

5. Resultados

Para evaluar el desempeño del algoritmo para el análisis de la ID, se realizó una comparativa entre sus resultados y los de uno de los expertos que aportó su conocimiento al modelo ontológico. La Tabla 1 contiene los datos de los 715 elementos analizados durante las pruebas donde se coincidió en 658 (92.03%) de ellos. De estos últimos, 496 (75.37%) se consideró que cumplían con el patrón de redacción y 162 (24.63%) fueron enviados directamente a revisión debido a la inexistencia de un verbo o subtema para acompañar al objeto didáctico.

Como punto negativo, en 57 (7.97%) de los elementos analizados, el algoritmo y el evaluador obtuvieron resultados diferentes de los cuales 29 (50.87%) se dieron en la relación entre la competencia específica y las actividades de aprendizaje. Cabe resaltar que la mayor parte de estas divergencias se dieron al incluir el conocimiento empírico del evaluador, debido a que en el texto la correlación no estaba explícitamente escrita y era necesario inferir conocimiento muy específico que no fue incluido en la ontología.

6. Conclusiones y trabajo a futuro

En las pruebas realizadas se encontraron deficiencias en la redacción de las actividades donde son omitidos elementos básicos como el verbo para identificar el dominio o el tema que será tratado con dicha actividad.

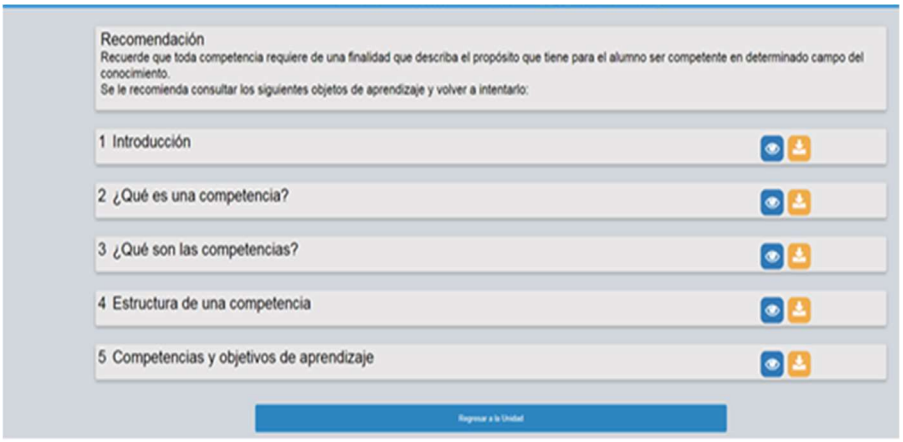


Fig. 9. Vista de una recomendación.

A pesar de que la mayor parte de los análisis se realizaron a ID redactadas por docentes que aún no han llevado el curso de capacitación y formación de la plataforma, en el caso de la prueba piloto quedó demostrado que es posible que un docente redacte su ID en base a la metodología del modelo EBC y el algoritmo sea capaz de detectar si cumple o no con el patrón.

Como principal contribución, está el diseño del modelo de conocimiento aplicado a ID de educación superior, así como el algoritmo de PLN para la extracción de información con el fin de encontrar los patrones de redacción. Los resultados obtenidos denotan el cumplimiento de los objetivos planteados al inicio de la investigación. Así mismo, se muestra la factibilidad de la aplicación del algoritmo para otorgar una evaluación al momento, ya que se obtuvo una eficiencia del 92% evaluando una ID del área superior tecnológica.

Por otra parte, el tiempo que este requiere para analizar una ID completa es de aproximadamente 25 segundos, a diferencia del tiempo que requiere un evaluador certificado que puede tardar más de 30 minutos para la misma labor. El módulo puede servir, de igual forma, para aminorar la carga de trabajo de este último al requerir de su participación únicamente en la evaluación final.

A futuro, a partir de una colección de ID redactadas, obtenidas a partir de la capacitación de docentes siguiendo la metodología de la plataforma, es factible el utilizar aprendizaje máquina para la detección de patrones entre ellas, extrayendo las características y relaciones que no han sido consideradas en el algoritmo actual debido a la cantidad limitada de ID con las que se contó para las pruebas. También, este apartado ayudaría a mejorar la detección de la relación entre la competencia y las actividades de aprendizaje.

Agradecimientos. Se agradece al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el apoyo económico otorgado al segundo autor mediante la beca número 15989 y al quinto autor mediante la beca número 56046.

Referencias

1. Muñoz-García, L. R. T., Gómez-Zermeño, M. G., Alemán de la Garza, L. Y.: Uso de la plataforma educativa Moodle en los procesos de capacitación de maestros de Educación Indígena en Jalisco, México. Zo. próxima Rev. el Instituo Estud. Super. Educ., no. 24, pp. 28–42 (2017) doi: 10.14482/zp.24.8719
2. Arroyo-Martínez, S.: Innovación en el diseño de modelos educativos basados en competencia en las instituciones de educación superior en México. Rev. Educ. Super., vol. 2, no. 5, pp. 20–31 (2018)
3. Secretaría de Educación Pública: Principales cifras del Sistema Educativo Nacional 2018-2019. Dir. Gen. Planeacion, Porgramacion y Estad. Educ., vol. 53, no. 9, pp. 130 (2019)
4. Tecnológico Nacional de México: TecNM. https://www.tecnm.mx (2021)
5. Torres-Ríos, H., Larios-Hijar, I. H., Medina-Alcázar, A. C.: Neuroaprendizaje, actividades de enseñanza, actividades de aprendizaje e instrumentos de evaluación del aprendizaje. Debates en evaluación y currículum, vol. 24, no. 4 (2018)
6. Feo-Mora, R.: Orientaciones básicas para el diseño de estrategias didácticas. Tendencias pedagógicas, vol. 16, pp. 221–236 (2010)
7. Suárez-González, M.: Influencia del relativismo en la instrumentación didáctica y el desempeño escolar. RECIE. Rev. Electrónica Científica Investig. Educ., vol. 4, pp. 179–189 (2018)
8. Chiappe, A., Sánchez, J.: Informática educativa: Naturaleza y perspectivas de una interdisciplina, Rev. Electrónica Investig. Educ., vol. 16, no. 1-18, pp. 135–151 (2014)
9. Sánchez-Vera, M. M., Tomás-Fernández, J., Serrano-Sánchez, J. L. M., Prendes-Espinosa, M. P.: Practical experiences for the development of educational systems in the semantic web. J. New Approaches Educ. Res., vol. 2, no. 1, pp. 23–31 (2013) doi: 10.7821/near .2.1.23-31
10. Merino-Ruiz, L.: Desarrollo de ontología de competencias educativas. Universidad Politécnica de Madrid (2017)
11. Guerra, D., Cárcamo, L.: Revisión Sistemática de modelos de representación del conocimiento para el dominio del diseño curricular basado en competencias. https://recursos.educoas.org/sites/default/files/2041.pdf (2021)
12. Cerón Garnica, C., Archundia Sierra, E., Beltrán Martínez, B., Cervantes Márquez, P., Galindo-Cruz, J. L.: Elaboración de una ontología para apoyar el diseño de secuencias didácticas basadas en competencias en la práctica del docente de educación media superior. Res. Comput. Sci., vol. 99, no. 1, pp. 115–126 (2015) doi: 10.13053/rcs-99-1-11
13. González-Eras, A., Aguilar, J.: A scheme for updating the competency ontologies based on natural language processing and semantic mining [Esquema para la actualización de ontologías de competencias en base al procesamiento del lenguaje natural y la minería semántica], RISTI - Rev. Iber. Sist. e Tecnol. Inf., no. E17, pp. 433–447 (2019)
14. Martínez-Rodríguez, R., Benítez-Corona, L., Vazquez Mora, J. A.: La instrumentación Didáctica en Transición de una Educación Tradicional a una Basada en Competencias. ANFEI Digital (2014)
15. Solórzano-Zamora, H., Caballero-Vera, H. H.: Innovación metodológica para elevar el nivel de aprendizaje de la Química. Rev. Didasc@lia Didáctica y Educ., vol. 10, pp. 161–176 (2019)
16. Gómez-Pérez, A., Fernández-López, M.: Ontological engineering: With examples from the areas of knowledge management, e-commerce and the semantic web. Springer (2004)
17. Protégé: https://protege.stanford.edu/ (2020)

ISSN 1870-4069

Detección temprana de enfermedades cardiovasculares a través de análisis de biomarcadores y modelos de predicción

Alejandra Montiel de Jesús[1], Nancy Aracely Cruz Ramos[1], Lisbeth Rodríguez Mazahua[1], Luis Ángel Reyes Hernández[1], Luis Rolando Guarneros Nolasco[1], José Luis Sánchez-Cervantes[2]

[1] Tecnológico Nacional de México, México

[2] CONACYT- Instituto Tecnológico de Orizaba, México

{alejandramontieldj,nancy.cramos5, luisguarneros}@gmail.com, {lrodriguezm, lreyesh} @orizaba.tecnm.mx, jlsanchez@conacyt.mx

Resumen. Las enfermedades cardiovasculares son una de las principales causas de muerte en el mundo. En México cada año fallecen alrededor de 140 mil personas a consecuencia de estas enfermedades. Por estas razones se están realizando soluciones para prevenir y dar seguimiento a las enfermedades cardiovasculares, mediante el uso de condiciones, árboles de decisión, algoritmos y modelos de predicción. En este trabajo se propone una solución enfocada a la detección temprana de enfermedades cardiovasculares, para apoyar a la población a prevenir alguna complicación en su salud, mediante el estudio y selección de biomarcadores, así como modelos de predicción que ofrecen la mejor exactitud. Con el propósito de ofrecer a las personas sintomáticas o asintomáticas una herramienta validada por médicos especialistas se ha desarrollado una aplicación para dispositivos móviles que contiene el medio de extracción de datos y proporciona los resultados obtenidos de la detección temprana a través de interfaces gráficas de usuario amigables.

Palabras clave: enfermedades cardiovasculares, modelos de predicción, biomarcadores, aplicación móvil.

Early Detection of Cardiovascular Diseases from Biomarker Analysis Using Prediction Models

Abstract. Cardiovascular diseases are one of the leading causes of death in the world. In Mexico each year around 140 thousand people die as a result of these diseases. For these reasons, solutions are being developed to prevent and monitor cardiovascular diseases, through the use of conditions, decision trees, algorithms and prediction models. In this work, a solution focused on the early detection of cardiovascular diseases is proposed, to support the population to prevent any

complication in their health, through the study and selection of biomarkers, as well as prediction models that offer the best accuracy. In order to offer symptomatic or asymptomatic people a tool validated by medical specialists, an application for mobile devices has been developed that contains the means of data extraction and provides the results obtained from the detection in friendly graphical interfaces.

Keywords: Cardiovascular diseases, prediction models, biomarkers, mobile app.

1. Introducción

En las últimas décadas se ha observado un notable incremento de las enfermedades crónicas no trasmisibles asociadas a estilos de vida no saludables. En población adulta el sobrepeso y la obesidad se asocian con una mayor mortalidad por distintas causas, particularmente el aumento del índice cintura-cadera revela mayor riesgo cardiovascular y el perímetro de cintura asociado a un índice de masa corporal elevado constituye un factor de riesgo independiente para enfermedad cardiaca coronaria y diabetes tipo 2 [1].

Sumándole a estos datos el hecho de que algunas personas fuman, tienen antecedentes familiares con enfermedades cardiovasculares y además no practican alguna actividad física, suma a la probabilidad de contraer una enfermedad cardiovascular.

Para evitar complicaciones a futuro, investigadores y médicos elaboran soluciones implementando calculadoras de riesgo cardiovascular, árboles de decisión y modelos de predicción, utilizando datos obtenidos de análisis sanguíneo y revisión de los pacientes con síntomas.

A diferencia de las soluciones revisadas, en este trabajo se realizó un estudio de biomarcadores y medios de adquisición que permitan visualizar los cambios de frecuencia cardiaca, síntomas, alteraciones y movimiento físico tanto de pacientes sintomáticos como asintomáticos; para proveer una herramienta que brinde el porcentaje de riesgo de alguna enfermedad cardiovascular, de esta manera ayudar a prevenir a tiempo y mejorar las condiciones de vida de las personas.

El documento se divide en tres apartados, la primera consta de la revisión de trabajos similares a la propuesta y datos actuales de las complicaciones que presentan las enfermedades cardiovasculares.

El segundo apartado trata de los artefactos de investigación utilizados, en la que se describe a detalle el proceso de análisis y selección de biomarcadores, modelos de predicción y el despliegue de datos en los dispositivos móviles. Por último, se encuentran las conclusiones y el seguimiento que se pretende dar en un futuro.

2. Trabajos relacionados

De acuerdo con Sacco et al., [2] las enfermedades cardiovasculares (ECV) son una de las principales causas de muerte en el mundo, sobre todo en países donde predominan el consumo de cigarro, alimentos altos en grasas saturadas y azucares, así como la falta de actividad física.

Los autores proyectaron que para el año 2025 más de cinco millones de hombres y 2,8 millones de mujeres tendrán una muerte prematura por ECV. Los resultados presentados muestran la variabilidad sustancial en la carga global de mortalidad con una probabilidad mucho mayor en los países de ingresos bajos y medios que en los países de ingresos altos

Con base en lo anterior, y de acuerdo con [2] los proyectos que analizan la detección temprana de ECV buscan reducir la muerte prematura a 3,5 millones de hombres y 2,2 millones de mujeres, contemplando principalmente factores de riesgo como: presión arterial, tabaquismo, diabetes mellitus y obesidad.

Para comprobar la efectividad de los modelos predictivos, en [3] realizaron una comparación de algunos modelos entre ellos se encontraron Random Forest, modelos de Cox y Regresión Logística. En las pruebas realizadas encontraron que los modelos basados en datos utilizados sobre conjuntos de datos extendidos pueden superar a los modelos convencionales para el pronóstico de enfermedades, sin preprocesamiento de datos ni imputación de valores perdidos.

Aunado a esto, Ogundimu et al., [4] indicó que otra manera de evaluar un modelo de predicción es el uso de una evaluación externa e independiente de la validación del conjunto de datos, es decir; se realizar una selección de variables autorizadas por médicos especialistas, que son utilizadas en el modelo, con una tarea llamada Evento por Variable (EPV).

Para obtener una detección confiable es importante manejar datos reales de las personas para ello, en [5] usaron sensores incorporados en teléfonos inteligentes para recopilar datos, los cuales fueron tratados con transformaciones matemáticas, para eliminar los componentes de vagabundeo y ruido.

Mientras que en [6] y; [7] utilizaron sensores con la finalidad de inspeccionar la actividad física que realizan las personas y los movimientos bruscos que alteran la frecuencia cardiaca, para brindar acompañamiento a distancia de cada persona y proporcionar sugerencias que mejoren su salud.

Los resultados cumplieron satisfactoriamente los objetivos propuestos para apoyar a las personas a prevenir las ECV.

En [8] revisaron los *wearables* para conocer su funcionalidad y las distintas herramientas que contienen para ser aprovechadas en la generación de soluciones tecnológicas para la detección de enfermedades. Finalmente, en [9] realizaron una comparación de las mediciones de frecuencia cardiaca que captan los dispositivos, usando tres *wearables*: un reloj inteligente, un rastreador de ejercicios y un dispositivo especializado.

Los autores demostraron que los datos capturados por los *wearables* son confiables, reales y aptos para ser manipulados por otros dispositivos para ser tratados con algoritmos, condiciones y cálculos para brindar a las personas los resultados sobre su salud actual.

Como se observa en estas aportaciones, es sumamente importante la participación que tiene el avance de la tecnología en el área de la salud, para la prevención y seguimiento de enfermedades cardiovasculares.

La extracción de datos y el uso de los modelos de predicción permiten conocer riesgos a futuro. Este trabajo tiene por objetivo apoyar a la detección temprana de ECV, de personas que presentan síntomas, tienen familiares con alguna enfermedad o son asintomáticas, utilizando dispositivos que proveen datos actuales y de manera

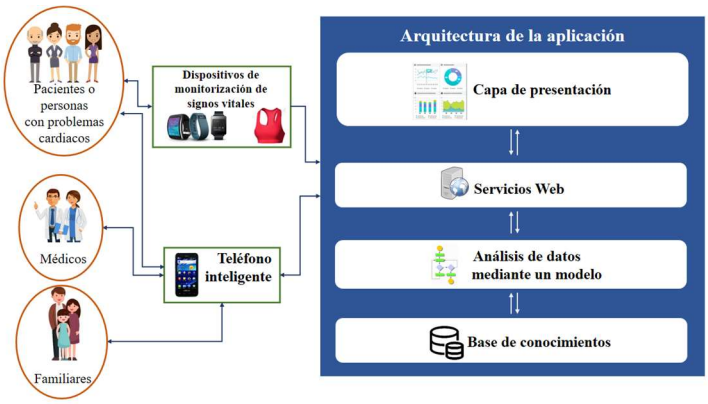


Fig. 1. Arquitectura de la solución.

automática sobre la frecuencia cardiaca, actividad física y presión arterial. Complementando la información con datos manuales para obtener un mejor resultado y dar a conocer la probabilidad general y parcial a las personas, es decir, mostrar los porcentajes de las posibles enfermedades que puede llegar a tener el paciente, a partir de esto brindarle sugerencias para mejorar sus condiciones de salud.

A diferencia de los trabajos analizados, nuestra iniciativa realiza un análisis para valorar el porcentaje de riesgo de padecer una enfermedad cardiovascular, utilizando el modelo que se ha seleccionado de acuerdo con las pruebas realizadas a partir del *wearable*.

3. Artefactos de la investigación

Para proporcionar una alternativa de solución en apoyo a la detección temprana de enfermedades cardiovasculares, se plantea una aplicación para dispositivos móviles que realice una detección temprana de las ECV, brindando a las personas el porcentaje total y parcial de obtener alguna ECV con los datos que se analicen.

La detección se realiza mediante la implementación de un modelo de detección. En la Fig. 1 se presenta la arquitectura de la solución.

La arquitectura muestra la interacción de los tipos de usuarios, los cuales son: 1) Pacientes o personas con problemas cardiacos: Personas sintomáticos o asintomáticos que tienen interés o recomendación de analizar la probabilidad de llegar a obtener alguna ECV, con una detección temprana; 2) Médicos: Personal médico especializado en enfermedades cardiacos y cardiovasculares, que aprobarán los datos a ingresar en el software y los resultados finales y, 3) Familiares: Personas a quienes será posible compartir la información obtenida en la aplicación.

También se representan los dispositivos de monitorización de signos vitales, esto corresponde a los *wearables* en este caso se utilizó el Fitbit Charge 4 para extraer de manera automática biomarcadores del paciente mientras que el teléfono inteligente es el medio para recuperar datos de manera manual, así como para presentar los resultados obtenidos de la detección temprana de ECV.

Tabla 1. Variables para los modelos de detección.

Variable	Descripción
Sexo	1 = M, 0 = F
Edad	En años
Fumador	0 = No, 1 = Sí
Uso de medicamento	0 = No, 1 = Sí
Antecedente de preinfarto	0 = No, 1 = Sí
Antecedentes familiares con hipertensión	0 = No, 1 = Sí
Antecedentes familiares con diabetes	0 = No, 1 = Sí
Colesterol	En mg/dl
Presión sistólica	En Mm
Presión diastólica	En Mm
Masa corporal	Kg/m
Frecuencia cardiaca	Frecuencia máxima
Glucosa	En mg
RiesgoECV	0 = No, 1 = Sí

Tabla 2. Clasificación de variables por método de extracción.

Automático (*wearable*)	Manual
Presión sistólica	Sexo
Presión diastólica	Edad
Frecuencia cardiaca	Fumador
	Uso de medicamento
	Antecedente de infarto
	Antecedente de hipertensión
	Antecedente de diabetes
	Colesterol
	Masa corporal
	Glucosa

Para llevar a cabo la investigación se llevó a cabo una metodología que consta de 5 fases:

1. Identificación de variables críticas en la ocurrencia de ECV.
2. Extracción de biomarcadores. Se extraen biomarcadores a partir del *wearable* del Fitbit.
3. Análisis de modelos para la detección temprana de ECV.
4. Diseño de pruebas de los modelos y selección de la muestra.
5. interpretación de resultados.

```
/*Uso de bibliotecas y extensiones*/
import React from 'react';
import { useTranslation } from 'react-i18next';
import { Col, Container, Row } from 'reactstrap';
import AnimatedLineFormWithLabels from
   './components/AnimatedLineFormWithLabels';
import showResults from '../Show';

const FloatingLabelsForm = () => {
const { t } = useTranslation('common');

/*Diseño de interfaz para las gráficas de resultados*/
 return (
   <Panel lg={12} xl={6} md={12} xs={12} title={t('Resultados      parciales')}>
     <div dir={dir}>
       <ResponsiveContainer className="dashboard__chart-
pie dashboard__chart-pie--commerce" height={360}>
         <PieChart className="dashboard__chart-pie-container">
           <Tooltip position={coordinates} {...getTooltipStyle
s(themeName)} />
           <Pie
             data={data01}
             dataKey="value"
             cy={180}
             innerRadius={130}
             outerRadius={160}
             label
             onMouseMove={onMouseMove}
           />
           <Legend layout="vertical" verticalAlign="bottom" wr
apperStyle={style(dir)} content={renderLegend} />
         </PieChart>
       </ResponsiveContainer>
     </div>
   </Panel>
 );
};
```

3.1. Módulo de recuperación de datos

Para recuperar los datos manuales se han generado interfaces gráficas utilizando las siguientes tecnologías: React Native, Visual Studio Code, JavaScript, CSS, PostgreSQL, PHP, MaterialX, Firebase, para las pruebas en el teléfono celular Samsung A10 se utilizó el Sistema Operativo Android.

A continuación, se presentan listados de código implementado en React Native. Es importante mencionar que una vez ingresada la información no es posible editarla, porque queda como historial médico y para consultas futuras.

3.2. Modelo de detección temprana

Para valorar el uso del modelo adecuado en la aplicación se siguió un proceso con los siguientes pasos: a) análisis de variables; b) selección de variables; c) selección de muestra; d) selección de modelos de predicción; e) pruebas de predicción en los modelos y; f) análisis de resultados. Para el primer paso se realizó un análisis de 22 variables que definen la probabilidad de padecer una enfermedad cardiovascular, de las cuales se valoraron con mayor importancia 13 datos.

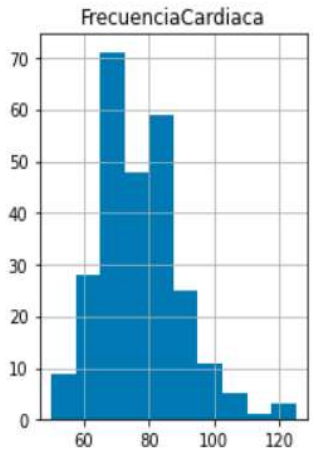


Fig. 2 Dispersión de datos sobre la frecuencia cardiaca.

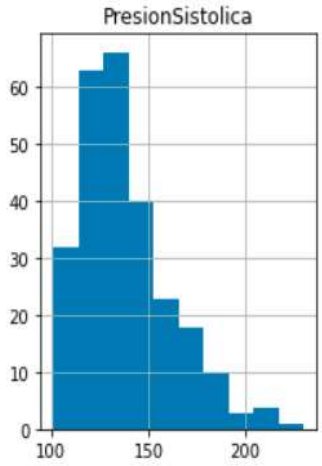


Fig. 3 Dispersión de datos sobre la presión sistólica.

Tabla 3. Rendimiento de modelos de predicción.

Modelo	Exactitud	Precisión	Recall	Puntuación Fl	Rocauc
Logistic Regression	67.95	**75.00**	29.03	41.86	61.32
Support Vector Classification	67.95	**80.00**	25.81	39.02	60.78
KNeighbors Classifier	66.67	**72.73**	25.81	38.10	59.71
Decision Tree Classifier	**74.36**	64.10	80.65	71.43	75.43
Random Forest Classifier	**69.23**	64.00	51.61	57.14	66.23
GradientBoosting Classifier	**69.23**	62.96	54.84	58.62	66.78
XGBRF Classifier	65.38	56.67	54.84	55.74	63.59
LGBM Classifier	65.38	57.69	48.39	52.63	62.49
CatBoost Classifier	**71.79**	69.57	51.61	59.26	68.36
MLP Classifier	62.82	60.00	19.35	29.27	55.42
AdaBoost Classifier	61.54	51.22	67.74	58.33	62.59

Durante la selección de variables se eligieron las que permitieran una valoración general de enfermedades cardiovasculares, que proporcionaran resultados parciales, así como un acercamiento a alguna ECV en particular.

En la Tabla 1 se describen cada una de las variables, el último dato es la variable clave para conocer el riesgo de una ECV. Además de los datos de la Tabla 1, también se recuperan datos de perfil del paciente: nombre completo, fecha de nacimiento, correo electrónico y ocupación.

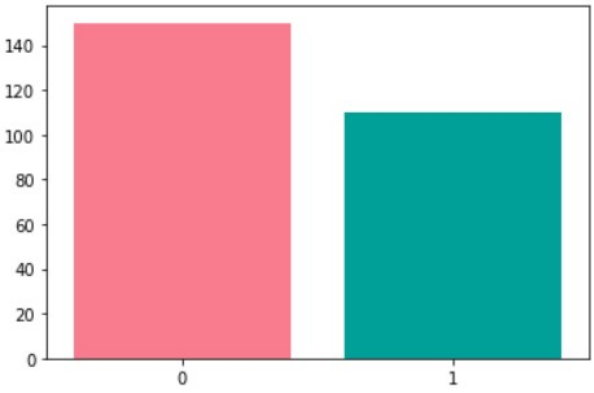


Fig. 4. Resultados del riesgo de alguna ECV.

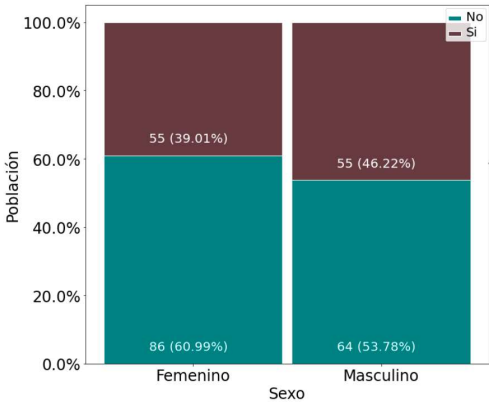


Fig. 5. Porcentaje de probabilidad de riesgo de acuerdo al sexo.

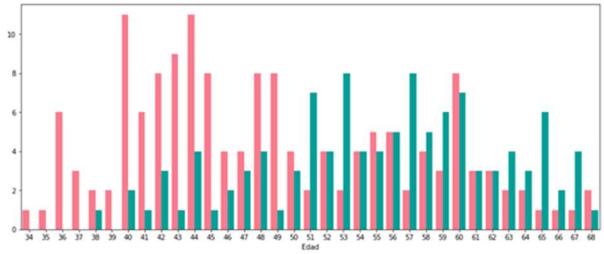


Fig. 6. Probabilidad de riesgo de acuerdo a la edad.

Con la finalidad de ofrecer personalización a los resultados finales de la detección temprana de ECV. Toda la información se agrupa en dos formas de extracción organizados en la Tabla 2. Para evaluar los modelos de predicción se tomó como muestra un grupo de datos de 260 personas, quienes otorgaron datos personales para la predicción de ECV, proporcionaron datos evaluados por médicos asignados e interactuaron con el *wearable* Fitbit Charge 4.

Fig. 7. Datos de la frecuencia cardiaca obtenido con el wearable.

Fig. 8. Detección de ECV persona 1.

Fig. 9. Detección de ECV persona 2**.**

Fig. 10. Datos ingresados.

Los datos se agrupan de acuerdo a las siguientes gráficas, la Fig. 2 muestra la diversidad que hay de los registros de frecuencia cardiaca, el dato que mayor repetición tiene es de 70 a 75 con alrededor de 72 personas. La Fig. 3 presenta los datos sobre la presión sistólica con mayor frecuencia de 125 a 130 con 65 personas y le continúa el rango de 120 a 125 con 62 personas.

Como siguiente paso fue elegir los modelos de predicción, las que tienen mayor confiabilidad y certeza, por ello se seleccionaron once: Logistic Regression, Support Vector Classification, KNeighbors Classifier, Decision Tree Classifier, Random Forest Classifier, GradientBoosting Classifier, XGBRF Classifier, LGBM Classifier, CatBoost Classifier, MLP Classifier y AdaBoost Classifier. Para probar los modelos se utilizaron las siguientes tecnologías: Anaconda, Python, Jupiter y datos en formato CSV.

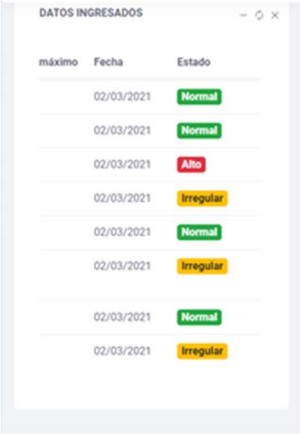


Fig. 11. Datos ingresados con alerta.

Fig. 12. Sugerencias para el cuidado de la salud.

Al momento de realizar las pruebas de predicción en los modelos se utilizaron los datos recuperados y la técnica de validación cruzada, el rendimiento generado y las mejores puntuaciones se presentan en la siguiente tabla.

Como se aprecia en la Tabla 3 la mayor exactitud lo brindan Decision Tree, CatBoost y en empate Random Forest y GradientBoosting; y la mayor precisión lo brindan Support Vector Classification, Logistic Regression y KNeighbors Classifier. Una vez analizado los resultados se aprecia~~r~~ en la Fig. 4 que no existe un riesgo de obtener una ECV en la mayoría de las personas y un resultado afirmativo en el menor número de la muestra.

De acuerdo al sexo en la Fig. 5 se percibe que las mujeres tienen un 60.99% de no padecer alguna ECV y un porcentaje de 39.01% de padecer alguna enfermedad. Los hombres tienen un 46.22% de tener alguna enfermedad y un 53.78% de no, por lo que la mayoría no tienen riesgo en el futuro si cuidan sus condiciones de salud.

Además de los valores generales, se creó una gráfica de acuerdo a la dispersión de edades para obtener una visualización sobre los riesgos que tiene la población en las ECV; en la Fig. 6 se aprecia que las personas de 40 a 44 años tienen menor probabilidad de obtener alguna enfermedad cardiovascular, mientras que personas en el rango de con 51 a 60 años tienen mayor posibilidad de padecer alguna ECV.

Con este análisis se puede definir que las personas mayormente susceptibles son los adultos mayores, a quienes se les debe tener un seguimiento a los factores de riesgo y mayor cuidado en las condiciones de salud para prevenir complicaciones futuras.

3.3. Módulo de despliegue de datos en dispositivos móviles

Para dar a conocer los resultados obtenidos en el tratamiento de datos, se desarrollaron interfaces gráficas de usuario utilizando las mismas tecnologías que se mencionaron en el apartado 3.1.

El uso de estas tecnologías permite al usuario interpretar la información de manera fácil para obtener el conocimiento necesario respecto a padecer alguna ECV. Al momento en que el usuario ingresa a la aplicación se presentan los resultados generales de las evaluaciones realizadas para detectar la probabilidad de una ECV, la monitorización que realiza el *wearable* utilizado, en la que se detecta la frecuencia cardiaca y la actividad física.

En la opción Gráfica general se puede observar el resultado general del análisis, pero es necesario dar a conocer a las personas un valor definido. En las gráficas parciales se visualizan las enfermedades más comunes que derivan de las ECV, como lo son la arritmia, hipertensión, angina de pecho y posibilidad de infarto.

La gráfica se desglosa de acuerdo a la información que se recuperó del paciente; en las Figuras 8 y 9 se presentan los resultados de dos personas diferentes, la primera presenta mayor porcentaje en contraer hipertensión, en cambio el segundo tiene una posibilidad de llegar a padecer alguna arritmia.

Además de las gráficas proporciona de manera tabular la información obtenida y se representan las irregularidades en la salud del paciente mediante colores utilizando un formato de tipo semáforo. La Fig. 10 muestra las alertas y el enfoque a esos datos, también se describe el valor máximo, permitiendo al paciente revisar los límites que sobrepasa. Mientras que la Fig. 11, muestra su estado de salud en formato de semáforo.

Para apoyar a la persona a mejorar sus condiciones de vida y reducir los riesgos y complicaciones de salud, se ofrecen sugerencias dentro de la aplicación (ver Fig. 12), las cuales han sido aportadas por médicos expertos en casos de prevención.

Las sugerencias buscan apoyar a distancia, pero no sustituyen a la asistencia personal con el médico especialista en diagnóstico, prevención y monitorización de ECVs. En este contexto se recomienda hacer el análisis de detección temprana a cierto tiempo para valorar los avances del paciente.

4. Conclusiones y trabajo a futuro

Para apoyar a la disminución de muertes a causas de enfermedades cardiacas, actualmente hay soluciones que se enfocan principalmente en personas que presentan síntomas o ya padecen alguna enfermedad y la solución es para dar seguimiento. Como se observa en este trabajo, el enfoque va hacia la detección temprana de ECV, sin importar si la persona no ha presentado algún síntoma. Para generar la detección de la ECV se seleccionaron datos que permiten conocer a la persona, su estado de salud, sus relaciones familiares y con apoyo del *wearable* se obtiene la información real y de manera automática sobre las condiciones cardiacas que presenta, por lo que se genera un resultado acertado.

Con las sugerencias al paciente se apoya a cambiar las actividades cotidianas, los hábitos y la alimentación, si el paciente sigue al pie las recomendaciones y realiza otro análisis se mostrará una reducción en los porcentajes. La aplicación para dispositivos móviles permite una interacción amigable y fácil de utilizar, con la finalidad de que el usuario disponga de él en los casos necesarios y comparta la información con médicos y familiares para su seguimiento. El trabajo con los tres módulos permitió el desarrollo de una solución viable a usuarios de cualquier edad y condición.

Como trabajo a futuro, se mejorará el método de extracción de datos utilizando técnicas de Web Scraping y de Procesamiento de Lenguaje Natural, así como la integración de otros *wearables* y sensores que proporcionen biomarcadores del estado de salud de los pacientes con el propósito de apoyarlo con la disminución de la captura manual de datos.

También se analizarán otros modelos para la predicción de ECVs que serán implementados dentro de la aplicación, para realizar una comparación detallada de los resultados que permita identificar los biomarcadores indispensables para detectar otras ECVs y clasificar los modelos de predicción que proporcionan mayor precisión para la detección temprana de ECVs más comunes en México, incluyendo arritmias, arteriopatía coronaria, insuficiencia cardiaca, valvulopatías cardiacas, miocardiopatías y pericarditis.

Agradecimientos. Este trabajo de investigación fue patrocinado por el Consejo de Investigaciones Científicas y Desarrollo Tecnológico de Veracruz (COVEICYDET). Los autores agradecen a COVEICYDET por apoyar este trabajo a través del proyecto Prevención y detección temprana de enfermedades cardiovasculares (arritmias & taquicardias) mediante técnicas de aprendizaje automático, Big Data e Internet de las Cosas con identificador 12 1806, así como al Consejo Nacional de Ciencia y Tecnología (CONACYT) y al TecNM-ITOrizaba.

Referencias

1. Navarro-Zarza, J., Tello-Divicino, T., Parra-Rojas, I., Zaragoza-García,O., Guzmán-Guzmán, I.: Detección de riesgo cardiovascular en trabajadores del sector salud con base en los criterios OMS/JNC 7/ATP III. Rev. Med. Inst. Mex. Seguro Soc., vol. 55, no. 3, pp. 300–308 (2017)
2. Sacco, R. L. Roth, G. A., Reddy, K. S., Arnett, D. K., Bonita, R., Gaziano, T. A., Heidenreich, P. A., Huffman, M. D., Mayosi, B. M., Mendis, S., Murray, C. J. L., Perel, P., Piñeiro, D. J., SmithJr, S. C., Taubert, K. A., Wood, D. A., Zhao, D., Zoghbi, W. A.: The heart of 25 by 25: Achieving the goal of reducing global and regional premature deaths from cardiovascular diseases and stroke: A modeling study from the American Heart Association and World Heart Federation. Circulation, vol. 133, no. 23, pp. e674–e690 (2016) doi: 10.1161/CIR.0000000000000395
3. Vickers, A. J., Van Calster, B., Steyerberg, E. W.: Net benefit approaches to the evaluation of prediction models, molecular markers, and diagnostic tests. vol. 352 (2016) doi: 10.1136/bmj.i6
4. Ogundimu, E. O., Altman, D. G., G., Collins, S.: Adequate sample size for developing prediction models is not simply related to events per variable. J. Clin. Epidemiol., vol. 76, pp. 175–182 (2016) doi: 0.1016/j.jclinepi.2016.02.031
5. Iftikhar, Z., Lahdenoja, O., Tadi, M. J., Hurnanen, T., Vasankari, T., Kiviniemi, T., Airaksinen, J., Koivisto, T., Pämkäälä, M.: Multiclass classifier based cardiovascular condition detection using smartphone mechanocardiography. Scientific Report, vol. 8, no. 1, pp. 9344 (2018) doi: 10.1038/s41598-018-27683-9
6. Moreno-Alsasua, L., Garcia-Zapirain, B., Rodrigo-Carbonero, J., David, I. Oliogordio-Ruiz, Hamrioui, S., de la Torre Díez, I.: Primary prevention of asymptomatic cardiovascular disease using physiological sensors connected to an iOS App. Journal of Medical Systems, vol. 41, no. 12, pp. 191 (2017) doi: 10.1007/s10916-017-0840-2

7. Prabhu, G., Kuklyte, J., Gualano, L., Venkataraman, K., Ahmadi, A., Duff, O., Walsh, D., Woods, C., O'Connor, N. E., Moran, K.: Design and development of the MedFit App: A mobile application for cardiovascular disease rehabilitation. In: Perego, P., Rahmani, A., TaheriNejad, N. (eds) Wireless Mobile Communication and Healthcare, MobiHealth 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 247, pp. 20–28 (2018) doi: 10.1007/978-3-319-98551-0_3
8. Lobelo, F., Kelli, H. M., Tejedor S. C., Pratt, M., McConnell, M. V., Martin, S. S.,Welk, G. J.,: The Wild Wild West: A framework to integrate mhealth software applications and wearables to support physical activity assessment. Counseling and Interventions for Cardiovascular Disease Risk Reduction, Prog. Cardiovasc. Dis., vol. 58, no. 6, pp. 584–594 (2016) doi: 10.1016/j.pcad.2016.02.007
9. De Pessemier, T., Cailliau, E., Martens, L.: Heart rate monitoring and activity recognition using wearables. In Proceedings of Sixth International Conference on Building and Exploring Web Based Environments, WEB´18 pp. 10–15 (2018)

ISSN 1870-4069

Sistema basado en inteligencia artificial para la identificación de cubrebocas y su correcto uso

Julio Cesar Elizalde-Silva, Carlos Avilés-Cruz,
Arturo Zúñiga-López

Universidad Autónoma Metropolitana,
Departamento de Electrónica, División de Ciencias Básicas e Ingeniería,
México

cesaruaeh@gmail.com,
{caviles, azl}@azc.uam.mx

Resumen. Derivado de la pandemia que se vive en México y en el mundo por el SARS-CoV-2 causante de la COVID-19, la Organización Mundial de la Salud (OMS) recomendó ampliamente el uso del cubreboca. La OMS puntualiza en el correcto uso del cubreboca para personas que se encuentran en lugares públicos, abiertos o que no puedan mantener como mínimo dos metros de distancia entre ellas. Por ende, surge la necesidad de desarrollar una herramienta computacional que permita identificar a personas en zonas públicas sin cubreboca o que no lo usen de manera correcta. El modelo que se ha propuesto en el presente trabajo utiliza técnicas de aprendizaje profundo tales como las redes neuronales convolucionales (RNC) así, como también algunos módulos de las siguientes herramientas de programación en Python: OpenCV, Scikit-learn, TensorFlow, Keras, etc. El modelo toma como entrada el video que se está transmitiendo en vivo desde una cámara, e identifica rostros de personas que no llevan puesto el cubreboca. Finalmente se muestra en la pantalla de la computadora una alarma visual, indicando que la persona no trae cubreboca o no lo usa de manera correcta. El sistema propuesto tiene una exactitud de 97 % de buen reconocimiento de cubrebocas.

Palabras clave: Covid-19, redes neuronales convolucionales, python, OpenCV, scikit-learn, tensorflow, keras, extracción de características, clasificación.

Artificial Intelligence Based System for the Identification of Mask and their Correct Use

Abstract. Derived from the pandemic that is experienced in Mexico and in the world by the SARS-CoV-2 that causes COVID-19, the World Health Organization (WHO) widely recommended the use of the mask. The WHO points out the correct use of the mask for people who are in public, open places or who cannot keep at least two meters of distance between them. Therefore, the need arises to develop a computational system that allows identifying people in public areas without a mask or who do not use it correctly.The model that has

been proposed in this work uses deep learning techniques such as convolutional neural networks (CNN) as well as some modules of the following Python programming tools: OpenCV, Scikit-learn, TensorFlow, Keras, etc. This work makes it possible to identify people with and without face masks, using CNN; consists of two main stages: a) Characteristics extraction; are the elements that describe an object (size, shape, color, texture, etc.) contain the greatest amount of information, b) Classification; At this stage is where the model learns from the training set, allowing it to distinguish between different images and thus be able to classify them from a visual representation of the same. This project can be implemented for security purposes in any hospital, clinic, school, company (public or private), etc., since its implementation is easy and efficient in terms of the resources it needs.

Keywords: Classification, content—based image retrieval, image retrieval, image classification, Wiener-Granger causality, convolutional neural network, OpenCV, scikit-learn, tensorFlow, keras.

1. Introducción

Las enfermedades virales son sumamente comunes, de hecho, son las principales razones por las cuales visitamos al médico. Según la OMS, las enfermedades respiratorias son aquellas que afectan directamente las vías nasales, los bronquios y los pulmones. La mayoría de las infecciones que causan los virus se pueden prevenir si se toma en cuenta una serie de recomendaciones pertinentes; lavarse las manos con suficiente agua y jabón, evitar el contacto directo con personas infectadas y el uso correcto del cubreboca, pueden ser algunas de estas medidas.

En diciembre del 2019 hubo un brote epidémico de neumonía de causa desconocida en Wuhan, China; las autoridades chinas confirmaron 41 casos detectados entre el 8 de diciembre de 2019 y el 2 de enero de 2020. La rápida expansión de una enfermedad desconocida, hasta ese momento, hizo que la OMS la declarara emergencia sanitaria de preocupación internacional. El 11 de marzo de 2020 la enfermedad se hallaba ya en más de 100 territorios a nivel mundial [15].

Para prevenir la expansión de dicha enfermedad los gobiernos han propuesto una serie de restricciones tales como: cancelación de viajes, cancelación de eventos, cierre de establecimientos, etc. La reciente enfermedad ha tenido un alto impacto en muchos países alrededor de todo el mundo; ha cambiado de manera impresionante la vida de las personas modificando las rutinas a las cuales estaban acostumbradas. Los profesionales de la salud, hospitales, clínicas, organizaciones sanitarias e investigadores han hecho un enorme esfuerzo para sacar una vacuna que pueda ayudar a superar está grave enfermedad.

A pesar de tener resultados exitosos, esto no es suficiente puesto que el virus se propaga a través del aire. Cuando una persona infectada habla o estornuda las gotas que salen de su boca o nariz se diseminan por el aire y así, afecta a otras personas [14], por lo que usar cubreboca puede ser una medida de gran importancia para disminuir el riesgo de ser contagiado.

La inteligencia artificial (IA) comprende un amplio conjunto de algoritmos, y su efectividad está estrechamente relacionada con la calidad de la información de la cual se aprende [16]. El aprendizaje profundo es uno de los campos de la IA, que más ha despertado el interés en científicos e investigadores debido a su enorme campo de aplicación [17]. La detección de objetos en imágenes digitales es quizá la tarea más importante del aprendizaje profundo y, la técnica más utilizada para llevar a cabo dicha terea son las RNC que han logrado resultados prometedores.

Con el enorme desarrollo que han tenido estas técnicas, la detección de rostros en imágenes digitales, parece ser un problema que se ha solventado de manera correcta [3], sin embargo, la detección de rostros humanos con objetos tales como: lentes, pasamontañas, gorras, sombreros, cubreboca etc., hacen que los detectores de rostros sean cada vez más complejos y que su implementación, no proporcione los resultados esperados.

En el presente trabajo hemos propuesto un sistema que identifica a personas que no portan cubreboca o que no lo usan de manera correcta. El sistema hace uso de un modelo de aprendizaje profundo, el cual está compuesto de una RNC de dos capas internas, una capa que recibe todos los atributos obtenidos de las capas anteriores, una capa interna que evita el sobre entrenamiento del modelo, una capa oculta compuesta por 50 valores de salida y finalmente una capa de salida con dos clases.

El trabajo propuesto en este artículo también hace uso de un método de la librería OpenCV que se llama "detectMultiScale" de la interfaz "CascadeClassifier" la cual nos permite identificar el rostro de una o más personas de manera frontal. Este método permite obtener un rectángulo delimitador, donde se encuentra la cara de la (s) persona (s) en la imagen. Para entrenar el modelo se utilizaron 1376 imágenes de rostros humanos; de las cuales 690 imágenes son de personas que traen puesto el cubreboca y 686 imágenes son de personas que no traen cubreboca.

Para reducir el costo computacional que implica el preprocesamiento de la información, las imágenes fueron convertidas a escala de grises, así, como también se redujo el tamaño de estas quedando de 32×32. Finalmente, las imágenes en su representación matricial fueron normalizadas para comprimir aún más los datos que entran al modelo. Los datos obtenidos han sido divididos con el método "train_test_split" de la librería Scikit-learn, utilizando el $80\,\%$ para entrenamiento, $20\,\%$ para prueba y dejando la configuración del método de manera aleatoria. La exactitud del modelo es de $97\,\%$ con 20 épocas.

El resto del trabajo está organizado de la siguiente manera: en la sección 2, se describen los trabajos que se han llevado a cabo con respecto a esta área de investigación, en la sección 3, se explica la metodología propuesta que se llevó a cabo, en la sección 4, se muestran los resultados obtenidos. Finalmente, en la sección 5, se ponen las conclusiones y el trabajo futuro.

2. Trabajos relacionados

Los algoritmos convencionales de detección de objetos fueron principalmente derivados del aprendizaje automático; siendo este una disciplina que construye modelos matemáticos y algoritmos para realizar tareas específicas utilizando computadoras [12].

Los científicos de la computación han desarrollado numerosas arquitecturas que ayudan a identificar y clasificar objetos en una imagen digital, utilizando técnicas de aprendizaje profundo. El aprendizaje profundo ha mostrado un enorme potencial en muchas aplicaciones en la vida científica y práctica, la detección de rostros, texto, logotipos, video, vehículos, imágenes médicas; son algunas de estas aplicaciones [11].

Proporciona un grupo de algoritmos que se pueden emplear para problemas de aprendizaje supervisado, no supervisado y/o reforzado, sobre cualquier tipo de datos, señales, imágenes digitales, videos, etc.

La idea fundamental del aprendizaje profundo proviene de neuronas artificiales, el cual es un modelo de aprendizaje jerárquico muy robusto que permite aprender representaciones complejas directamente de los datos de entrada, el aprendizaje profundo ha ido de la mano de las RNC obteniendo resultados prometedores.

En esencia la identificación y clasificación de objetos es utilizar cuadros rectangulares delimitadores para localizar los objetos en la imagen [20]. Se compone principalmente de cuatro etapas:

1. Identificación de objetos; obtiene la categoría a la que pertenecen los objetos en la imagen.
2. Detección de objetos; ubica los objetos con cuadros delimitadores rectangulares.
3. Segmentación semántica; predice las categorías de cada píxel.
4. Segmentación de instancias; necesita predecir tanto las categorías de cada píxel como de cada objeto [13].

Las RNC han tenido una evolución significativa en la clasificación de imágenes[19] siendo el tipo de aprendizaje supervisado preferido para problemas de visión por computadora. Sin embargo, diseñar mejores arquitecturas de RNC sigue siendo una pregunta inicial. A continuación, se muestra una breve descripción de los algoritmos más utilizados de RNC.

– Detección de objetos: Probablemente es la tarea más importante de la visión por computadora. Esta técnica utiliza una ventana de tamaño MxN que selecciona el objeto en la imagen, los objetos ubicados son encerrados en cuadros delimitadores [7]. Inicialmente, un clasificador se prepara con un conjunto de datos de entrenamiento (imágenes), si encuentra el objeto de interés, lo marca como una imagen positiva, de lo contrario como una imagen negativa.
– Faster R-CNN: Comprende dos módulos, la red de propuesta de región (RPN), en la que se distingue la región del objeto en la imagen y una red que permite clasificar los objetos en la región propuesta [1]. Faster R-CNN es considerada una de las técnicas de detección de objetos más precisas.
– YOLO – You Only Look Once: Una sola RNC predice simultáneamente múltiples cuadros delimitadores y probabilidades de clase para detectar objetos en imágenes [18]. Está técnica divide la imagen en regiones, poniendo cuadros de identificación y probabilidades por cada región. Los cuadros son ponderados a partir de las probabilidades predichas y utiliza sus funciones para predecir cada cuadro delimitador [6].

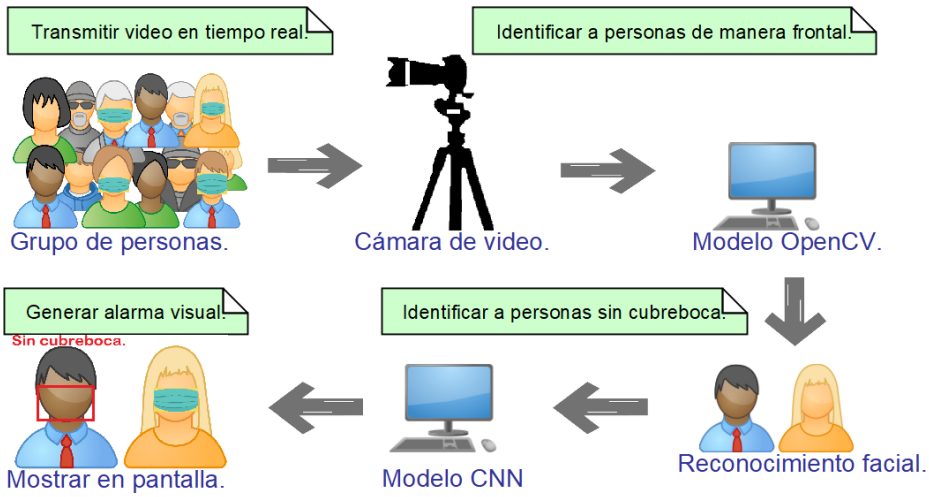


Fig. 1. Metodología general de sistema de identificación de cubreboca.

– Single shot detector (SSD): Encuentra numerosos objetos en la imagen con tan solo una muestra. La RNC se ejecuta una sola vez obteniendo el mapa de características de la imagen. Utiliza rectángulos delimitadores para predecir objetos que tienen diferente tamaño, la RNC combina n predicciones de mapas de características con diferentes resoluciones. El núcleo de SSD es predecir puntuaciones de cada categoría para un conjunto fijo de cuadros delimitadores predeterminados que utilizan pequeños filtros convolucionales aplicados a los mapas de características [10].

– RetinaNet: Es una técnica que está compuesta de dos arquitecturas: YOLO y SSD, cuenta con redes piramidales para la detección de objetos. RetinaNet es un modelo unificado, compuesto por una red que utiliza como columna vertebral y dos subredes específicas de tareas. La red que utiliza como columna vertebral se encarga de obtener las características de la imagen de entrada. La primera subred realiza la clasificación de objetos. La segunda subred crea un cuadro rectangular delimitador, utilizando la salida de la red troncal [5].

Como se ha mencionado anteriormente las RNC han demostrado gran capacidad para resolver problemas de clasificación de imágenes, este aprendizaje se obtiene gracias a la enorme cantidad de información (millones de parámetros) que utilizan las redes para su entrenamiento. Por otro lado, el reconocimiento facial representa un enorme campo de aplicación para estos modelos. La detección de rostros se refiere al uso de un método para determinar la posición del rostro en cualquier imagen o video [21].

Los detectores de rostros modernos pueden identificar fácilmente el rostro de una persona de manera frontal[9]. Sin embargo, la precisión de la detección de rostros puede verse afectada por varios factores ajenos al modelo, ejemplo: iluminación desigual, elementos extraños, los rostros están de perfil, ángulos complicados, oclusión, etc. La investigación sobre detección y reconocimiento facial se ha estudiado ampliamente en los últimos años.

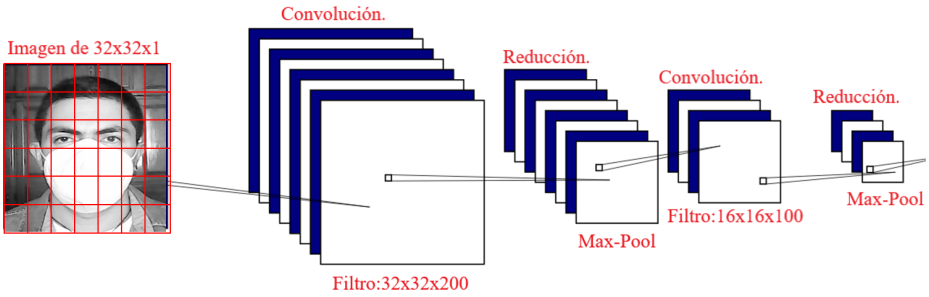


Fig. 2. Red neuronal convolucional.

Las aplicaciones de reconocimiento facial juegan un papel importante en muchas áreas como la seguridad, vigilancia con cámaras, verificación de identidad en dispositivos electrónicos modernos, investigaciones criminales, sistemas de gestión de bases de datos, aplicaciones de tarjetas inteligentes, etc. [2] Algunas de las ventajas que tienen los sistemas de reconocimiento facial son: utilizan medios no invasivos, pueden encontrar a una persona dentro de una gran base de datos, identificar a personas concretas en tiempo real, etc. [8].

El objetivo principal del reconocimiento facial es autenticar e identificar los rasgos faciales de una persona. Sin embargo, los rasgos faciales se capturan en tiempo real y son procesados por un modelo previamente entrenado, las RNC han demostrado ser un enfoque altamente eficiente para esta tarea[4]; para lograr esta precisión óptima el conjunto de datos de entrenamiento es elevado.

3. Sistema propuesto

La metodología general del sistema propuesto se muestra en la figura 1. La propuesta inicia con: a) La captura de video en tiempo real a través de una cámara óptica, b) Identificación de rostros de manera frontal, c) Red neuronal convolucional, y d) Mostrar resultados en pantalla. Los módulos son descritos a continuación:

Etapa de entrenamiento

– **Aumento de datos:**
El proyecto está compuesto por el módulo “DataAugmentation”, encargado de aumentar los datos que son utilizados para el entrenamiento del modelo, las imágenes son leídas de las carpetas “mask” y “without_mask”. La clase se llama “ImageDataGenerator” e implementa el método “flow”, su tarea principal consiste en cambiar las propiedades de cada imagen obtenida, generar una nueva imagen y finalmente guardar las imágenes generadas en las mismas carpetas de donde se obtuvieron.

– **Preprocesamiento de la información:**
Viene la etapa de preprocesamiento de la información, es decir las imágenes que fueron generadas ahora van a hacer los insumos de nuestro modelo. La clase encargada de llevar a cabo esta tarea se llama “DataIn” y su tarea inicial es leer

las imágenes de los directorios "mask" y "without_mask", convertirlas a escala de grises y así poder obtener un solo canal que va desde 0 hasta 255. Después de haber realizado este primer paso, también se ha disminuido el tamaño de cada imagen, quedando de la siguiente manera: $32 \times 32 \times 1$, lo cual equivale a tener una matriz de 1024 valores (pixeles) o lo que es lo mismo tener una matriz de 32 filas por 32 columnas. Los datos han sido normalizados entre 0 y 1 para trabajar con valores más pequeños. Finalmente usando el método "to_categorical" de Keras obtenemos una matriz binaria etiquetando con 0 a las personas que traen puesto el cubreboca y con 1 a personas sin cubreboca.

– **Imágenes de entrenamiento:**
El módulo central del proyecto se llama "Classify" y su tarea inicial consiste en obtener los datos de entrenamiento y prueba que ocupa el modelo. Como se mencionó anteriormente los datos están divididos en dos partes: $80\,\%$ para entrenamiento y $20\,\%$ para prueba; esta tarea se llevó a cabo haciendo uso del método "train_test_split" del api de Scikit-learn y dejando la configuración del método de manera aleatoria. Para la extracción de características se ha utilizado una RNC secuencial con dos capas internas. El diagrama se puede ver en la figura [2].

– **Parámetros de entrenamiento de la RNC:**
Como se muestra en el diagrama la RNC toma como entrada una imagen de 32×32 de un solo canal (escala de grises), para el proceso de convolución se han utilizado 200 filtros de 3×3 y una función de activación ReLu, finalmente para obtener la información más representativa se configuro un submuestreo al máximo de 2×2 (Max-Pool) y con esto se da por terminada la configuración de la capa 1.

Para el proceso de convolución de la capa 2, se tomaron 100 filtros de 3×3, un "padding same" para no reducir el tamaño de la imagen y una función de activación ReLu. Para obtener la información más representativa de las imágenes se utilizó un Max-Pool de 2×2, así, como un stride de 2×2 que minimiza aun más el tamaño de las imágenes.

– **Etapa de clasificación de la RNC:**
Para la etapa de clasificación, se cuenta con una capa completamente conectada, por sus siglas en ingles "Full-connected layer" que recibe todos los atributos extraídos de las capas anteriores y se colocan en un vector Nx1, donde N es igual al número de atributos; en nuestro caso igual a $6,400$. Para evitar el sobre entrenamiento del modelo (overfitting), utilizamos una capa de regularización (Dropout), inhabilitando el $20\,\%$ de los parámetros.

El modelo cuenta con dos capas internas de salida; la primera está configurada para tomar 50 valores o lo que es lo mismo tener 50 clases diferentes, además de una función de activación ReLu; en la segunda capa el número de valores ha sido reducido, quedando el modelo con 2 clases, se utilizó una función de activación "Softmax", con la cual forzamos a tener una sola clase ganadora, es decir la clase que tenga mayor probabilidad entre 0 y 1 es la clase que gana. En la figura [3] se muestra el diagrama de la capa "Full-connected layer".

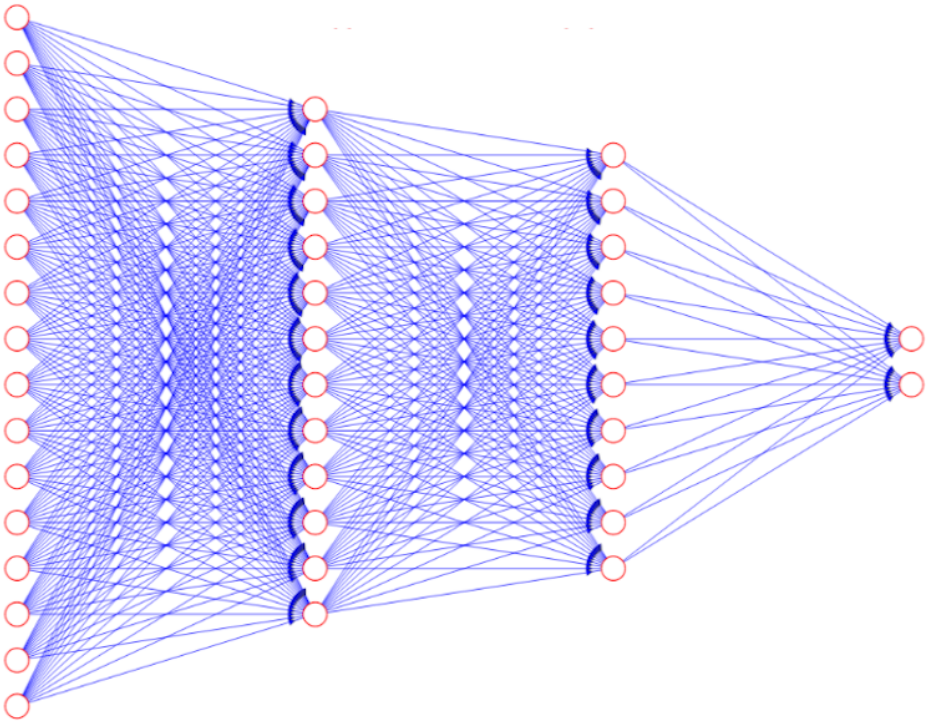


Fig. 3. Capa completamente conectada.

Para optimizar el costo del modelo se utilizó la función del descenso del gradiente estocástico “adam”, dejando su configuración por default, así como la medida de precisión “entropía cruzada binaria”, utilizada para ajustar los pesos del modelo durante el entrenamiento. Finalmente, el modelo ha sido entrenado, obteniendo el $97\,\%$ de exactitud con 20 épocas.

El ultimo módulo del sistema se llama “DataOut”, su tarea inicial consiste en cargar en el disco de la computadora el modelo obtenido en el paso anterior, posteriormente se utiliza la interfaz “VideoCapture” del api de OpenCV, la cual toma el control de la cámara y la entrada del modelo es el resultado de la interfaz “VideoCapture”, la cual consta de un video en escala de grises que se transmite en vivo desde la cámara interna de la computadora.

– **Detección de rostros en la imagen:**
Para detectar el rostro de una persona, se utiliza el método “detectMultiScale” de la interfaz “CascadeClassifier”, utilizando un rectángulo delimitador de color verde donde se encuentra el rostro de la (s) persona (s) identificadas en el video. Finalmente se muestra en la pantalla de la computadora una alarma visual, indicando que la persona no trae puesto el cubreboca. Para finalizar, el sistema libera los recursos de la computadora, es decir, destruye la ventana emergente que aparece en la pantalla y libera la cámara de la computadora.

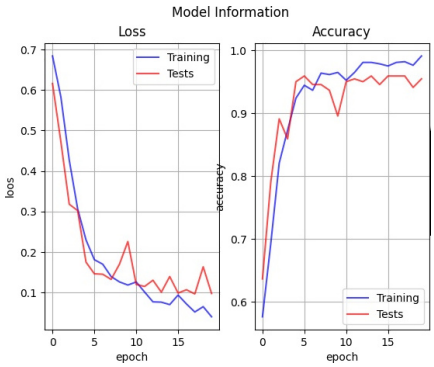


Fig. 4. Perdida y exactitud.

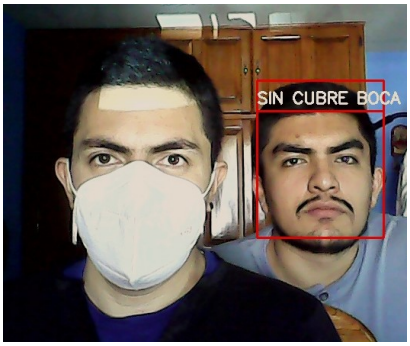


Fig. 5. Personas capturadas en tiempo real.

4. Resultados

Para probar la metodología propuesta, así como evaluar su eficiencia, se utilizó la base de imágenes del repositorio[1]. La base de imágenes esta conformada de $1,376$ imágenes, de las cuales 690 son de rostros de personas con cubreboca y 686 son de rostros de personas sin cubreboca. El trabajo se realizó sobre el lenguaje de programación Python, utilizando además las principales API's de aprendizaje automático (TensorFlow, Keras y Scikit-learn) que ayudan a la implementación del modelo propuesto, así como también OpenCV que permite la manipulación de imágenes e implementación de la interfaz "CascadeClassifier" para detectar el rostro de las personas de manera frontal.

El modelo propuesto en el presente trabajo, fue entrenado con 20 épocas. La figura 4 muestra las gráficas de pérdida y exactitud respectivamente. Como se puede apreciar en dicha gráfica, el error (gráfica izquierda) de entrenamiento es menor al $0{,}5\,\%$ (ver trazo azul). Por otra parte, la precisión alcanza un desempeño del $98\,\%$ (ver trazo azul, de la gráfica izquierda). En cuanto al desempeño global, usando toda la base de imágenes, se logra una exactitud del $97\,\%$. En la figura 5 se muestra un ejemplo de la detección correcta del cubreboca, así como la detección de la persona que no porta el cubreboca. Como se puede apreciar en la figura 5, la persona del lado derecho no porta el cubreboca, indicándolo en un recuadro rojo y con el texto respectivo.

5. Conclusiones y perspectivas

El sistema que se ha desarrollado en el presente trabajo es una medida de seguridad de las muchas que recomienda la OMS para disminuir el riesgo de contagio de la COVID-19, sirve para identificar a personas en tiempo real que no traen cubreboca o que no lo usan de manera correcta, esta herramienta podría permitir automatizar el proceso que se lleva a cabo en establecimientos públicos, donde hay una persona encargada de revisar que las personas que quieren ingresar al lugar lleven puesto su cubreboca o que lo usen de manera correcta.

[1] https://github.com/prajnasb/observations/tree/master/experiements/data

El sistema propuesto proporciona una eficiencia del 97 %, presentando un recuadro rojo en pantalla con el mensaje "sin cubreboca" cuando la persona no lleva puesto su cubreboca o no lo use de manera correcta. Una ventaja del sistema es que trabaja en tiempo real (procesar al menos 24 imágenes por segundo) en una computadora estándar.

Una limitante del sistema propuesto es que solo sirve para detectar a personas de manera frontal. Para trabajo futuro se ha pensado en complementar el sistema; el cual permitirá identificar el rostro de una persona, no solo de manera frontal, sino también de perfil, además de detectar la temperatura corporal utilizando una cámara termográfica.

Referencias

1. Ajitha, S., Judy, M. V.: Faster R-CNN classification for the recognition of glaucoma. Paper presented at the Journal of Physics: Conference Series, vol. 1706, no. 1, pp. 1–12 (2020) doi: 10.1088/1742-6596/1706/1/012170
2. Asif Hussain, S., Al Balushi, A. S. A.: A real time face emotion classification and recognition using deep learning model. Journal of Physics: Conference Series, vol. 1432, no. 1, pp. 14 (2020) doi: 10.1088/1742-6596/1432/1/012087
3. Basha, C. Z., Pravallika, B. N. L., Shankar, E. B.: An efficient face mask detector with pytorch and deep learning. European Alliance for Innovation, Endorsed Transactions on Pervasive Health and Technology, vol. 7, no. 25, pp. 1–8 (2021) doi: 10.4108/eai.8-1-2021.167843
4. Ben-Fredj, H., Bouguezzi, S., Souani, C.: Face recognition in unconstrained environment with cnn. Visual Computer, vol. 37, pp. 217–226 (2021) doi: 10.1007/s00371-020-01794-9
5. Dhillon, A., Verma, G. K.: Convolutional neural network: A review of models, methodologies and applications to object detection. Progress in Artificial Intelligence, vol. 9, no. 2, pp. 85–112 (2020) doi: 10.1007/s13748-019-00203-0
6. Inthiyaz, S., Ahammad, S. H., Sai Krishna, A., Bhargavi, V., Govardhan, D., Rajesh, V.: YOLO (you only look once) making object detection work in medical imaging on convolution detection system. International Journal of Pharmaceutical Research, vol. 12, no. 2, pp. 312–326 (2020) doi: 10.31838/ijpr/2020.12.02.0003
7. Kang, K., Ouyang, W., Li, H., Wang, X.: Object detection from video tubelets with convolutional neural networks. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 817–825 (2016) doi: 10.1109/CVPR.2016.95
8. Lawrence, S., Giles, C. L., Tsoi, A. C., Back, A. D.: Face recognition: A convolutional neural-network approach. IEEE Transactions on Neural Networks, vol. 8, no. 1, pp. 98–113 (1997) doi: 10.1109/72.554195
9. Li, H., Lin, Z., Shen, X., Brandt, J., Hua, G.: A convolutional neural network cascade for face detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5325–5334 (2015) doi: 10.1109/CVPR.2015.7299170
10. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C.: SSD: Single shot multibox detector. Lecture Notes in Computer Science, pp. 21–37 (2016) doi: 10.1007/978-3-319-46448-0_2
11. Loey, M., Manogaran, G., Taha Mohamed, H. N., Nour Eldeen, M. K.: Fighting against COVID-19: A novel deep learning model based on yolo-v2 with resnet-50 for medical face mask detection. Sustainable Cities and Society, vol. 65, pp. 1–8 (2021) doi: 10.1016/j.scs.2020.102600
12. Masita, K. L., Hasan, A. N., Shongwe, T.: Deep learning in object detection: A review. In: Proceedings of the International Conference on Artificial Intelligence, Big Data, Computing

and Data Communication Systems (icABCD), pp. 11 (2020) doi: 10.1109/icABCD49160.2020.9183866
13. Meenpal, T., Balakrishnan, A., Verma, A.: Facial mask detection using semantic segmentation. In: Proceedigns of the 4th International Conference on Computing, Communications and Security, pp. 1–5 (2019) doi: 10.1109/CCCS.2019.8888092
14. Megahed, N. A., Ghoneim, E. M.: Antivirus-built environment: Lessons learned from COVID-19 pandemic. Sustainable Cities and Society, vol. 61, pp. 23 (2020) doi: 10.1016/j.scs.2020.102350
15. Nagrath, P., Jain, R., Madan, A., Arora, R., Kataria, P., Hemanth, J.: SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. Sustainable Cities and Society, vol. 66, pp. 11 (2021) doi: 10.1016/j.scs.2020.102692
16. Peter Norvig, S. J. R.: Artificial Intelligence: A Modern Approach (1995)
17. Ponce-Gallegos, J. C., Torres-Soto, A., Quezada-Aguilera, F. S., Silva-Sprock, A., Martínez-Flor, E. U., Casali, A., Scheihing, E., Túpac-Valdivia, Y. J., Torres-Soto, M. D., Ornelas-Zapata, F. J., Hernández, J. A., Zavala, C., Vakhnia, N., Pedreño, O.: Inteligencia Artificial. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIn) (2014) doi: 10.13140/2.1.3720.0960
18. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016), doi: 10.1109/CVPR.2016.91
19. Saca, F., Ramírez, A., Cruz, C., Villegas Cortez, J.: Red neuronal convolucional con extracción de características multi-columna para clasificación de imágenes. Research in Computing Science, vol. 148, no. 7, pp. 391–404 (2019) doi: 10.13053/rcs-148-7-29
20. Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., Lan, X.: A review of object detection based on deep learning. Multimedia Tools and Applications, vol. 79, no. 34, pp. 23729–23791 (2020) doi: 10.1007/s11042-020-08976-6
21. Yan, H., Wang, X., Liu, Y., Zhang, Y., Li, H.: A new face detection method based on faster RCNN. Journal of Physics: Conference Series, vol. 1754, pp. 1–6 (2021) doi: 10.1088/1742-6596/1754/1/012209

ISSN 1870-4069

Reducción y clasificación de una base de datos de audio mediante redes neuronales artificiales y minería de datos para el diagnóstico de pacientes con enfermedad De Parkinson

Luis Alberto Hernández Montiel, Jesús Velázquez Vásquez, Carlos Edgardo Cruz Pérez

Universidad del Istmo Campus Ixtepec,
Departamento de informática,
México

{hmontiel, carloscruz}@bianni.unistmo.edu.mx;
chuyvelazquez@hotmail.com

Resumen. En este artículo, se propone un método para reducción y clasificación de audios de personas con la enfermedad de Parkinson. Primero se genera una preselección de las mejores señales utilizando un método de filtro de datos. Después, un esquema de clasificación es implementado utilizado una red neuronal artificial. El sistema busca clasificar una señal de audio para generar un posible diagnóstico de la enfermedad de Parkinson. Los resultados obtenidos se comparan con diferentes resultados de métodos reportados en la literatura.

Palabras clave: Bioseñales, enfermedad de Parkinson, red neuronal artificial, preprocesamiento, clasificación.

Reduction and Classification of an Audio Database Using Artificial Neural Network and Data Mining for Diagnosis of Patients with Parkinson's Disease

Abstract. In this paper, a method for reduction and classifying audios from people with Parkinson's disease is proposed. First, a pre-selection of the best signals is generated using one data filter method. Then, a classification scheme is implemented using an artificial neural network. The system seeks to classify an audio signal to generate a possible diagnosis of Parkinson's disease. The results obtained are compared with different results of methods reported in the literature.

Keywords: Biosignals, Parkinson's disease, artificial neuronal networks, preprocessing, classification.

1. Introducción

Una señal es un medio de transmisión que contiene información sobre la fuente que la generó. En el caso de las bioseñales, la fuente de información son los diferentes sistemas fisiológicos de un organismo. Su captación permite al biólogo extraer información sobre el funcionamiento de los diferentes órganos para emitir un diagnóstico y/o pronóstico de alguna enfermedad como el Parkinson [1]. Sin embargo, las bioseñales son difíciles de estudiar, su característica principal es una alta dimensión debido a que el número de frecuencias emitidas por el organismo son considerablemente mayor (usualmente miles), en comparación con la cantidad de muestras analizadas (usualmente menos de 100) [2].

Además, los audios almacenados combinan información relevante de personas que padece la enfermedad de Parkinson (EP) con señales ruidosas y redundantes, lo que ocasiona que su estudio sea complejo y los resultados sean erróneos y no sirvan para generar un diagnóstico efectivo.

Debido a esto, diferentes trabajos en la literatura proponen métodos novedosos para la selección y clasificación de señales que ayuden en el diagnóstico de la enfermedad de Parkinson, por ejemplo, el trabajo de Kit Pun [3], propone un método basado en el diagnóstico estadístico y prueba de diagnóstico precisa, el resultado proporciona una precisión del 90% para diagnosticar pacientes con EP.

Kostas M. Tsiouris [4] empleó técnicas de minería de datos para mejorar el rendimiento y la toma de decisión para la conversión de la escala de calificación de la enfermedad de Parkinson, los resultados estiman una precisión en la etapa clasificación de 87% de aciertos para diagnosticar síntomas de la enfermedad de Parkinson. Geeta Yadav [5] formula tres métodos para la clasificación de síntomas de la EP.

Los métodos son: árbol clasificador, clasificador estadístico y una máquina de vectores de soporte, el rendimiento de estos tres clasificadores se mide con tres matrices: precisión, sensibilidad y especificidad, la principal tarea de este documento es averiguar qué modelo identifica mejor a las personas afectadas por enfermedad de Parkinson. En el trabajo Spielman, & Rami [6] se emplearon 4 algoritmos diferentes para a selección de los audios, los cuáles son: LASSO, mRMR, RELIEF, LLBFS. Los conjuntos de características obtenidos se clasificaron con una SVM y un clasificador bosques aleatorios.

Los resultados fueron validados con un 10-fold croosvalidation y una iteración de 100 veces consiguiendo aproximadamente un 99 % de tasa de éxito. A pesar del número de métodos que se han implementado, aun no se tiene una respuesta exacta al momento de clasificar frecuencias que ayuden en el diagnóstico de la enfermedad de Parkinson, lo que conlleva a proponer nuevos modelos basados en técnicas especializadas para generar un mejor estudio de los audios de personas con enfermedad de Parkinson.

En este documento, se propone un modelo híbrido basado en métodos de filtro combinada con una red neuronal artificial, para la selección y clasificación de señales de audio de personas con la enfermedad de Parkinson.

El método se ha divido en dos fases, en la fase uno se realiza una primera reducción del tamaño de la base de datos utilizando un método estadístico para el filtrado de las señales.

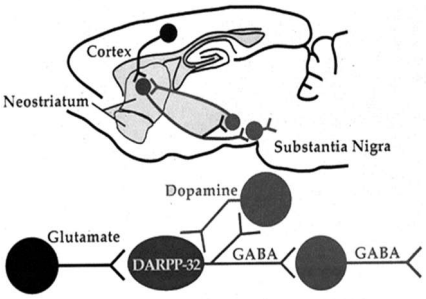


Fig. 1. Conexiones entre corteza, neostriatum y sustancia negra. DARPP-32 ([7]).

En la fase dos, se clasifica una señal de audio utilizando una red neuronal artificial. Con la combinación de estas técnicas, se buscan las mejores señales de audio que ayuden en el diagnóstico y/o pronóstico de pacientes con EP.

2. Enfermedad de Parkinson

La Enfermedad de Parkinson es una enfermedad neurodegenerativa crónica caracterizada por bradicinesia (movimiento lento), rigidez (aumento del tono muscular), temblor y pérdida del control postural. Se crea debido a una inestabilidad postural combinada con caídas recurrentes, generan pérdidas de células en ciertas partes del cerebro, especialmente en la sustancia negra la cual es una porción heterogénea del mesencéfalo, una región específica del cerebro, concretamente, constituye una porción dorsal del cerebro que se caracteriza por poseer neuronas que contienen neuromelanina, un pigmento oscuro específico del cerebro (véase Figura.1) [7]. La pérdida de estas células conduce a una escasez de dopamina en el cerebro lo que conduce a los síntomas de la EP.

3. Bioseñales

Los potenciales bioeléctricos de los organismos son el resultado de una actividad electroquímica de un determinado tipo de células, conocidas como células excitables; las cuales componen tejidos musculares, nerviosos y glandulares [8]. Otra fuente para obtener señales es a partir de los sonidos que emite un organismo. Captar las vibraciones que se producen en diferentes frecuencias ayuda al diagnóstico o pronóstico de algunas enfermedades [9].

Las señales de audio generadas por una persona, se usan como una herramienta de cobertura simple con un enfoque en seleccionar voces que exhiben diferentes tipos de frecuencia. El estudio de bioseñales, fomenta el desarrollo de nuevos algoritmos de selección de características, que originalmente, se habían diseñado para bases de datos provenientes de otras fuentes. Ahora, con varios miles de señales de audio, la extracción del mejor subconjunto de audios relevantes es un proceso computacionalmente viable.

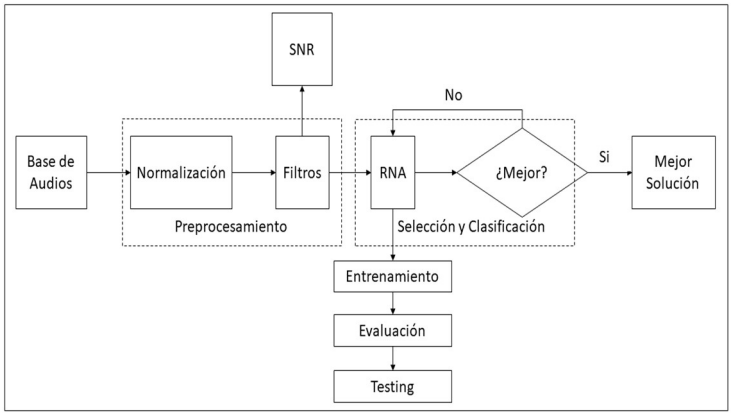


Fig. 2. Proceso general de selección y clasificación de señales de audio de pacientes con EP con el modelo propuesto.

Probando y generando modelos novedosos que utilizan diferentes técnicas de minería de datos, aprendizaje máquina y/o procesamiento de señales con el objetivo de encontrar las mejores bioseñales que ayude a la predicción de una enfermedad.

4. Selección y clasificación de audios de pacientes con enfermedad de Parkinson

La selección y clasificación de señales de audio para identificar si existe o no la posibilidad de que el paciente presente la EP es un problema que se intenta resolver de diferentes formas.

En este trabajo se emplea un modelo híbrido que se ha creado para seleccionar un conjunto de señales de audio dentro de una base de datos de pacientes con EP, el modelo propuesto está dividido en dos etapas (Figura. 2), donde cada etapa del modelo cumple una fase de selección y clasificación de los audios, las dos etapas se describen a continuación.

4.1. Preprocesamiento de datos

Esta etapa se divide por dos pasos. En el primer paso se utiliza una normalización de las señales, utilizando el método min/máx. [10], con el objetivo de tener los audios en un rango entre cero y uno para mejorar su clasificación y evitar un sobre entrenamiento.

4.2. Normalización min/máx

Las frecuencias vocales contenidas en los audios de personas con EP, se encuentra en diferentes escalas numéricas, esto podría generar un sobrentrenamiento al clasificar.

Al entrenar con datos dispersos, el algoritmo puede confundir una variable ruidosa y clasificarla como una relevante.

Para solucionar este problema se genera una transformación de las frecuencias a un rango entre cero y uno para facilitar su estudio. En este trabajo como primer paso del preprocesamiento se realiza una normalización basada en una técnica min-máx. [10]:

$$X' = \frac{X - min(X)}{max(X) - min(X)}, \quad (1)$$

donde: *X* es la base de datos de audios de pacientes con EP, *min(X)* es el valor mínimo dentro de la base de datos de audios, *max(X)* es el valor máximo dentro de la base de datos de audios de Parkinson usada.

Después de normalizar la base de datos, el paso dos de esta etapa es hacer una primera selección de señales efectivas utilizando un método estadístico de filtrado de datos. Los filtros funcionan en una etapa de preprocesamiento, su función principal es limpiar o seleccionar subconjuntos de características tomando cada variable individualmente y calcular una medida de puntuación para utilizarla posteriormente como indicador discriminatorio para descartar o filtrar las características redundantes o irrelevantes [11].

En este estudio se utiliza un filtro basado en la Relación Señal Ruido (SNR) [12]. Este filtro se utiliza por sus capacidades estadísticas, ya que prioriza una señal de audio en particular, evaluando su comportamiento dentro de la base de datos, colocando las señales más estables en las primeras posiciones y las más ruidosas al final. La forma de cómo trabaja el filtro de describe a continuación:

4.3. Relación señal a ruido (SNR)

Este filtro identifica los patrones de expresión con una diferencia máxima en la expresión media entre dos grupos y la variación mínima de expresión dentro de cada grupo. En este método, los audios se clasifican de acuerdo a sus niveles de expresión [12]:

$$SNR = |\frac{\mu_1 - \mu_2}{\sigma_1 + \sigma_2}|, \quad (2)$$

donde μ_1 y μ_2 denotan los valores medios de expresión de la clase 1 y clase 2, respectivamente, $\sigma_1 y \sigma_2$ son las desviaciones estándar de las muestras en cada clase.

4.4. Clasificación

El resultado que se obtiene al utilizar el método de filtro en la etapa de preprocesamiento, da la posibilidad de que se obtenga información no ruidosa y no redundante, generando una primera reducción significativa de la base de datos, pero aún es imposible obtener información relevante de este resultado.

En la segunda etapa se busca una mejor clasificación de las señales de audio de personas con EP, para esto utilizamos una técnica de clasificación basada en una red neuronal artificial, la idea es obtener un porcentaje de clasificación aceptable para poder distinguir a personas que presentan la enfermedad de personas que no la presentan y así identificar que audios nos pueden ayudar a generar un diagnóstico efectivo de la EP. Los resultados que obtiene la red neuronal se validan mediante un método k-fold cross validation.

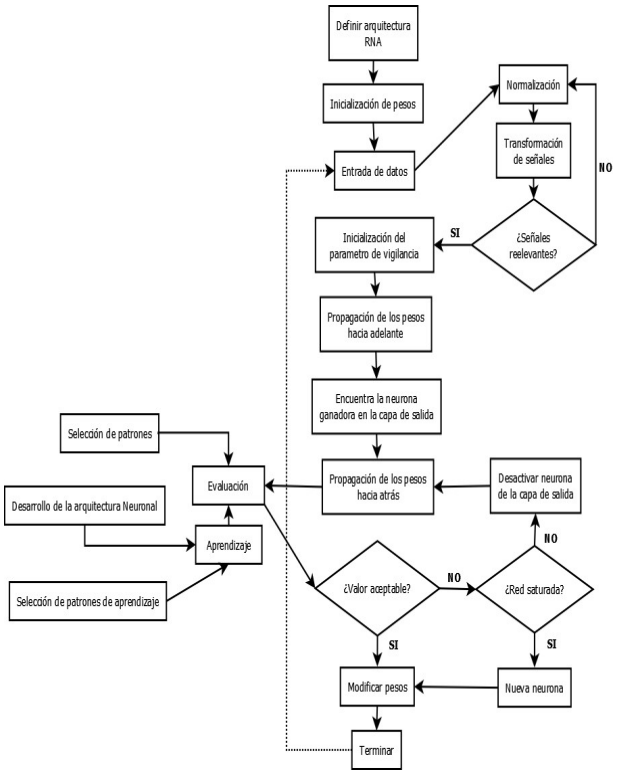


Fig. 3. Funcionamiento de la RNA.

4.5. Red neuronal artificial (RNA)

Las RNA son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales interactúan con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico [13]. Las redes neuronales son sistemas de simples elementos de proceso fuertemente interconectados [14]. Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro.

Por ejemplo, son capaces de aprender de la experiencia, de generalizar casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan información irrelevante. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se aplique en múltiples áreas [15].

En nuestro caso utilizamos un perceptrón multicapa (MLP). Para el entrenamiento de nuestra red neuronal MLP se implementó el algoritmo BP, esté ajusta los pesos entre las conexiones mediante la regla de aprendizaje delta widrow-Hoff. Está regla calcula el error cuadrado medio de la salida de la RNA y respectiva salida (salida deseada), el conjunto de las muestras de entrada se pasa por la red para minimizar el valor del error, establecido por 4 pasos principales:

- Se inicializan los pesos, a cada peso en la conexión se le asigna un pequeño valor comprendido entre 0,1 (valor aleatorio).
- Se calcula el avance; cada neurona en la capa de entrada recibe un valor, esté valor es propagado a cada neurona en la capa oculta. Para cada neurona oculta, la función de activación se calcula y propaga a cada neurona en la capa de salida, luego la neurona de salida calcula la función de activación para formar la respuesta del patrón de entrada dado.
- Se propagan los errores, cada neurona de salida calcula la diferencia entre su salida y la salida deseada para determinar el error asociado en esa neurona, luego, estos errores se distribuyen desde la capa de salida a todas las neuronas en las capas anteriores.
- Se actualizan los pesos y sesgos, y se repite el proceso.

El algoritmo se describe a continuación.

1. Según la naturaleza del problema, fácilmente se puede determinar la cantidad de neuronas en la capa de entrada y en la capa de salida, sin embargo, el número de capas ocultas y la cantidad de neuronas en estas capas no sigue ningún patrón o regla definida y se atiende más a la experiencia en la solución de problemas similares y a la complejidad matemática del problema a resolver; para este caso en particular se implementaron 14 neuronas de entrada y una de salida para configuraciones de la RNA, así como una capa oculta con dos neuronas.
2. Los datos de entrada para el MLP son los obtenidos en el preprocesamiento. La información fue normalizada por un método de nombre Min – Max. Después se hizo una primera reducción del tamaño de la base utilizando un filtro basado en SNR. Como entrada de datos, el Algoritmo Híbrido inicia con una base de dato de audio de pacientes con EP dividida en Test (evaluación) y Train (entrenamiento), cuya población es de **1040***2**6** y **128***2**6** patrones sonoros, respectivamente.
3. Selección de datos para entrenamiento y validación: En la implementación será utilizado el 70% de los datos para realizar el entrenamiento de la red, y el 30% restante se utilizará para realizar la validación del modelo obtenido y verificar si realmente el modelo entrega resultados aceptables al presentarle patrones que pueden ser desconocidos.
4. La inicialización de los pesos para el entrenamiento del Perceptrón se hace de forma aleatoriamente en el vector de pesos asociado, el cual se irá actualizando para conseguir mejores resultados.
5. Se propagan los pesos hacia atrás para que los errores sean cada vez más cercanos a 0 y por tanto el aprendizaje en las capas más alejadas de la capa de salida sea casi nulo, con la finalidad de entrenar a las redes neuronales con un número elevado de capas.
6. La red neuronal propaga los pesos hacía adelante lo cual se conoce como una función R^n en R^m con n unidades en la capa de entrada y m unidades en la capa de salida. En el desarrollo de este proyecto es utilizada como un clasificador booleano de conjuntos en R^n, donde m = 1, con la que se tiene dos opciones de clasificación:
   a. Si se tienen funciones de activación o bipolar, se considera un valor de salida

Tabla 1.Párametros usados en el método propuesto de Enfermedad de Parkinson.

Método	Parámetros	
RNA	Número de iteraciones	50, 100, 150, 200
	Número de neuronas en la capa entrada	14
	Número de neuronas en la capa oculta	1,2,3,4,5,6,7,10
	Número de neuronas en la capa de salida	1
	Umbral	0.1

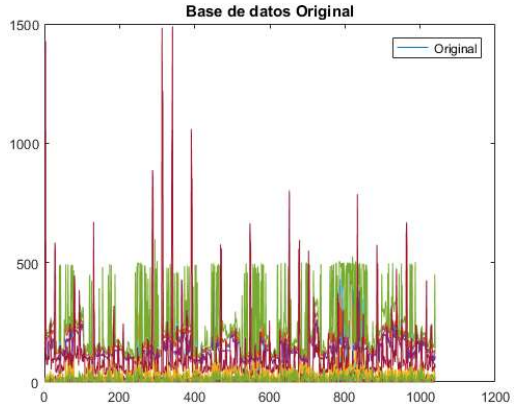


Fig. 4. Base de datos de EP Original.

(el 1, por ejemplo) como “SI” y el otro como “NO”.

b. Si se usa el sigmoide, se considera un valor de salida por encima de 0.5 como “SI” y un valor por debajo como “NO”.

c. Para clasificaciones con m posibles valores, cada unidad de salida corresponde con un valor de clasificación; la unidad con mayor salida es la que indica el valor de clasificación.

7. Aprendizaje: Los pesos sinápticos de la red son ajustados con el objetivo de capturar la información que se presenta, y de esta forma obtener respuestas adecuadas. Este proceso básicamente consiste en la presentación de un conjunto de datos, conocido como conjunto de patrones de entrenamiento, determinado número de veces, conocido como ciclos, hasta que se produzca uno de los siguientes eventos:
   - El error entre la salida de la red y la deseada alcanzo un valor aceptable.
   - Se alcanzó el número máximo de ciclos.

8. Para una señal de audio X se aplica la validación de la RNA, el cual sirve para calcular la exactitud (el resultado arrojado cuando es mayor a 0.5 indica que es un paciente con EP de lo contrario, el paciente no presenta la EP) promedio de un subconjunto de audios. Los datos de la muestra se dividen en K subconjuntos de prueba y el resto (K-1) como datos de entrenamiento.

9. Se efectúa la clasificación de señales de audio de pacientes con EP en base a su

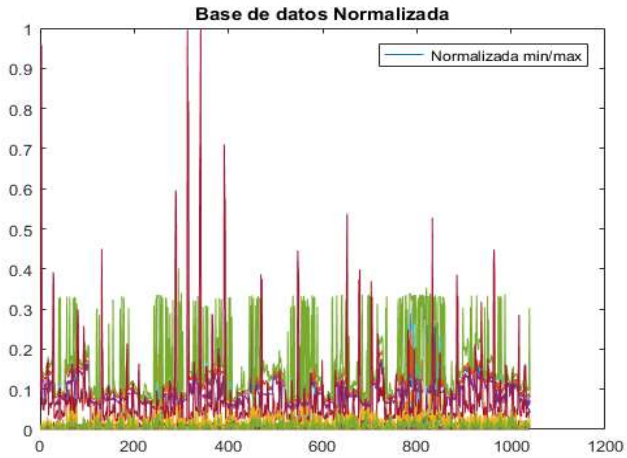


Fig. 5. Base de datos de EP Normalizada min/máx.

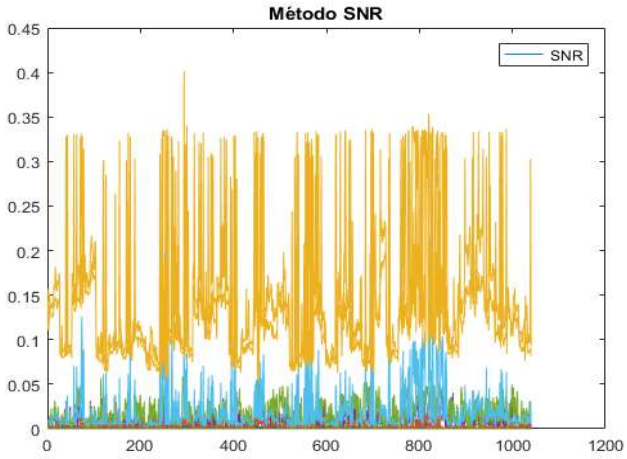


Fig. 6. Método SNR.

experiencia con la base de datos Train, clasifica y evalúa la información de la base de datos Test de la cual obtiene características determinadas a la salida final.

10. El MLP genera una salida en formato binario 0 / 1, la cual representa pacientes con EP o sin ella. En la Figura 3 se muestra el funcionamiento de la RNA.

5. Experimentos y resultados

En esta sección, se presentan los parámetros bajo los que ha trabajado el algoritmo híbrido, también se muestra los resultados obtenidos y un estudio de comparación con otros métodos reportados en la literatura.

5.1. Parámetros

El método AHEP ha sido implementado en Matlab (Versión 9.0.0.341360), el cual trabaja con una base de datos de dominio público que está conformada por un total de 1040 señales de audio tomadas de la frecuencia cuando un paciente emite una vocal especifica con un poco de tartamudeo. La Tabla 1 muestra los parámetros utilizados en este experimento.

5.2. Base de datos

El sistema trabaja con una base de datos de audio donde los pacientes dicen una vocal y se toma la frecuencia con la que se emite una señal de tartamudeo. Está formada por 20 Pacientes con EP (6 mujeres, 14 hombres) y 20 individuos sanos (10 mujeres, 10 hombres), obteniendo un total de 168 señales de audio [16]. Las señales se pueden ver en la Figura 4.

La imagen muestra que cada señal genera 1024 patrones para ser analizadas, por lo cual, es necesario reducir la información de la base de datos para poder estudiar las señales. Para solucionar este problema se propone un algoritmo de dos fases, la fase de preprocesamiento y la fase de clasificación, los resultados que se obtuvieron se describen a continuación.

5.3. Resultados de preprocesamiento

En esta etapa se realizó una estandarización y una preselección de las señales de audio que tienen una mejor información sobre la EP. Los resultados obtenidos en esta etapa se describen a continuación.

5.4. Normalización

A menudo, los atributos no se encuentran en un rango fijo, esto puede generar un sobre entrenamiento del algoritmo de clasificación, al estandarizar los datos colocamos los atributos en un rango entre 0 - 1 con la finalidad de obtener un resultado aceptable por el algoritmo de clasificación, la Figura 5 muestra la base de datos de Parkinson normalizada.

Como se muestra en la figura 5, los patrones de audio toman una nueva escala numérica sin perder características representativas. La intención de este paso es homogeneizar los patrones a datos más pequeños de su escala normal, ya que trabajar con los datos en bruto (escalas numéricas reales), pueden generar un sobreajuste al momento de ser clasificados.

5.5. Filtro SNR

Después de la normalización, se hace una pequeña selección de las mejores señales existentes dentro de la base de datos de Parkinson. La idea en esta fase, es seleccionar una o varias señales efectivas para que el clasificador (RNA) trabaje con un subconjunto de audios y pueda identificar que señal ayuda en el diagnóstico de la EP.

Tabla 2. Tasa de clasificación obtenida por la RNA.

Iteraciones	RNA
50	0.9762
100	**0.9996**
150	0.9975
200	0.9944

Tabla 3. Comparación de resultados obtenidos por los tres clasificadores.

Iteraciones	SVM	KNN	RNA
50	0.9300	0.9227	0.9762
100	0.9649	0.9628	**0.9996**
150	0.9769	**0.9749**	0.9975
200	**0.9825**	0.9808	0.9944

El filtro se emplea para precisar el nivel sonoro ponderado de la señal obtenida de pacientes con EP. Este filtro calcula el nivel efectivo de precisión sonora, eliminando la incertidumbre contenida en la BD de audio de Parkinson. Los resultados obtenidos por el filtro SRN se muestran en la Figura 6.

El filtro reduce la dimensión de la base de datos, seleccionando solo señales efectivas que ayudan a entrenar mejor al algoritmo de clasificación. Con el subconjunto obtenido por el filtro, se entrena una red neuronal como proceso de clasificación de las señales de audio. Los resultados obtenidos se describen a continuación.

5.6. Resultados de la clasificación

En el protocolo experimental, se realizó una selección de un subconjunto de señales de audio para hacer una primera reducción de la dimensión de la base de datos, utilizando el filtro estadístico Relación señal a ruido (SNR). Con el resultado obtenido por este proceso, se entrena una Red Neuronal Artificial para obtener una mejor tasa de clasificación y distinguir pacientes que presentan la enfermedad de Parkinson y los que no lo tiene.

El algoritmo es ejecutado con diferentes iteraciones que van desde 50,100, 150 y 200 veces para obtener una mejor clasificación. Los resultados obtenidos en cada iteración demuestran un rendimiento eficaz de la red neuronal, clasificando con exactitud las señales de audio.

Las tasas de clasificación obtenidas por la red neuronal se muestran en la Tabla 2. Para verificar que la RNA ha obtenido la mejor tasa de clasificación, se clasifican las señales de audio con dos métodos clásicos de aprendizaje máquina.

La idea es medir el rendimiento que obtienen cada uno de los métodos y comparar su mejor resultado. Los métodos con los que se ha comparado son el clasificador SVM y el clasificador KNN con 3 k-vecinos. Los resultados obtenidos se muestran en la Tabla

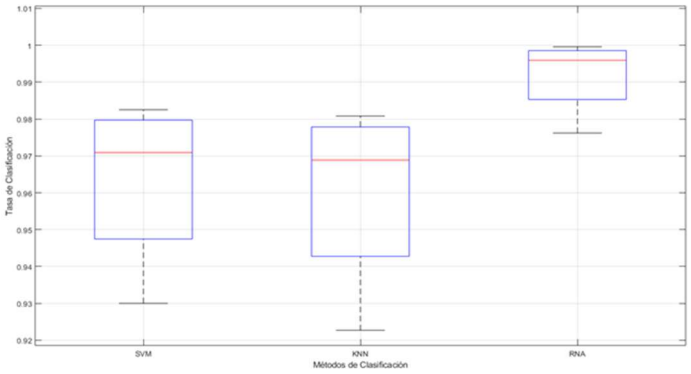


Fig. 7. Comparación de los tres clasificadores.

Tabla 4. Comparación con métodos reportados en la literatura.

Autores	EP %	Iteraciones	Métodos
Pun et al. [3]	90 %	--	Minería de datos
Tsiouriset al. [4]	87 %	--	Minería de datos
Yadavet al. [5]	**97 %**	50	SVM
Brezocnik et a.l [17]	86.47 %	250	RNA + PCA de Kendall
Christensen et al. [18]	91 %	50	RNA
Acton1 et al. [19]	94.4 %	--	RNA
Venhovens et al. [20]	90 %	--	Aplicación de electrodos de superficie
Najiya.M. [21]	**98.5 %**	700	RNA
AHEP	**99.96 %**	100	RNA+SRN

3 En la Tabla 3 se observa que la RNA alcanza una tasa de clasificación del 99.96 % con 100 iteraciones.

Para el SVM son 200 iteraciones obteniendo un rendimiento de 98.25% y para el KNN consiguen un rendimiento de 97.49% con 150 iteraciones. Otra evaluación que se ha realizado al método propuesto es la comparación con diferentes métodos reportados en la literatura.

La Tabla 4 muestra el estudio de comparación de la siguiente manera, en la primera columna, se muestran los autores con los que se han comparado los resultados obtenidos. En la segunda columna se muestra el porcentaje obtenido por cada autor. La tercera columna muestra el número de iteraciones y la cuarta muestra el método utilizado. Cada autor con el que se han comparado los resultados obtenidos, trabaja con un modelo similar al que se propone. Utilizando diferentes modelos de aprendizaje máquina o métodos de clasificación.

El estudio de comparación muestra que el método propuesto obtiene una tasa de clasificación alta. Al comparar los resultados obtenidos con los diferentes métodos como el SVM, PCA, RNA propuestos por los autores. Se nota que se ha logrado superar sus tasas de clasificación y en ocasiones utilizar un número menor de iteraciones. Demostrando que el algoritmo propuesto mejora la clasificación de la base de datos de EP.

6. Conclusiones

En este trabajo, se presentó un método para la selección y clasificación de un conjunto de señales de audio que son utilizados para la identificación de la EP. Se utilizó un método de filtrado de datos para hacer una primera reducción de la dimensión de la base de datos.

Para realizar la clasificación dentro del subconjunto obtenido por el método de filtro, se ha usado una Red Neuronal Artificial. El método propuesto determina una tasa de clasificación alta obtenida con un subconjunto de señales de audio pequeño. Se realizaron dos estudios de comparación, el primero consiste en utilizar otros clasificadores (SVM y KNN) para verificar si la RNA obtiene un mejor porcentaje de clasificación. El segundo estudio se realizó al comparar los resultados con otros métodos reportados en la literatura.

En este estudio, se puede notar que el método propuesto supera a los demás métodos implementados, logrando obtener una tasa de clasificación alta a diferencia de cada método reportado. La meta es aumentar al máximo la exactitud de la clasificación y, por otro lado, minimizar el número de señales de audio a utilizar.

Referencias

1. Herman-Bartstra, A. L.: Manganisme of Parkinson? Beroepsziekten. pp. 1–4 (2017)
2. Dietrichs, E., Odin. P.: Algorithms for the treatment of motor problems in Parkinson's. Acta Neurol Scand, vol. 136, no. 5, pp. 378–385 (2017)
3. Pun, U. K., Gu, H., Dong, Z., Artan, N. S.: Classification and visualization tool for gait analysis of Parkinson's disease. In: Proceedings of 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2407–2410 (2016) doi: 10.1109/EMBC.2016.7591215
4. Tsiouris, K. M., Rigas, G., Antonini, A., Gatsios, D., Konitsiotis, S., Koutsouris, D. D., Fotiadis, D. I.: Mining motor symptoms UPDRS data of parkinson's Disease Patients for the Development of Hoehn and Yahr Stimulation Decision Support System. In: Proceedings of International Conference on Biomedical & Health Informatics (BHI), IEEE EMBS, pp. 445–448 (2017) doi: 10.1109/BHI.2017.7897301
5. Yadav, G., Kumar Y., Sahoo, G.: Predication of Parkinson's disease using data mining methods: A comparative analysis of tree, statistical and support vector machine classifiers. In: Proceedings of National Conference on Computing and Communication Systems, pp. 1–8 (2012) doi: 10.1109/NCCCS.2012.6413034
6. Romano, J. I.: Introducción a la digitalización de bioseñales. IV Congreso Microelectrónica Aplicada (2013)
7. Duval, B., Hao, J. K., Hernández-Hernández, C.: A memetic algorithm for gene selection and molecular classification of cancer. In: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, GECCO '09, pp. 201–208 (2009) doi: 10.1145/1569901.1569930
8. Witten, I. H., Eibe, F.: Data mining practical machine learning tools and techniques. San Francisco CA: Elsevier (2005)
9. Montiel, L. A. H.: Hybrid algorithm applied on gene selection and classification from different diseases. IEEE Latin America Transactions, vol. 14, no. 2, pp. 930–935 (2016) doi: 10.1109/TLA.2016.7437242

10. Mishra, D., Sahu, B.: Feature selection for cancer classification: A signal-to-noise ratio approach. International Journal of Scientific & Engineering Research, vol. 2, no. 4, pp. 1–6 (2011)
11. Krogh, A.: What are artificial neural networks. Nat biotechnology, pp. 195–197 (2008) doi: 10.1038/nbt1386
12. Enzo, G. and Massimo, B.: Introduction to artificial neural networks. European Journal of Gastroenterology & Hepatology, vol. 19, no. 12, pp. 1046–1054 (2007) doi: 10.1097/MEG.0b013e3282f198a0
13. Nedjah, N., Ajith, A., Mourel, L. M.: Hybrid artificial neural network. Neural Comput & Applic, pp. 207–208 (2007) doi: 10.1007/s00521-007-0083-0
14. Sakar, B. E., Isenkul, M. E., Sakar, C. O., Sertbas, A., Gurgen, F., Delil, S., Apaydin, H., Kursun, O.: Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings. IEEE Journal of Biomedical and Health Informatics, vol. 17, no. 4, pp. 828–834 (2013) doi: 10.1109/JBHI.2013.2245674
15. Brezocnik, M., Berus, L., Klancnik, S.: Classifying Parkinson's disease based on acoustic measures using artificial neural networks. Sensors, vol. 19, no. 1. pp. 1–15 (2018). doi: 0.3390/s19010016
16. Christensen, E., Abosch, A., Thompson, J. A., Zylberberg. J.: Inferring sleep stage from local field potentials recorded in the subthalamic nucleus of Parkinson's patients. European Sleep Research Society, vol. 28, no. 4 (2018) doi: 10.1111/jsr.12806
17. Acton, P., Newberg, A.: Artificial neural network classifier for the diagnosis of Parkinson's disease using TRODAT-1 and SPECT. Physics in Medicine & Biology, vol. 51, no. 12 (2006) doi: 10.1088/0031-9155/51/12/004
18. Venhovens, J., Meulstee, J., Bloem, B. R., Verhagen, W. I. M.: Neurovestibular analysis and falls in Parkinson's disease and atypical parkinsonism. Federation of European Neuroscience Societies, vol. 43, no. 12, pp. 1636–1646 (2016) doi: 10.1111/ejn.13253
19. Najiya, M. O., El-Hawary, M. E.: Optimizing classifier performance for Parkinson's disease detection. In: Proceedinbgs of IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–6 (2017) doi: 10.1109/CCECE.2017.7946697

ISSN 1870-4069

Determinación automática del color del semáforo Mexicano del COVID-19 a partir de las noticias

Miguel Á. Álvarez-Carmona, Ramón Aranda

Centro de Investigación Científica y de Educación Superior de Ensenada,
Unidad de Transferencia Tecnológica Tepic,
México

{malvarez, aranda}@cicese.edu.mx

Resumen. Este trabajo presenta el análisis de modelos de clasificación textual para determinar automáticamente el semáforo epidemiológico regional mexicano a través de noticias de COVID. Se recolectó una base de datos con 4270 noticias referente a COVID, desde el 1 de junio de 2020 hasta el 28 de marzo de 2021. La etiqueta de cada noticia es el color del semáforo epidemiológico que el gobierno mexicano catalogó en la semana de la publicación de la noticia. Se aplicaron clasificadores como: SVM, KNN, Random Forest y Deep Learning. Los resultados muestran que es posible aprovechar la información que se publica en las noticias para determinar el color del semáforo hasta con 4 semanas de anticipación obteniendo resultados de hasta 0.74 de F-measure, el cual es un resultado competitivo tomando en cuenta el desbalance de clases de esta tarea.

Palabras clave: COVID-19, procesamiento de lenguaje natural, clasificación textual, semáforo epidemiológico.

Automatic Determination of the Color of the Mexican Traffic Light of COVID-19 from the News

Abstract. This paper presents the analysis of textual classification models to automatically determine the Mexican regional epidemiological traffic light through COVID news. A database with 4,270 news items regarding COVID was collected from June 1, 2020 to March 28, 2021. The label of each item is the color of the epidemiological traffic light that the Mexican government cataloged in the week of publication of the news. Classifiers such as: SVM, KNN, Random Forest and Deep Learning were applied. The results show that it is possible to take advantage of the information published in the news to determine the color of the traffic light up to 4 weeks in advance, obtaining results of up to 0.74 of F-measure, which is a competitive result taking into account the imbalance of classes. of this task.

Keywords: COVID-19, natural language processing, textual classification, epidemiological traffic light.

1. Introducción

En 2020, México, al igual que todos los países del mundo, se enfrentó a la pandemia generada por el COVID-19, declarada como emergencia de salud pública de importancia internacional [4]. Esta pandemia obligó a varios sectores económicos a pausar su actividad (principalmente el sector turístico), lo que provocó una pérdida económica importante en distintos niveles como en alojamiento, restaurantes, transporte, comercio, entre otros [5].

Para mitigar en mayor medida las pérdidas económicas derivadas de la pandemia, mientras se minimiza el peligro de contagio, el gobierno mexicano implementó un semáforo epidemiológico, el cual, dependiendo de su color permite ciertas actividades. De esta manera, el color del semáforo se vuelve muy importante para conocer las medidas que se deben tomar y estimar el movimiento que podría haber entre clientes y prestadores de servicios.

En la Figura 1 se muestran los 4 niveles del semáforo epidemiológico. Es un sistema de cuatro colores ordenados: rojo, naranja, amarillo y verde, donde rojo es el color más restrictivo y verde el que concede mayor movimiento e interacción. El color del semáforo se actualiza de manera semanal y es independiente en cada estado del país.

El color se calcula a partir de diversos factores, como la inercia de la curva epidemiológica, camas disponibles en los hospitales, ritmo de contagio entre otros [11]. Aunque el gobierno publica el color semanal unos días antes de su entrada en vigor, si se pudiera conocer con mayor antelación esta información, podrían tomarse mejores medidas y estar mejor preparados para los cambios que involucra una variación en el semáforo.

Los datos relevantes que se consideran para calcular el color semanal del semáforo, son dados a conocer indirectamente a través de las noticias estatales, normalmente a través de sitios web de noticieros. De esta manera surge la posibilidad de tomar todos estos datos y con ayuda de Inteligencia Artificial (IA), tratar de predecir el color del semáforo epidemiológico.

Esto se puede aterrizar como una tarea de clasificación, donde las instancias son las noticias que tienen que ver con el COVID–19 en algún estado de la república y la clase sería el color del semáforo. De esta manera, para el tratamiento del texto de las noticias se aplicarían métodos de Procesamiento de Lenguaje Natural (PLN).

La problemática de tratar de solucionar la predicción del semáforo, recae en la poca cantidad de información, ya que desde que hay registro del semáforo epidemiológico a la fecha, han pasado 43 semanas. Por lo que las técnicas de clasificación textual que han tenido mucho éxito en los últimos años, como los Transformers [16], no funcionarían de manera óptima en este tema.

También, existe un claro desbalance de datos, ya que colores como el verde han ocurrido pocas veces, desde que inició la pandemia. En este trabajo de investigación, se propone implementar modelos de clasificación textual a través de las noticias de COVID, planteando dar respuestas a 3 preguntas de investigación:

1. ¿Existen datos relevantes en las noticias de COVID regionales de tal manera que pueden ser aprovechados para un modelo de clasificación textual y determinar el color del semáforo epidemiológico?

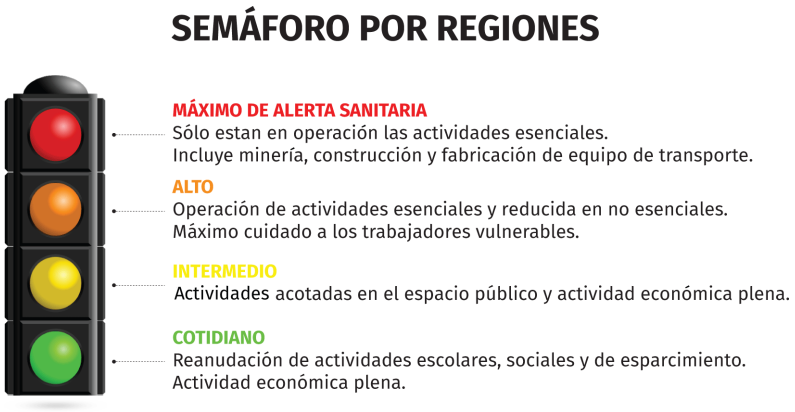


Fig. 1. Semáforo epidemiológico regional en México[1].

2. ¿Cuáles son las mejores representaciones textuales y qué algoritmos de clasificación funcionan mejor para la tarea de clasificación del semáforo epidemiológico a través de las noticias tomando en cuenta la cantidad de datos y el desbalance de clases?
3. ¿Con cuántas semanas de anticipación se puede predecir el color del semáforo epidemiológico de tal manera que se obtenga un resultado razonable?

De todos los estados de la república mexicana, el que más cambios en el color del semáforo ha tenido es Veracruz. Este estado ha cambiado de color de una semana a otra 10 veces. Además de que es de los pocos que ha pasado por los 4 colores del semáforo e incluso ha tenido retrocesos de color en sus cambios. De esta manera, se utilizarán los datos de este estado para llevar a cabo esta investigación. Se recolectó una colección de 4270 noticias de Veracruz relacionadas con el COVID–19 (aproximadamente 99.3 noticias por semana).

El resto del documento está organizado de la siguiente manera: en la sección 2 se describen algunos trabajos de investigación sobre noticias que se han llevado a cabo para analizar el COVID-19 desde el punto de vista informativo. En la sección 3 se detalla la metodología que se llevó a cabo para determinar el color del semáforo epidemiológico de Veracruz a través de las noticias. En la sección 4 se muestran los experimentos y resultados obtenidos. Finalmente, en la sección 5 se presentan las conclusiones y el trabajo a futuro derivado de esta investigación.

2. Noticias y COVID–19

Hoy en día, el apoyo de nuevas tecnologías y aplicaciones de la IA, Internet de las Cosas (IoT), —Big Data y Machine Learning contra el COVID ha sido de gran importancia debido al poder de detección, seguimiento, predicción y toma de decisiones ante los diferentes panoramas asociados a la pandemia [15, 8, 6].

[1] https://augecorp.com.mx/asi-funciona-el-semaforo-de-reinicio-de-actividades/

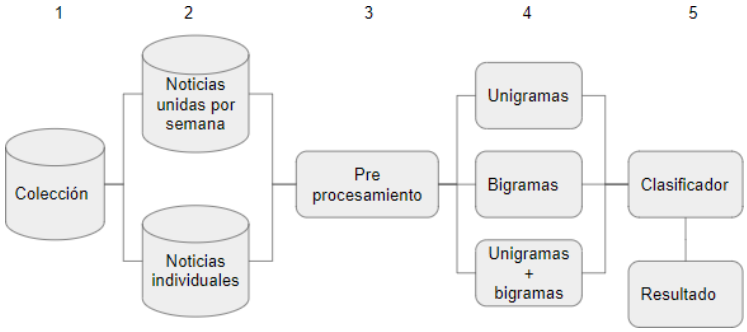


Fig. 2. Metodología propuesta.

Particularmente, las noticias en línea han tomado un papel importante para mantenerse informado sobre la pandemia. Y en relación a esto, han surgido recientes estudios sobre la relación entre la COVID–19 y las noticias. En [1] modelan los predictores sobre compartir noticias falsas entre los usuarios de las redes sociales, en [10], los autores crearon un sistema automatizado para verificar noticias e información sobre el COVID-19.

En [12] predicen la rentabilidad de las acciones combinando noticias financieras con noticias relacionadas a salud. Los autores de [2] analizan segmentos de videos con noticias de COVID para determinar la información transmitida sobre el COVID. En [7] desarrollan nuevas técnicas para medir la detección de mentiras usando noticas falsas sobre COVID–19.

Con los ejemplos mencionados, se puede ver como las noticas están jugando un papel importante en la era del COVID–19. En este trabajo, a diferencia de los trabajos mencionados, proponemos predecir el color del semáforo epidemiológico de los estados de la república mexicana, particularmente del estado de Veracruz, mediante la implementación de modelos de clasificación textual aplicado a las noticias relacionadas con el COVID.

3. Metodología

La principal idea detrás de este trabajo es tratar de capturar las características importantes de las noticias que hablan de COVID-19 para determinar el color del semáforo epidemiológico. Para lograr esto se propone una metodología dividida en 5 etapas:

1. Recolección de datos.
2. Agrupaciones de datos.
3. Pre–procesamiento de datos.
4. Representación de datos.
5. Clasificación de los datos.

En la Figura 2 se muestra una representación gráfica de la metodología propuesta. A continuación se describirán cada una de las etapas.

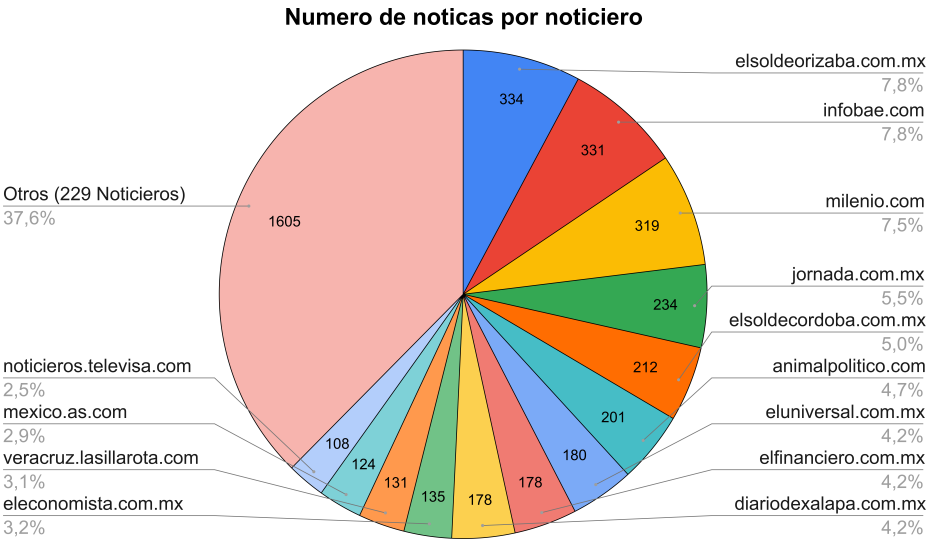


Fig. 3. Cantidad de noticias de la colección divididas por noticiero.

3.1. Recolección de datos

Para recolectar las noticias en este trabajo, se desarrolló una herramienta en lenguaje de programación Python y el uso de la librería BeautifulSoup. La herramienta generada fue configurada para utilizar el motor de búsqueda de Google Noticas de México, con noticias solo en español, usando las palabras claves "COVID y Veracruz". La búsqueda de las noticas se realizó por periodos semanales a partir del 1 de junio de 2020 hasta el 28 de marzo de 2021, dando un total de 43 semanas.

Se limitó a recolectar un máximo de 100 noticias por semana. Es importante mencionar que una ventaja de usar el motor de búsquedas de Google es que ordena las noticias por relevancia, lo cual nos da certeza de que la información recolectada no contenga noticias falsas. Un total de 4270 noticias fueron recolectadas en todo el periodo de tiempo de 242 noticieros online.

La Figura 3 muestra el número de noticias obtenidas por noticiero. Se puede observar que en 13 noticieros se concentra más del 60 % de los datos (2665 noticias), donde elsoldeorizaba.com.mx aporta más información con un total de 334 noticias (7.8 % del total de la información recolectada). El 37.6 % de las noticias se repartió en 229 noticieros con un promedio de 7.01 noticias y una desviación estándar de 14.03.

De estos noticieros, 8 aportaron entre 100 y 50 noticias, 27 noticieros aportaron entre 50 y 10 noticias, y 193 con menos de 10 noticias (donde 114 noticieros reportaron solo 1 nota). Todos los datos recolectados se pueden descargar a través del siguiente repositorio[2]. De las 43 semanas recolectadas, en 9 semanas hubo semáforo en rojo, 13 semanas el semáforo estuvo en naranja, 19 en amarillo y solo 2 en verde. Estas cifras complican el problema de clasificación por el evidente desbalance de datos.

[2] https://github.com/moncho-arac/News-COVID-Veracruz.git

(a) (b)

Fig. 4. Ejemplos de noticias durante (a) semáforo en rojo, y (b) semáforo verde[3].

3.2. Agrupaciones de datos

Como se mencionó en la sección 3.1 las noticias están agrupadas en semanas. Se proponen dos maneras de agrupar los datos. Primero, utilizar cada noticia individual como una instancia. Así se contarían con 4270 instancias etiquetadas. La segunda manera propuesta es juntar todas las noticias de una semana en un solo texto concatenado. De esta forma, se tienen 43 instancias etiquetadas.

3.3. Pre–procesamiento de datos

Con el fin de extraer las características más importantes de los textos, se aplicó una fase de pre–procesamiento. Las transformaciones que se llevaron a cabo son:

1. Se convirtieron las mayúsculas en minúsculas.
2. Se retiraron las palabras vacías.
3. Se removieron los signos de puntuación.
4. Se reemplazaron los dígitos por la letra 'd'.
5. Para evitar que las fechas influyan, también se eliminaron las palabras de los meses del año.
6. Se aplicó stemming a cada una de las palabras en los textos.
7. Se removieron los tokens que aparecen menos de 15 veces[4] en toda la colección.

3.4. Representación de datos

Ya que lo importante del texto se centra en el contenido de la noticia y no en el estilo, se plantea trabajar a nivel de palabras. Una de las técnicas que funcionan bien para representar el contenido con pocos datos es el de N–gramas de palabras [3]. Para representar los datos se propone utilizar N = $\{1,2,\{1,2\}\}$. Es decir, se extraerán unigramas, bigramas y finalmente se fusionarán los dos espacios de unigramas y bigramas.

[3] eluniversal.com.mx/tag/veracruz

[4] Número elegido de manera empírica

Tabla 1. Características del algoritmo de Deep Lerning aplicado.

Capas ocultas	5
Número de neuronas por capa	1000
Función de activación de las capas ocultas	Relu
Neuronas de la capa final	4
Función de activación de la capa final	Softmax
Función de pérdida	Categorical Cross Entropy
Optimizador	Adam
Épocas	50

A partir del pre–procesamiento de datos descrito en la sección 3.3 y de la representación de N-gramas se propone aplicar al texto el modelo de pesado de TF-IDF (Term frequency – Inverse document frequency) [9].

Esta representación se basa en no solo premiar las palabras más frecuentes de un documento (en este caso, una noticia), sino que también castiga a las palabras que aparecen en muchos documentos. Así palabras que no son importantes o que aparecen a lo largo de toda la colección tendrán un peso bajo, mientras que las palabras importantes en una noticia y que no aparecen mucho en otras noticias tendrán un valor de pesado alto.

En la Figura 4 se presentan dos ejemplos de noticias, la primera (4a) es una noticia cuando el semáforo estaba en rojo y la segunda (4b) cuando el semáforo estaba en verde. Como es posible observar, el vocabulario que se utiliza en ambas noticias es muy distinto, lo cual permite pensar que un pesado como el de TF-IDF es ideal para esta tarea.

3.5. Clasificación de los datos

Para clasificar las noticias, se optó por aplicar una división de datos de 10–fold cross–validation [13]. Posteriormente se aplicaron los algoritmos más populares para tareas de clasificación supervisada [14]. Estos algoritmos son:

1. Support Vector Machine (SVM).
2. K–Nearest–Neighbor (KNN), con K = $\{1,3,5,7\}$.
3. Decision Tree (DT).
4. Random forests (RF).
5. Naive Bayes (NB).
6. Deep learning (DL).

Para el algoritmo de Deep Learning (DL) se hizo una implementación en python 3 con la paquetería de keras con TensorFlow versión 2. En la Tabla 1 se describen las características de la arquitectura del algoritmo de Deep Learning. Para los demás algoritmos de clasificación se utilizó la paquetería de sklearn de python. Para todos y cada uno de los algoritmos aplicados, se debe evaluar el rendimiento. Las métricas utilizadas para los resultados son Accuracy y F-measure [14].

Tabla 2. Mejores resultados para las noticias separadas.

m	Representación	Algoritmo	Accuracy	F-measure
0	Unigramas	DL	54.56	0.53
0	Bigramas	DL	57.23	**0.57**
0	Unigramas + Bigramas	DL	57.75	0.55
1	Unigramas	DL	53.46	0.54
1	Bigramas	DL	58.50	**0.58**
1	Unigramas + Bigramas	DL	57.35	0.57
2	Unigramas	DL	55.92	0.54
2	Bigramas	DL	58.26	**0.58**
2	Unigramas + Bigramas	DL	56.15	0.56
3	Unigramas	DL	55.48	0.54
3	Bigramas	DL	59.13	**0.57**
3	Unigramas + Bigramas	DL	59.13	0.57
4	Unigramas	DL	56.88	0.57
4	Bigramas	DL	59.92	**0.59**
4	Unigramas + Bigramas	DL	59.53	0.59

3.6. Clasificación de semanas futuras

La razón principal de construir los modelos mencionados de clasificación es predecir el color del semáforo epidemiológico. Sin embargo, no es útil determinar dada una noticia de alguna semana, el semáforo de esa misma semana, esto debido a que ese semáforo ya se sabe. Lo interesante sería determinar los semáforos futuros.

Para este trabajo proponemos inferir, dada una noticia de alguna semana S, el semáforo epidemiológico de la semana $S + m$ donde $m \in \{0, 1, 2, 3, 4\}$. Cuando $m = 0$, se estará infiriendo la semana de la noticia que se está analizando y cuando $m = 4$ se estará infiriendo el semáforo del siguiente mes (4 semanas) desde que la noticia se publicó.

4. Experimentos y resultados

En esta sección se describen los resultados obtenidos a partir de los experimentos seguidos por la metodología descrita en la sección 3. En la Tabla 2 se muestran los mejores resultados obtenidos cuando se analizan las noticias separadas. En este caso, la colección tiene un total de 4270 instancias. La dimensión de características para unigramas fue de 6595, para bigramas fue de 12975 y la combinación de unigramas y bigramas obtiene una dimensionalidad de 19560 características.

En esta tabla aparecen los mejores resultados obtenidos por cada combinación, tanto de semanas a futuro como de representación de datos. Es notable ver que en todos los resultados el mejor algoritmo de clasificación fue el de Deep Learning (DL).

También es importante mencionar que los bigramas obtuvieron los mejores resultados respecto a los unigramas y su combinación. También, en estos resultados es posible ver que los resultados van desde 0.57 de F-measure hasta 0.59 lo que da una idea de que los resultados varían muy poco entre ellos.

Tabla 3. Resultados para las noticias unidas por semana.

m	**Representación**	**Algoritmo**	**Accuracy**	**F-measure**
0	Unigramas	DT	69.76	**0.54**
0	Bigramas	KNN-1	69.76	0.53
0	Unigramas + Bigramas	KNN-3	67.44	0.51
1	Unigramas	KNN-1	69.76	0.68
1	Bigramas	KNN-1	67.44	**0.74**
1	Unigramas + Bigramas	KNN-1	67.44	0.74
2	Unigramas	DT	62.79	0.47
2	Bigramas	DL	63.50	**0.63**
2	Unigramas + Bigramas	DL	62.50	0.58
3	Unigramas	DL	57.00	0.50
3	Bigramas	KNN-1	60.46	**0.63**
3	Unigramas + Bigramas	KNN-1	51.16	0.56
4	Unigramas	KNN-1	62.79	0.47
4	Bigramas	KNN-1	65.11	0.50
4	Unigramas + Bigramas	KNN-7	69.76	**0.52**

Tabla 4. Resumen de resultados para las noticias unidas.

m	**Representación**	**Algoritmo**	**Accuracy**	**F-measure**
0	Unigramas	DT	69.76	0.54
1	Bigramas	KNN-1	67.44	0.74
2	Bigramas	DL	63.50	0.63
3	Bigramas	KNN-1	60.46	0.63
4	Unigramas + Bigramas	KNN-7	69.76	0.52

El peor resultado se obtiene cuando $m = 0$ y el mejor cuando $m = 4$. Por otro lado, en la Tabla 3 se muestran los resultados obtenidos por el análisis cuando las noticias se unen de manera semanal en un solo documento. En esta tabla es posible ver que los resultados son más altos que los resultados obtenidos por las noticias separadas.

Es importante mencionar que para este caso se analizan 43 documentos en lugar de 4270. Sin embargo parece que al unir esta información los algoritmos son capaces de capturar mejor información a pesar de la poca cantidad de instancias. También es importante notar que los algoritmos con mejores resultados son más variados, ya que aparecen otros algoritmos diferentes a Deep Learning.

Esto se puede explicar por la misma disminución de instancias. En este caso aparecen algoritmos como DT y KNN, sin embargo, también vuelve a aparecer DL en 3 ocasiones. En la Tabla 4 se muestra el resumen de resultados obtenidos al analizar las noticias unidas semanalmente. En esta tabla se puede ver el peor resultado se obtiene cuando $m = 4$ con 0.52 de F-measure mientras que el mejor resultado se obtiene cuando $m = 1$ con 0.74.

También es importante ver que aunque los bigramas dan buena información, los unigramas son mejores cuando m=0 y la unión de unigramas y bigramas es mejor cuando $m = 4$.

Tabla 5. Top 10 bigramas con ganancia de información.

Top	0	1	2	3	4
1	cambi roj	acces vacun	asciend dd	aglomer paseant	acat med
2	cas defuncion	actualiz semafor	baj la	aislamient domicili	fiest play
3	centr hospitalari	antros bar	batall covid	amarill naranj	reabr puert
4	confirm fallec	asciend dd	contagi ddd	ampli capac	reactiv activ
5	confirm millon	aument contagi	cuatr seman	buen result	realiz fiest
6	cuant fallec	confirm fallec	ddd sospech	cam diput	sospech ddd
7	ddd victim	protocol sanitari	diagnost posit	concentr person	sospech port
8	mil contagi	contagi ddd	impact econom	reactiv activ	muert contagi
9	muert acumul	transmision virus	industri turist	realiz fiest	confirm millon
10	quedat cas	ola cov	mal manej	variant sudafrican	muert acumul

4.1. Análisis de los resultados

En esta sección se pretende responder a las preguntas de investigación planteadas para este trabajo.

Pregunta 1: ¿Existen datos relevantes en las noticias de COVID regionales de tal manera que pueden ser aprovechados para un modelo de clasificación textual y determinar el color del semáforo epidemiológico?

A partir de los resultados, es posible notar que los algoritmos están aprovechando información importante de las noticias para poder tomar decisiones. Para poder observar esta información se obtienen las características con mayor ganancia de información. Estas características representan los tokens que contribuyen más para clasificar entre los cuatro colores del semáforo.

En la Tabla 5 se muestran las 10 características más importantes para clasificar el color del semáforo con $m = \{0, 1, 2, 3, 4\}$. Lo primero que se observa es que cuando $m = 0$, 1 y 2, lo que más información aporta son estadísticas de contagios y fallecidos como se ve en bigramas como "confirm fallec", "ddd victim", "contagi ddd" o "baj la".

Esto indica que efectivamente, esta información es compartida por las noticias y se está aprovechando. También se debe notar que en la semana 2 empiezan a aparecer temas como impacto económico o industria turística, lo cual puede dar evidencia de que el factor económico también ayuda a que el gobierno cambie el color del semáforo.

Por otro lado, mientras se aleja en el futuro, las características más importantes se centran en noticias que hablen de aglomeraciones y fiestas, por ejemplo en bigramas como "aglomer paseant", "realiz fiest" o "concentr person". Esto podría indicar que es posible calcular el efecto que un evento masivo puede tener en el semáforo epidemiológico futuro a través de este tipo de técnicas.

También es interesante ver que "cam diput" es un bigrama importante para calcular el color del semáforo. Finalmente, algo a notar es la importancia del bigrama "variant sudafrican" el cual se refiere a la variante sudafricana que apareció a mediados de diciembre de 2020 y que empeoró la situación de la pandemia.

Tabla 6. Promedio del ranking de los algoritmos de clasificación.

Noticias Unidas		**Noticias Separadas**	
Algoritmo	**Promedio**	**Algoritmo**	**Promedio**
DL	1.93	DL	1.00
KNN-1	2.86	DT	2.86
KNN-5	4.13	RF	3.33
KNN-3	4.53	KNN-1	3.66
KNN-7	4.80	KNN-7	6.20
NB	5.26	SVM	6.40
RF	5.40	KNN-3	6.66
DT	6.86	KNN-5	6.93
SVM	8.66	NB	7.93

Pregunta 2: ¿Cuáles son las mejores representaciones textuales y qué algoritmos de clasificación funcionan mejor para la tarea de clasificación del semáforo epidemiológico a través de las noticias tomando en cuenta la cantidad de datos y el desbalance de clases?

Cuando se trabajó con noticias separadas, las mejores combinaciones siempre fueron con bigramas de palabras. Cuando se trabajó con las noticias unidas semanalmente, cuando $m = 0$ y a 4 el mejor resultado se obtuvo con unigramas y bigramas con unigramas respectivamente. En las demás combinaciones lo mejor fue obtenido con bigramas.

Esto da evidencia de que los bigramas son una mejor representación que los unigramas ya que captan de mejor manera el contenido del texto. También parece que la unión de los bigramas con los unigramas no representa una mejora importante por lo que parece que no vale la pena mezclar ambas características. Por otro lado, cuando se comparan los algoritmos de clasificación, cuando se analizan las noticias separadas, siempre el mejor algoritmo es el de Deep Learning (DL).

Esto sería posible porque existen más instancias que cuando se analizan las noticias unidas, que es cuando funcionan mejor este tipo de algoritmos. Para las noticias unidas semanalmente, el algoritmo que mejores resultados obtuvo fue KNN, apareciendo 10 veces de las 15 combinaciones posibles, como se puede ver en la Tabla 3. De estas 10, 8 veces fue utilizando k=1, y una vez k= 3 y finalmente una vez k=7.

Dos combinaciones tuvieron mejores resultados con el algoritmos de Decision Tree (DT) y 3 combinaciones con el algoritmo de Deep Learning. En la Tabla 6 se puede observar el promedio del ranking obtenido, por algoritmo, para cada una de las posibles combinaciones. Es decir, por cada experimento que se hizo, se otorgó un número entre 1 y 9 a los algoritmos de clasificación dependiendo el lugar que alcanzaron, donde 1 se le otorgaba al algoritmo con el F-measure más alto y 9 al más bajo.

Todos estos ranks se sumaron y se dividieron entre 15 (5 semanas, incluyendo a la semana cero multiplicado por 3 representaciones textuales de unigramas, bigramas y su combinación) para obtener el promedio del ranking de cada algoritmo.

Tabla 7. F-measure por clase de los mejores resultados pora los valores de m.

m	Rojo	Naranja	Amarillo	Verde
0	**0.85**	0.46	0.76	0.00
1	0.70	0.58	0.71	**1.00**
2	0.80	0.41	0.67	0.66
3	0.80	0.43	0.65	0.66
4	0.71	**0.62**	**0.78**	0.00

Tabla 8. Matriz de confusión para $m = 1$ con KNN-1 y bigramas.

	Rojo	Naranja	Amarillo	Verde
Rojo	7	2	0	0
Naranja	3	9	1	0
Amarillo	1	7	11	0
Verde	0	0	0	2

Esto se llevó a cabo para el análisis de las noticias unidas y separadas. Sorpresivamente, para las noticias unidas, aunque el algoritmo de KNN obtuvo mejores resultados en 10 de 15 combinaciones, el algoritmo de Deep Learning obtuvo un mejor promedio. Sin embargo, después de este algoritmo todas las variantes de KNN vienen detrás, siendo la mejor opción utilizar $k = 1$.

Es muy probable que conforme la base de datos siga creciendo, el algoritmo de Deep Learning empiece a tener mejores resultados que el resto de algoritmos. Por otra parte, para las noticias separadas, como ya se había dicho, el algoritmo de DL obtuvo siempre el primer lugar, sin embargo los algoritmos que le siguen en la tabla son Decision Tree (DT) y Random Forest (RF).

Pregunta 3: ¿Con cuántas semanas de anticipación se puede predecir el color del semáforo epidemiológico de tal manera que se obtenga un resultado razonable?

En la Tabla 4 Se observan los mejores resultados obtenidos para clasificar el semáforo epidemiológico. Estos resultados se obtienen analizando las noticias unidas de manera semanal. Es interesante ver que los peores resultados se obtienen cuando $m = 0$ y a 4, es decir, cuando se analiza el semáforo actual a las noticias y el semáforo del mes siguiente con un F-measure de 0.54 y 0.52 respectivamente.

Cuando $m = 2$ y 3 el resultado es de 0.63 que, considerando el desbalance de clases y los pocos datos es un resultado competitivo. El mejor resultado se obtiene con $m = 1$, es decir, cuando se predice el semáforo de la semana siguiente a la noticia, en este caso se obtiene 0.74 de F-measure. Este resultado se obtiene con KNN-1 y su ventaja es que es capaz de capturar, en buena medida, instancias minoritarias.

En la Tabla 7 se muestran los resultados de F-measure por cada clase para los mejores resultados en cada valor de m. En estos resultados se puede ver que los bajos resultados en $m = 0$ y 4 se deben a que no pudieron clasificar instancias de la clase verde, sin embargo obtienen buenos resultados para los otros colores.

Por otro lado cuando $m = 1$ se observa que el resultado se debe a que fue capaz de clasificar correctamente las instancias verdes. En la Tabla 8 se muestra la matriz de

confusión resultante, donde se puede ver que tanto con las instancias en rojo como en verde se tiene un buen rendimiento.

De esta manera es posible capturar información de los 4 colores al menos para $m = \{1, 2, 3\}$, sin embargo para $m = \{0, 4\}$, aunque no se obtienen buenos resultados para la clase verde, los demás colores tienen buen rendimiento.

5. Conclusiones y trabajo a futuro

En este trabajo de investigación se evaluaron diversas representaciones textuales y modelos de clasificación para determinar de manera automática el color del semáforo epidemiológico del estado de Veracruz. Los resultados dieron evidencia de que, de las noticias locales, es posible extraer información importante para poder alimentar clasificadores automáticos y generar modelos capaces de determinar el color del semáforo, incluso hasta con 4 semanas de antelación, siendo el mejor resultado obtenido a la semana 1 después de publicada una noticia.

Los mejores resultados se obtuvieron uniendo todas las noticias de una semana en un solo documento utilizando el color del semáforo como etiqueta. También, la mejor representación para esta tarea resultó ser la de bigramas de palabras. Los mejores clasificadores fueron una arquitectura de Deep Learning y el algoritmo de KNN.

También es importante ver que mientras se vaya alimentando estos modelos con información semanal, los resultados pueden mejorar. Como trabajo a futuro se propone extender estos modelos para los demás estados de la república mexicana. También se plantea implementar estrategias de fusión ya que muchos modelos se complementan entre sí.

Referencias

1. Apuke, O. D., Omar, B.: Fake news and covid-19: Modelling the predictors of fake news sharing among social media users. Telematics and Informatics, vol. 56 (2021) doi: 10.1016/j.tele.2020.101475
2. Basch, C. H., Hillyer, G. C., Meleo-Erwin, Z., Mohlman, J., Cosgrove, A., Quinones, N.: News coverage of the covid-19 pandemic: Missed opportunities to promote health sustaining behaviors. Infection, Disease and Health, vol. 25, no. 3, pp. 205–209 (2020) doi: 10.1016/j.idh.2020.05.001
3. Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., Nissim, M.: N-gram: New groningen author-profiling model (2017) doi: 10.48550/ARXIV.1707.03764
4. Cervantes-Holguín, E., Gutiérrez-Sandoval, P. R.: Resistir la COVID-19. Intersecciones en la educación de Ciudad Juárez, México, Revista Internacional de Educación para la Justicia Social, vol. 9, no. 3 (2020)
5. Crick, J. M., Crick, D.: Coopetition and COVID-19: Collaborative business-to-business marketing strategies in a pandemic crisis. Industrial Marketing Management, vol. 88, pp. 206–213 (2020) doi: 10.1016/j.indmarman.2020.05.016
6. Cury, R. C., Megyeri, I., Lindsey, T., Macedo, R., Batlle, J., Kim, S., Baker, B., Harris, R., Clark, R. H.: Natural language processing and machine learning for detection of respiratory illness by chest CT imaging and tracking of COVID-19 pandemic in the united states. Radiology: Cardiothoracic Imaging, vol. 3, no. 1 (2021) doi: 10.1148/ryct.2021200596

7. Escolá-Gascón, A.: New techniques to measure lie detection using COVID-19 fake news and the multivariable multiaxial suggestibility inventory-2 (MMSI-2). Computers in Human Behavior Reports, vol. 3 (2021) doi: 10.1016/j.chbr.2020.100049
8. Hu, Z., Ge, Q., Li, S., Jin, L., Xiong, M.: Artificial intelligence forecasting of COVID-19 in China (2020)
9. Iwendi, C., Ponnan, S., Munirathinam, R., Srinivasan, K., Chang, C. Y.: An efficient and unique TF/IDF algorithmic model-based data analysis for handling applications with big data streaming. Electronics, vol. 8, no. 11 (2019) doi: 10.3390/electronics8111331
10. Kolluri, N. L., Murthy, D.: CoVerifi: A COVID-19 news verification system. Online Social Networks and Media, vol. 22 (2021) doi: 10.1016/j.osnem.2021.100123
11. Salinas-Zuñiga, J. I.: Estudio descriptivo de la aplicación del semáforo epidemiológico y su efecto en el comercio de la Ciudad de Babahoyo, B.S., Thesis, Universidad Técnica de Babahoyo (2020)
12. Salisu, A. A., Vo, X. V.: Predicting stock returns in the presence of COVID-19 pandemic: The role of health news. International Review of Financial Analysis, vol. 71 (2020) doi: 10.1016/j.irfa.2020.101546
13. Thabtah, F., Hammoud, S., Kamalov, F., Gonsalves, A.: Data imbalance in classification: Experimental evaluation. Information Sciences, vol. 513, pp. 429–441 (2020) doi: 10.1016/j.ins.2019.11.004
14. Umadevi, S., Marseline, K. J.: A survey on data mining classification algorithms. In: Proceedings of the International Conference on Signal Processing and Communication, pp. 264–268 (2017) doi: 10.1109/cspc.2017.8305851
15. Vaishya, R., Javaid, M., Khan, I. H., Haleem, A.: Artificial intelligence (AI) applications for COVID-19 pandemic. Diabetes and Metabolic Syndrome: Clinical Research and Reviews, vol. 14, no. 4, pp. 337–339 (2020) doi: 10.1016/j.dsx.2020.04.012
16. Wang, H., Czerminski, R., Jamieson, A. C.: Neural networks and deep learning. The Machine Age of Customer Insight, Emerald Publishing Limited, pp. 91–101 (2021) doi: 10.1108/978-1-83909-694-520211010

ISSN 1870-4069

Análisis de expedientes clínicos para el diagnóstico de cáncer de mama a partir de memorias asociativas evolutivas: un primer avance

Juan Villegas-Cortez[1], Beatriz A. González-Beltrán[1], Fernando Torres-Vizueth[1], Salomón Cordero-Sánchez[2]

[1] Universidad Autónoma Metropolitana, Unidad Azcapotzalco, Departamento de Sistemas, México

[2] Universidad Autónoma Metropolitana, Unidad Iztapalapa, Departamento de Química, México

{juanvc, bgonzalez}@azc.uam.mx, al2192800401@alumnos.azc.uam.mx, scs@xanum.uam.mx

Resumen. La enfermedad del cáncer en todos sus tipos se sigue estudiando para poder entenderla mejor, dado que su padecimiento es de diversas formas y son muchos los factores que pueden relacionarse con un diagnóstico final de si una persona tiene o no un determinado tipo de cáncer. En este trabajo presentamos una primera propuesta de análisis del cáncer de mama, a partir de una base de datos de expedientes clínicos bien reconocida en el medio de la comunidad de estudio del reconocimiento de patrones. Se propone el uso de memorias asociativas evolutivas como herramienta de análisis desde el aprendizaje automático, que de acuerdo a la investigación realizada en el estado del arte de nuestro problema no ha sido usada hasta el momento, y estas han demostrado resultados prometedores. Nuestro objetivo es brindar un nuevo punto de vista de los factores de la enfermedad como componentes de los patrones; y analizar el comportamiento de la clasificación desde una base de datos conocida. Cabe señalar que no se busca una reducción de dimensiones del patrón, sino de arrojar luces de los factores posiblemente relacionados con la enfermedad.

Palabras clave: Memorias asociativas evolutivas, diagnóstico clínico, reconocimiento de patrones, programación genética.

Analysis of Clinical Records for Breast Cancer Diagnosis by Means of Evolutionary Associative Memories: A First Approach

Abstract. Cancer disease in all its types is still being studied in order to better understand it, given that its condition is of various forms and there are many factors that can be related to a final diagnosis of whether or not a person has a certain type of cancer. In this work, we present a first approach for the analysis

of breast cancer, based on a well-recognized database of clinical records in the pattern recognition study community. The use of evolutionary associative memories is proposed as an analysis tool from machine learning, which according to the research carried out in the state of the art has not been used so far, and these have shown promising results. Our goal is to provide a new point of view of cancer factors as components of patterns; and analyzing the classification behavior from a known database. It should be noted that it is not intended to reduce the size of the pattern, but rather to shed light on the factors possibly related to the disease.

Keywords: Evolutionary associative memories, pattern recognition, clinical records, genetic programming.

1. Introducción

El cáncer es una de las primeras causas de muerte en el mundo alcanzado 8,2 millones de muertes en 2012 [6]. Para el caso de México, entre enero y agosto de 2020 se tuvieron 683,823 muertes, de los cuales el 9 % fue debido a esta enfermedad (60,421). Un año antes, en 2019 se registraron 747,784 defunciones, de las cuales el 12 % (88,683) fue debido al cáncer [3]. En México, con datos de 2017, y considerando los diferentes tipos de cáncer, el cáncer de mama constituye la principal causa de morbilidad en la población de 20 años y más [3].

Existen diferentes bases de datos que han sido extraídas de expedientes clínicos electrónicos que pueden ser analizadas con el objetivo de reconocer los patrones que posiblemente estén relacionados con la enfermedad. En este trabajo se utilizó la base de datos para cáncer de mama de la Universidad de Wisconsin [14], una base de datos clásica para la comunidad de ciencia de datos.

En este trabajo presentamos una propuesta de análisis de una base de datos de cáncer, del tipo expediente clínico, desde la herramienta de una red neuronal artificial evolutiva específica para este tipo de patrones, la memoria asociativa evolutiva (MAE) [13], con la finalidad de brindar una perspectiva nueva sobre la caracterización de los patrones en sus componentes acerca de cuáles son relevantes para la clasificación, y por ende, buscando presentar a los profesionales de la salud un enfoque de cuáles parámetros o datos de los expedientes son los relevantes para poder determinar si el paciente tiene cáncer o no.

En la sección 2, presentamos el estado del arte de este problema de estudio, tanto desde el estudio de este tipo de bases de datos, como de las MAE como herramienta de análisis de datos desde el cómputo evolutivo con la programación genética. La metodología de análisis se describe en la sección 3, y nuestros experimentos con los resultados obtenidos en la sección 4. Finalmente, en la sección 5 compartimos nuestras conclusiones y líneas de trabajo futuro.

2. Estado del arte

El estudio de la enfermedad del cáncer, a partir de expedientes clínicos, se puede abordar desde la parte ingenieril, y por ende desde la inteligencia artificial, a partir de un análisis numérico de los datos de los expedientes conformados como patrones.

Tabla 1. Descripción de las diez variables de la DB, siendo las 9 primeras las características del expediente clínico, y la última la de clasificación del tumor como benigno o maligno.

No.	Nombre de la variable	Descripción	Dominio
1	Clump Thickness	Espesor del grupo	[1,10]
2	Uniformity of Cell Size	Uniformidad del tamaño de la célula	[1,10]
3	Uniformity of Cell Shape	Uniformidad de la forma de la célula	[1,10]
4	Marginal Adhesion	Adhesión marginal	[1,10]
5	Single Epithelial Cell Size	Tamaño de célula epitelial simple	[1,10]
6	Bare Nuclei	Núcleos desnudos	[1,10]
7	Bland Chromatin	Cromatina blanda	[1,10]
8	Normal Nucleoli	Nucléolos normales	[1,10]
9	Mitoses	Mitosis	[1,10]
10	Class	Clase	2:Benigno, 4:Maligno

Se tienen diversas bases de datos en repositorios tales como el de la Universidad de Californa, Irvine (UCI)[3], y de ahí es que se muestra en este trabajo la posibilidad de realizar un estudio a partir de una base de datos representativa. La base de datos (DB) usada en nuestro estudio para cáncer de mama es de la Universidad de Wisconsin y se denomina Breast Cancer Wisconsin [14].

Esta DB está compuesta de 699 instancias y de nueve atributos para el análisis: espesor del grupo, uniformidad del tamaño de la célula, uniformidad de la forma de la célula, adhesión marginal, tamaño de célula epitelial simple, núcleos desnudos, cromatina blanda, nucléolos normales y mitosis. Cada uno de estos atributos tienen un valor $\in [1, 10]$.

Para nuestro propósito, la DB está formada por patrones tipo vector-renglón de 10 variables-componentes o atributos, 9 son independientes y la entrada final, la componente número 10, indica si el tumor fue benigno (indicado con el número 2) o si es maligno (indicado con número 4). En la Tabla 1 se muestra esta descripción con detalle.

En [5], se propone la mejora en la precisión de la clasificación del diagnóstico de cáncer de mama. En este trabajo realizan la clasificación de características utilizando un algoritmo genético. Además, extraen la características óptimas utilizando el algoritmo Cost-Sensitive Support Vector Machine (CSSVM). Los autores utilizaron también el conjunto de datos de Wisconsin Breast Cancer y Wisconsin Breast Cancer Diagnosis. El resultado de la clasificación obtuvo un 95.7 % de precisión.

Las redes neuronales artificiales (RNA) se conciben como un paradigma de aprendizaje y procesamiento automático que está bio-inspirado en la forma en que se describió el funcionamiento del sistema nervioso de animales en la década de los 60s [7, 1]. Tal que las RNA se presentan como un sistema de interconexión de neuronas en una red que coopera para producir un estímulo de salida.

[3] UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

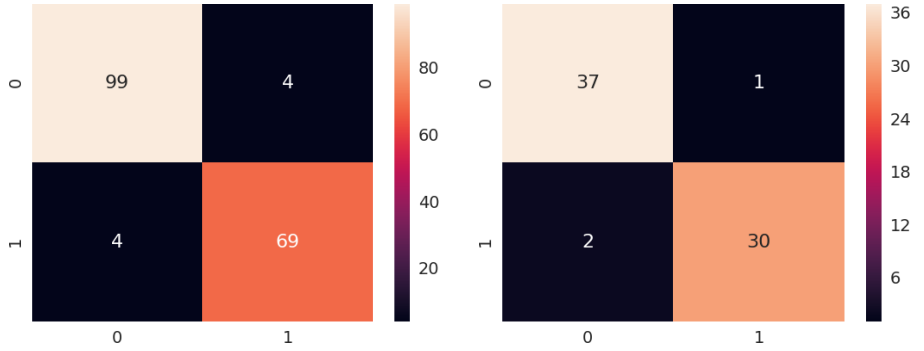


Fig. 1. Matrices de confusión de la primer prueba con el 75 % de entrenamiento, y 25 % de prueba (a), y de la segunda prueba (b), con partición del 90 % y 10 %, respectivamente.

Las RNA han sido diversificadas en su diseño buscando lograr clasificar patrones de clases que no son linealmente separables, así como de clases mezcladas, pero también proporcionan una herramienta para entender el problema de clasificación en su complejidad numérica ante el desafío del aumento de la cantidad de patrones, y de sus componentes. Hace 20 años, un dilema al momento de aplicar una RNA a un problema de reconocimiento de patrones era la cantidad de componentes de los patrones, los rasgos característicos del patrón.

Se buscaba tener una reducción de estos componentes con el objetivo de reducir la complejidad numérica de la RNA y por la limitante del hardware (cantidad de memoria a usar, los ciclos de reloj y la precisión numérica). Es así que se trabajaron nuevos modelos de RNA a partir del cómputo evolutivo [13, 10], buscando dos puntos fundamentales: por un lado, la reducción de la complejidad de la red en el número de capas y neuronas; y por otro lado, proporcionar una nueva perspectiva para entender los patrones de cada caso de estudio.

De entre los modelos RNA, es importante resaltar las memorias asociativas (MA), un tipo de red que no tiene arquitectura de capas, ni involucra la retro-propagación. Este modelo se enfoca en realizar una asociación de los patrones utilizando dos aspectos: “auto asociativo”, cuando se entrena la red asociando al patrón-vector consigo mismo, y “hetero-asociativo”, cuando al patrón se le asocia con otro. Un modelo más común de MA es el inspirado por las memorias morfológicas y la regla de aprendizaje de Hebb [9].

Una MAE se desarrolla a partir de la programación genética (PG), como un proceso co-evolutivo. Se realiza una primera evolución para la etapa de asociación, y se lleva a cabo otro proceso evolutivo, en co-evolución cooperativa con el primero, para la recuperación del patrón a partir de la MA construida en el primer proceso, y con una función de aptitud conjunta [12].

La PG es considerada un método automático para la creación de programas de cómputo como solución en alto nivel para el problema a solucionar [4]. Además, la PG se considera también una técnica de aprendizaje automático para optimizar una población de programas, de acuerdo a una función de aptitud que evalúa la capacidad de cada programa para resolver la tarea en cuestión.

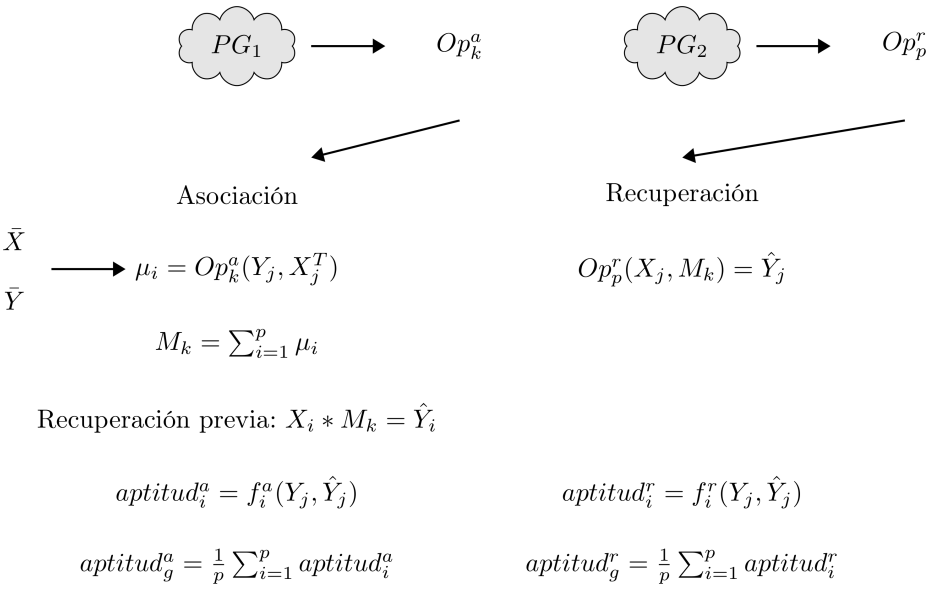


Fig. 2. Metodología co-evolutiva implementada para el desarrollo de MAE con programación genética.

En [13], se presenta una revisión del estado del arte de las MA a profundidad, y se muestra el detalle del desarrollo de la MAE, mostrando su efectividad para problemas en patrones reales y binarios, y también para el caso duro de patrones con ruido mixto y en problemas de visión artificial.

3. Metodología

Las MAE nos proporcionan en su análisis de datos una perspectiva de cuáles son las componentes de los patrones involucrados que tienen mayor importancia o preponderancia [13]; por lo anterior, nuestro problema consiste en analizar la enfermedad del cáncer desde la perspectiva numérica, ya que no somos profesionales de la salud.

Primero hicimos un análisis de clasificación de los patrones usando una RNA con multicapa clásica, tomando la DB de Wisconsin y, dando como valores en la capa de entrada las nueve variables de los patrones como vectores; para las capas internas se consideraron tres capas de 100 neuronas y, finalmente, una sola salida para la clasificación si el tumor es benigno o maligno. Se realizó una primera prueba, con una partición del 75 % de los patrones para entrenamiento y 25 % para prueba, donde se obtuvo un porcentaje de clasificación del 96 %.

Posteriormente, se realizó una segunda prueba, ahora con una partición del 90 % de los patrones por clase para entrenamiento, y con 10 % de los patrones para prueba, obteniendo un porcentaje de clasificación del 95 %. En la RNA se trabajó con el optimizador de descenso de gradiente estocástico (sgd) para el cálculo del mínimo de la función de costo.

Tabla 2. Resumen de los parámetros evolutivos involucrados en las pruebas.

Caso	Métrica en función de aptitud	Número de MAE a generar	Generaciones	Individuos por generación
I	Arco coseno	5	20	10
II	Error cuadrático medio	5	20	17

Tabla 3. Resumen de las duplas como MAE generadas para el caso I.

MAE índice	Regla de Asociación	Regla de Recuperación	Aptitud
1	1	1	100 %
2	1	2	100 %
3	2	3	99.8 %
4	1	4	100 %
5	2	5	100 %

Estos resultados se aprecian mejor en la Figura 1 con las matrices de confusión de la clasificación, donde los porcentajes están redondeados a valores enteros de la cantidad de patrones a considerar para caso de prueba. Como podemos ver los resultados obtenidos de la clasificación son bien conocidos en el medio de estudio de las RNA, luego ahora la propuesta es estudiar a los patrones en sí, con sus nueve características clínicas reportadas, en una MAE usando la auto-asociación.

3.1. Parámetros evolutivos para la PG

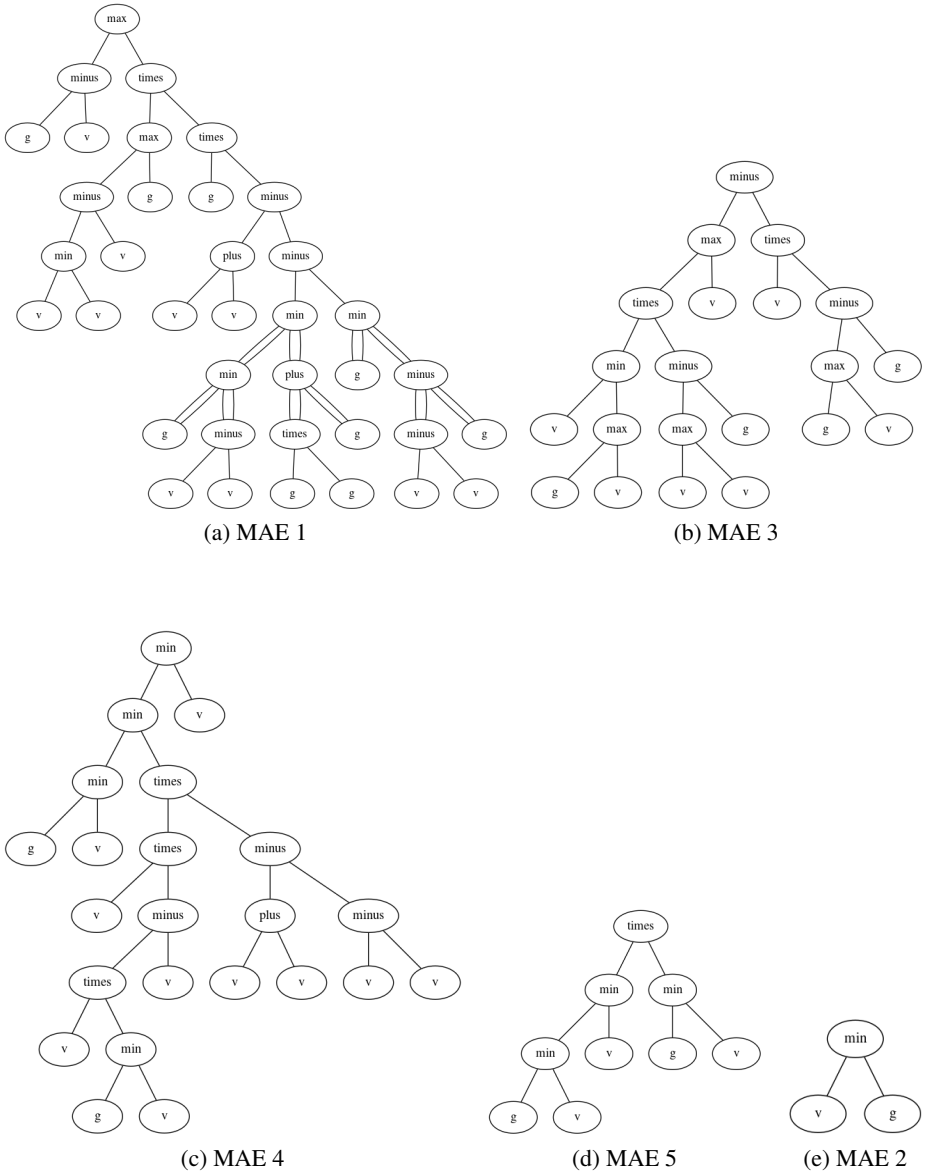
Se plantea desde la PG que ahora el individuo es un programa tentativo a dar solución al problema de hallar una asociación del patrón, cada individuo se asocia consigo mismo, tal que se genera un "dispositivo" de almacenamiento de sus características. Esto es, presentando un patrón de entrada se tiene la recuperación del mismo, diferenciándolo de otros patrones de ese repositorio, representado como un "conocimiento" en la MAE.

En el proceso co-evolutivo, se trabajaron poblaciones con regeneración en cada ciclo evolutivo con cruza al 70 % y mutación del 10 % del individuo para generar un 30 % de individuos mutados de la nueva generación, preservando con elitismo al mejor individuo de cada generación.

Este criterio es con base a la experiencia de analizar otros problemas de reconocimiento de patrones con este tipo de proceso evolutivo de la PG. Cabe señalar que, como criterio de paro está el alcanzar el 100 % de recuperación, o tener cero error, o bien agotar el número de generación en co-evolución.

En la etapa de la asociación se trabajan operaciones a nivel escalar, con las entradas del patrón, tal que el conjunto terminal de la auto-asociación es $T_a = \{x_i, y_i\}$, para los patrones de entrada y salida $\{X, Y\}$, respectivamente; y el conjunto de funciones en esta etapa es: $F_a = \{+, -, \min, \max, \text{times}\}$.

La función de aptitud local, aptitud$_g^a$, se aplica a considerar el porcentaje de recuperación por pareja asociada, luego el promedio de ellas. En la Figura 2, se muestra el detalle tanto para el proceso de asociación como de recuperación.

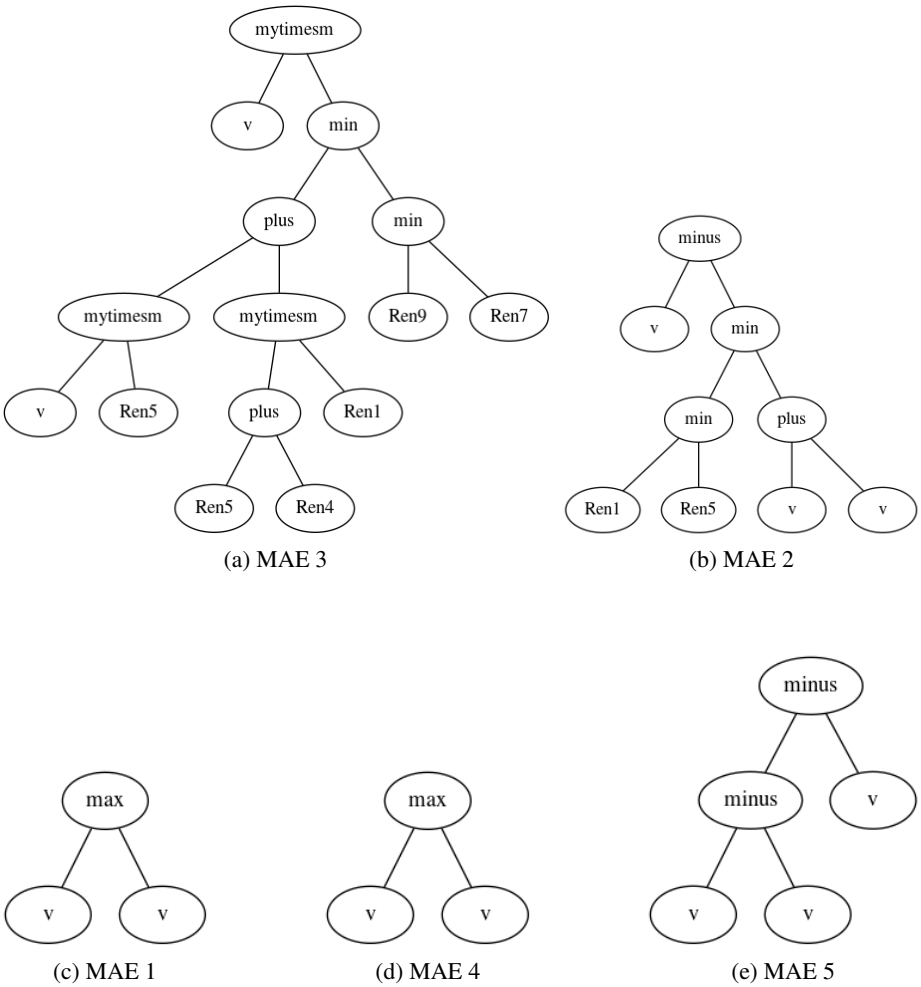


(a) MAE 1 (b) MAE 3

(c) MAE 4 (d) MAE 5 (e) MAE 2

Fig. 3. Reglas de asociación de patrones de las MAE para el caso I.

Para la etapa de recuperación se trabajan las operaciones considerando a los renglones de la matriz de asociación generada en la primera parte, adicionales a los patrones de asociación involucrados, v, y a la misma matriz de asociación generada, M_k, tal que ahora el conjunto de terminales es $T_r = \{v,\ \text{Ren}_1,\ \text{Ren}_2,\ ...,\ \text{Ren}_m,\ M_k\}$, y la función de aptitud co-evolutiva es aptitud$_g^r$. En [13], se tiene mayor detalle de la operación y metodología de las MAE.



(a) MAE 3 (b) MAE 2

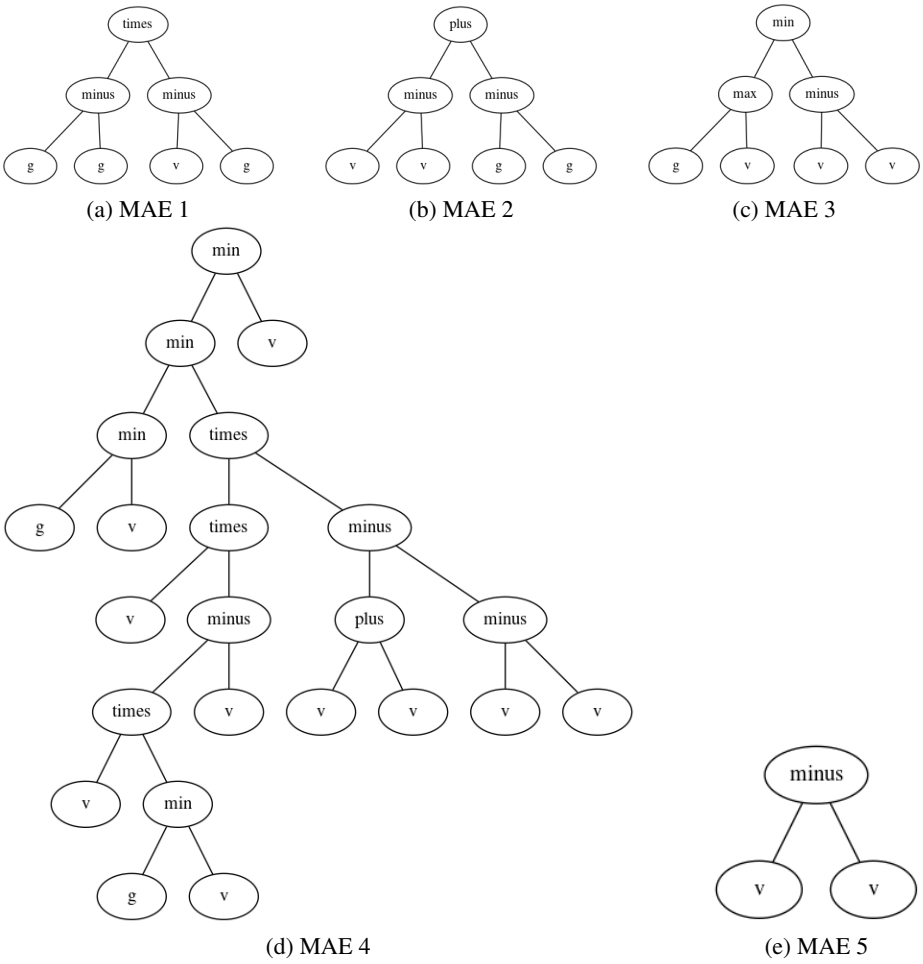
(c) MAE 1 (d) MAE 4 (e) MAE 5

Fig. 4. Reglas de recuperación de patrones de las MAE para el caso I.

4. Experimentos y resultados

La implementación se realizó en una computadora tipo WorkStation con procesadores Intel Xeon 64-bit, con sistema operativo Linux y Matlab con Toolbox GPLab [8], versión 3.0. Realizamos dos experimentos de auto-asociación con las MAE sobre la DB buscando brindar dos aportaciones de cómo se comportan estos patrones.

El primer caso (I), fue considerando a la métrica del arco coseno como función de aptitud, y el segundo caso (II), fue aplicando la medida del error cuadrático medio. La intención del caso I fue medir la similitud de los patrones en un espacio multidimensional, dada la limitante del arco coseno [2]; y la intención del caso II fue analizar la posibilidad de cercanía de estos patrones en ese espacio dimensional usando el error cuadrático medio.



(a) MAE 1 (b) MAE 2 (c) MAE 3

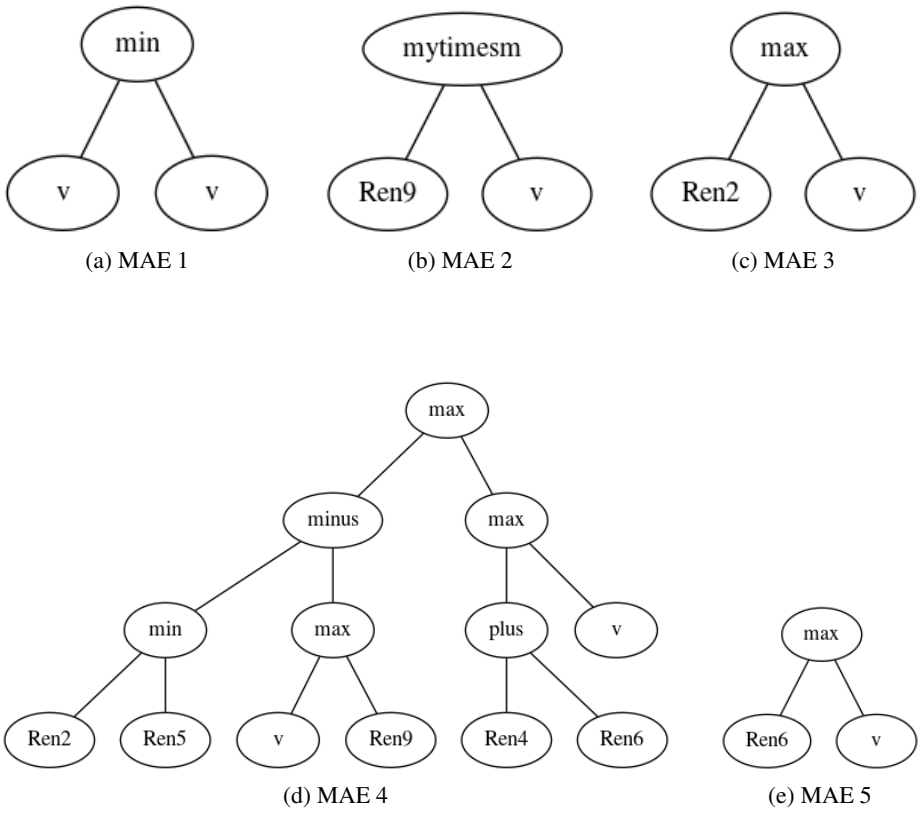
(d) MAE 4 (e) MAE 5

Fig. 5. Reglas de asociación de patrones de las MAE para el caso II.

En la Tabla 2, se muestra el resumen de valores para ambas pruebas. El número de individuos y generaciones se considera con base a la experiencia del estado del arte [13, 11].

4.1. Resultados para el caso I

Para el primer caso, la Figura 3 muestra las reglas de asociación obtenidas al final del proceso co-evolutivo de las 5 MAE generadas; mientras que la Figura 4 presenta las reglas de recuperación halladas. También podemos ver en el resumen de la Tabla 3 que, de las cinco duplas obtenidas, cuatro de ellas logran una recuperación perfecta, en este caso, una auto-asociación por similitud de los patrones con esas reglas de asociación y recuperación respectivas.



(a) MAE 1 (b) MAE 2 (c) MAE 3

(d) MAE 4 (e) MAE 5

Fig. 6. Reglas de recuperación de patrones de las MAE para el caso II.

Analizando los individuos-árboles generados para la recuperación de la Figura 4, la variable v representa el vector de entrada a recuperar, cada uno de la DB, y acorde a los resultados de la Tabla 3, cuatro MAE lograron recuperación perfecta por asociación de similitud, mientras que uno casi lo logra, la tercer dupla, que curiosamente involucra al individuo más complejo, el árbol con más nodos de la figura.

A partir de estos individuos, los resultados nos reflejan que son necesarios los nueve rasgos de la DB para lograr la recuperación perfecta, esto es en las duplas MAE con índice 1, 4 y 5; mientras que el individuo 2 involucra únicamente los rasgos del renglón 1 y 5, que son el espesor del grupo y el tamaño de la célula epitelial simple.

4.2. Resultados para el caso II

Ahora, analizando el resultado para el caso segundo, en la Tabla 4 se muestran los valores de recuperación obtenidos. Tras una búsqueda exhaustiva, por parte del proceso evolutivo, se hallaron cuatro MAE con recuperación al 100 %, y una, la segunda dupla con recuperación muy mala, del orden del 41.46 %. Aún así, en las Figuras 5 y 6 se muestran los árboles de las reglas de asociación y recuperación correspondientes.

Tabla 4. Resumen de las duplas como MAE generadas para el caso II.

MAE índice	Regla de Asociación	Regla de Recuperación	Aptitud
1	1	1	100 %
2	1	2	41.46 %
3	1	3	100 %
4	1	4	100 %
5	1	5	100 %

De este proceso evolutivo, para el caso de la auto-asociación por error cuadrático medio, en la Figura 6 podemos visualizar que los renglones 2, 6 y 9 son los más relevantes de las nueve características comprendidas de la DB, a saber de la descripción en la Tabla 1, los rasgos involucrados son: uniformidad de tamaño de célula, el núcleo desnudo, y la mitosis. Esto se visualiza en los árboles generados MAE 3, 4 y 5, el árbol-regla de recuperación 2 lo descartamos por no haber logrado la recuperación perfecta de acuerdo a los valores mostrados en la Tabla 4. De estos resultados obtenidos, las relaciones de asociación arrojados por las MAE son meramente especulativas a este nivel, el numérico, y lo comentamos porque nosotros como autores de este trabajo de análisis no somos profesionales de la salud.

5. Conclusiones

En este trabajo hemos presentado un primer avance en el análisis de expedientes clínicos para el diagnóstico de cáncer de mama, a partir de las memorias asociativas evolutivas, usando una base de datos reconocida en el medio de investigación del reconocimiento de patrones. Si bien es una DB con apenas 9 rasgos para tipificar si se tiene o no la enfermedad, con la clase tumor maligno o benigno, nos proporciona una herramienta poder adentrarnos en el estudio de este tipo de patrones para dar un resultado que pudiera ser de apoyo para los profesionales de la salud desde una técnica de inteligencia artificial.

Los resultados numéricos presentados no pretenden afirmar lo que pueda relacionar un médico o profesional de la salud, insistimos en esto reconociendo nuestras limitaciones como profesionales con formación en ingeniería y ciencias básicas. Las relaciones de asociatividad presentadas en este artículo sobre los rasgos característicos de la DB sobre el expediente de un paciente, nos indican una posible dirección de cómo seguir apoyando a los profesionales de la salud, esto también nos invita a poder extender esta línea de aplicación hacia bases de datos más extensas tanto en rasgos como en pacientes.

Como trabajo futuro planteamos abordar DB sobre enfermedades que apremian un estudio de apoyo, específicamente COVID-19 y diabetes, esta última enfermedad en México, nuestro país, donde 1 de cada 10 personas la padecen, y todo apunta a que no cambiará este dato estadístico.

Agradecimientos. Este trabajo es resultado del proyecto divisional “Evolución artificial de descriptores estadísticos de textura de superficie para implementación en clasificación de imágenes digitales”, clave: EL006-18, de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco.

Referencias

1. Fausett, L.: Fundamentals of neural networks: Architectures, algorithms, and applications. Prentice-Hall, Inc, Upper Saddle River (1994)
2. Fukunaga, K.: Introduction to statistical pattern recognition (2nd ed.). Academic Press Professional, Inc (1990)
3. INEGI. Estadísticas a propósito del día mundial contra el cáncer. Comunicado de prensa número 105/21 (2021) https://www.inegi.org.mx/contenidos/saladeprensa/aproposito/2021/cancer2021_Nal.pdf
4. Koza, J. R.: Genetic programming as a means for programming computers by natural selection. Statistics and Computing, Springer Science and Business Media LLC, vol. 4, no. 2 (1994) doi: 10.1007/bf00175355
5. Liu, N., Qi, E. S., Xu, M., Gao, B., Liu, G. Q.: A novel intelligent classification model for breast cancer diagnosis. Information Processing and Management, vol. 56, no. 3, pp. 609–623 (2019) doi: 10.1016/j.ipm.2018.10.014
6. Organizacion Mundial de la Salud. Datos y cifras sobre el cáncer, Página web Organización Mundial de la Salud, www.who.int/cancer/about/facts/es/
7. Rojas, R., Feldman, J.: Neural networks: A systematic introduction. Springer (1996) doi: 10.1007/978-3-642-61068-4
8. Silva, S., Almeida, J.: GPLAB - A genetic programming toolbox for MATLAB. In: Proceedings of the Nordic MATLAB Conference, pp. 273–278 (2003)
9. Sossa, H., Barrón, R., Vázquez, R. A.: New associative memories for recall real-valued patterns. Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp. 195–202 (2004) doi: 10.1007/978-3-540-30463-0_24
10. Sossa, H., Garro, B.A., Villegas, J., Olague, G., Avilés, C.: Evolutionary computation applied to the automatic design of artificial neural networks and associative memories. In: Advances in Intelligent Systems and Computing, Springer Berlin Heidelberg, pp. 285–297 (2013) doi:10.1007/978-3-642-31519-0_18
11. Villegas-Cortez, J., Olague, G., Aviles, C., Sossa, H., Ferreyra, A.: Automatic synthesis of associative memories through genetic programming: A first co-evolutionary approach. Applications of Evolutionary Computation, Springer Berlin Heidelberg, pp. 344–351 (2010) doi: 10.1007/978-3-642-12239-2_36
12. Villegas-Cortez, J., Olague, G., Sossa, H., Avilés, C.: Evolutionary associative memories through genetic programming. Parallel Architectures and Bioinspired Algorithms, Studies in Computational Intelligence, Springer, vol. 415, pp. 171–188 (2012) doi: 10.1007/978-3-642-28789-3_8
13. Villegas-Cortez, J.: Síntesis automática de memorias asociativas mediante programación genética. Tésis Doctoral, Instituto Politécnico Nacional, Centro de Investigación en Computación (2009)
14. Wolberg, W. H., Mangasarian, O. L.: Multisurface method of pattern separation for medical diagnosis applied to breast cytology. In: Proceedings of the National Academy of Sciences, vol. 87, no. 23, pp. 9193–9196 (1990) doi: 10.1073/pnas.87.23.9193

ISSN 1870-4069

Implementación del modelo matemático espacio–temporal para el COVID-19 en México

María Beatríz Bernábe Loranca[1],
Armando Benjamín Cruz Hinojosa[2]

[1] Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación,
México

[2] Universidad Nacional Autónoma de México,
Facultad de Ciencias,
México

beatriz.bernabe@gmail.com,
aleph_g@ciencias.unam.mx

Resumen. En este trabajo se presenta la implementación computacional de un modelo matemático-epidemiológico que estima distintas variables sobre el problema del virus SARS-COV2 que produce la enfermedad COVID-19. El modelo considera tanto los datos estadísticos del INEGI como los datos epidemiológicos de la Dirección General de Epidemiología. La implementación del modelo consiste en un programa que genera estimaciones del número de infecciones, el número de pacientes hospitalizados en una Unidad de Cuidados Intensivos UCI, el número de muertes y la tasa de contagio. La importancia del desarrollo del modelo matemático consiste en una herramienta que estima los contagios de COVID-19 en distintas regiones de México.

Palabras clave: COVID, modelo matemático, predicción, python.

Mathematical Spatiotemporal Model Implementation for COVID-19 in Mexico

Abstract. This paper presents the computational implementation of a mathematical-epidemiological model that estimates different variables on the SARS-COV2 virus problem that produces COVID-19 disease. The model considers both statistical data from INEGI and epidemiological data from the General Directorate of Epidemiology. The implementation of the model consists of a program that generates estimates of the number of infections, the number of patients hospitalized in an Intensive Care Unit (ICU), the number of deaths and the infection rate. The implication of the development of the mathematical model consists of a tool for the visualization of COVID-19 infections in different regions of Mexico.

Keywords: COVID, mathematical model, prediction, python.

1. Introducción

Un año ha pasado desde que comenzó la emergencia sanitaria mundial ocasionada por el virus SARS- CoV-2 pero los contagios y muertes no han cedido. El reto social ha sido dramático, incidiendo en la calidad de vida y cambiando la cotidianidad de las personas alrededor del mundo. A pesar de los esfuerzos, se continúa amenazando la integridad de los hospitales y sistemas de salud de las naciones.

Para enfrentar la crisis, se requieren tomar decisiones y generar medidas de políticas públicas precisas fundamentadas en información actualizada y tecnológica que permita sistematizar la información que se genera de distintas fuentes de datos confiables. En este trabajo se presenta la implementación computacional de un modelo matemático para la propagación de la epidemia espacio-temporal del COVID-19, el cual se adaptó a la realidad mexicana usando información estadística del INEGI y los datos abiertos de la DGE (Dirección General de Epidemiología) sobre los casos de COVID-19 en México.

El software desarrollado produce estimaciones de número de contagios, número de hospitalizaciones, número de muertes y tasa de contagios. El modelo formulado en este artículo se ha inspirado de un modelo español y se adaptó a la situación Mexicana. El modelo que se presenta se implementó en el lenguaje de programación Python [1]. Los coronavirus son una familia de virus que causan infección en seres humanos y en animales como camellos, gatos y murciélagos.

Los coronavirus que afectan al ser humano (HCoV) pueden provocar desde el resfriado común hasta otros más graves como los producidos por los virus del Síndrome Respiratorio Agudo Grave SARS y del Síndrome Respiratorio de Oriente Próximo (MERS-CoV). La primera aparición del SARS en 2003, ocasionó más de 8.000 casos en 27 países con mortalidad del 10 %. Por otro lado, desde 2012 se han reportado 2499 casos de MERS-CoV en 27 países, principalmente en Arabia Saudita hasta con una letalidad de 34 % [2].

El virus que enfrentamos desde el año 2020, es un coronavirus en un sentido amplio. Sin embargo, dentro de la taxonomía de los coronavirus, el virus actual pertenece a la categoría Beta, su nombre oficial es SARS-CoV-2, el cual produce la enfermedad COVID-19 en humanos. El SARS-CoV surgió en 2002 en China aproximadamente después del HCoV-HKU1, sin embargo, el SARS-CoV generaba un síndrome respiratorio agudo en grupos de riesgo. Por su carácter nocivo, produjo un brote que infectó a más de 8.400 personas en 26 países entre Asia, Europa y América.

Se registraron 800 muertes traducidas en una letalidad del 9,6 %. La pandemia fue controlada año y medio después [3]. A partir de este problema de salud mundial, muchos son los desafios en dar solución al aplanamiento de la curva de contagios, la cual obedece a que las medidas de restricción puedan ser modeladas en la matemática y la computación. Es en este punto donde se inserta el desarrollo del proyecto: Se describe un modelo matemático propuesto para mapear la evolución de la pandemia de 2020 a nivel nacional ante múltiples escenarios. El presente artículo se encuentra organizado en las siguientes secciones:

1. Introducción.
2. Modelo de esparcimiento epidémico.
3. Búsqueda y extracción de datos nacionales relacionados al COVID-19.

4. Implementación computacional de un modelo matemático flexible para predecir el comportamiento de la epidemia a nivel nacional, estatal y municipal en diferentes escenarios.
5. Hacia la interpretación de datos en un SIG.
6. Discusión de resultados y conclusiones.

2. Modelo de esparcimiento epidémico

Desde que la pandemia sorprendió a principios del 2020, distintos investigadores tanto de manera individual como en grupos y grandes empresas se dieron a la tarea de desarrollar propuestas diversas para dar respuesta al problema de salud del COVID-19. Respecto a estudios de cohorte retrospectivo y multicéntrico para adultos enfermos hospitalizados de COVID-19 y mayores a 18 años de edad en Wuhan-China, se analizaron los enfermos que habían sido dados de alta o habían fallecido antes del 31 de enero de 2020.

Se extrajeron datos demográficos, clínicos, de tratamiento y de laboratorio, incluidas muestras seriadas para la detección de ARN viral y se compararon entre supervivientes y no supervivientes para identificar los factores de riesgo asociados con la muerte intrahospitalaria. Para ello los autores utilizaron métodos de regresión logística univariable y multivariable [4].

Otros hallazgos metodológicos se ocupan de 3 prioridades iniciales para estimaciones de contagios y decesos categorizadas como: a) locales (la población, movilidad, morbilidad de la zona, cultura y economía), b) globales (impacto en otras zonas) y c) totales (casos confirmados, casos recuperados fallecimientos etc.). Para ello se utiliza el modelo SIR (Susceptible, Recuperada, Infectada) y técnicas multicriterio [5].

Recientemente una propuesta de simulador considera 13 variables extraídas de la secretaria de salud en México. El software simulador genera distintos escenarios para alertar a la población si las medidas de salud se relajan [6]. Por otra parte, los datos de la (Organización Mundial de la Salud) fueron procesados para la predicción del comportamiento de los contagios por COVID-19. Los datos adquiridos se sometieron a una estimación que utiliza una función logística para generar pronósticos de contagios con buena aproximación [5, 7].

En este apartado se describe el modelo de esparcimiento epidémico de COVID-19 descrito en el artículo de Arenas-Cota y Gómez-Gardeñes [1]. En tal manuscrito, los autores se basan en un estudio previo de evolución de pandemias en metapoblaciones estructuradas que incluyen agentes sujetos a patrones de mobilidad recurrentes. Es decir, se describe matematicamente que la dispersión de la enfermedad en una región es estimada en pequeñs grupos geográficos, donde el resultado se apoya de la observación de interacción entre personas que se derivan de la movilidad entre los grupos.

La idea subyacente da lugar a introducir el concepto de meta-población, las cuales son creadas a partir de una población que se distribuye en una colección de grupos poblacionales bien definidos, de distinto número de individuos. Los individuos dentro de un grupo están mezclados y el patógeno puede ser transmitido entre cualesquiera agentes del grupo con la misma probabilidad.

El otro aspecto importante de la meta-población es la movilidad de los agentes, cada individuo tiene la posibilidad de migrar de su grupo a otro, esparciendo el patógeno a nivel del sistema. La movilidad de estos agentes se representa como una red en la que cada nodo es un grupo y flechas entre ellos representan la posibilidad de moverse de un grupo a otro. Esta información de importancia sustancial, se codifica en una matriz denominada matriz de movilidad, R.

Las distintas investigaciones epidemiológicas relevantes al COVID-19, sugieren que se agreguen propiedades a los grupos poblacionales para simular adecuadamente la forma en la que el patógeno se propaga. Una de estas consiste en separar dentro de cada grupo a los individuos en distintos estratos de edad como jóvenes, adultos y ancianos. Esta consideración es relevante dado que se ha observado que personas jóvenes reaccionan de formas distintas al virus respecto a las personas mayores.

Esta situación mal atendida podría implicar consecuencias catastróficas a la salud. La otra propiedad en los grupos es que cada individuo puede pertenecer a uno de ocho grupos posibles: susceptible (S), expuesto (E), asintomático (A), infectado (I), hospitalizado en una unidad de cuidados intensivos (H), muerto (D), y recuperado (R). La evolución de las fracciones de agentes en el estado $m \in \{S, E, A, I, H, D, R\}$ y estrato de edad $g \in \{1, 2, 3\}$ en el grupo $i \in \{1, \dots N\}$, denotadas por $\rho_i^{m,g}(t)$ esta dada por:

$$\rho_i^{S,g}(t+1) = (1 - \Pi_i^g(t))\rho_i^{S,g}(t), \tag{1}$$

$$\rho_i^{E,g}(t+1) = \Pi_i^g(t)\rho_i^{S,g}(t) + (1-\eta^g)\rho_i^{E,g}(t), \tag{2}$$

$$\rho_i^{A,g}(t+1) = \eta^g\rho_i^{E,g}(t) + (1-\alpha^g)\rho_i^{A,g}(t), \tag{3}$$

$$\rho_i^{I,g}(t+1) = \alpha^g\rho_i^{A,g}(t) + (1-\mu^g)\rho_i^{I,g}(t), \tag{4}$$

$$\rho_i^{H,g}(t+1) = \mu^g\gamma^g\rho_i^{I,g}(t) + \omega^g(1-\psi^g)\rho_i^{H,g}(t) + (1-\omega^g)(1-\chi^g)\rho_i^{H,g}(t), \tag{5}$$

$$\rho_i^{D,g}(t+1) = \omega^g\psi^g\rho_i^{H,g}(t) + \rho_i^{D,g}(t), \tag{6}$$

$$\rho_i^{R,g}(t+1) = \mu^g(1-\gamma^g)\rho_i^{I,g}(t) + (1-\omega^g)\chi^g\rho_i^{H,g}(t) + \rho_i^{R,g}(t). \tag{7}$$

Estas ecuaciones representan la dinámica en tiempo discreto de los agentes (cada paso temporal representa un día), y están basadas en trabajos previos en dinámicas de pandemias con enfoque en cadenas de markov microscópicas. Los parámetros involucrados en estas ecuaciones están descritos en la Tabla 1, a excepción de $\Pi_i^g(t)$, que representa la probabilidad de que un agente susceptible sea contagiado por un agente expuesto o infectado, convirtiéndose en un agente expuesto. La expresión analítica de dicho valor es:

$$\Pi_i^g(t) = (1-p^g)P_i^g(t) + p^g\sum_{j=1}^{N} R_{i,j}P_j^g(t), \tag{8}$$

donde $P_i^g(t)$ es la probabilidad que agentes de estrato de edad g se infecten del patógeno en el grupo poblacional i y p^g es la probabilidad de movilidad de los agentes en el estrato de edad g. Para determinar $P_i^g(t)$ se asume que los contactos en el grupo i incrementan monotónicamente según la densidad poblacional, como dicta la función $f(x) = 1 + (1 + e^{-\xi x})$, además se considera la matriz de contactos C cuyas entradas

$C_{g,h}$ definen la fracción de contactos que un individuo el el estrato de edad g tiene con individuos en el estrato de edad h. De esta forma la fórmula para calcular $P_i^g(t)$ se define como:

$$P_i^g(t) = 1 - \prod_{h=1}^{3}\prod_{j=1}^{N}(1-\beta_A)^{z^g k^g f\left(\frac{n_i^{eff}}{s_i}\right) C_{g,h}\left(\frac{n_{j\to i}^{A,h}(t)}{(n_i^h)^{eff}}\right)}(1-\beta_I)^{z^g k^g f\left(\frac{n_i^{eff}}{s_i}\right) C_{g,h}\left(\frac{n_{j\to i}^{I,h}(t)}{(n_i^h)^{eff}}\right)}. \quad (9)$$

Los exponentes de las bases $(1-\beta_A), (1-\beta_I)$ en la Ecuación 9 representan el número de contactos estimado de un agente en el grupo i en el estrato g con pacientes asintomáticos e infectados respectivamente. De esta forma el doble producto es la probabilidad que un agente no se infecte después de dichos contactos. Es importante explicar que el término $z^g k^g f(n_i^{eff}/s_i)$ es el número promedio de contactos de un agente, ya sea con agentes infectados o no infectados, y z^g es un factor normalizante del efecto de la función $f(x)$, este se calcula como:

$$z^g = \frac{3}{\sum_{i=1}^{N} f\left(\frac{n_i^{eff}}{s_i}\right)(n_i^g)^{eff}}. \quad (10)$$

Y las poblaciones efectivas representan la cantidad esperada de agentes activos en el grupo poblacional i, tomando en consideración las migraciones:

$$n_i^{eff} = \sum_{g=1}^{3}(n_i^g)^{eff}, \quad (11)$$

$$(n_i^g)^{eff} = p_g \sum_{j=1}^{N} R_{j,i} n_j^g + (1-p_g) n_i^g. \quad (12)$$

Finalmente, los últimos miembros de los exponente en la Ecuación 9 son $n_{j\to i}^{A,h}(t)$ y $n_{j\to i}^{I,h}(t)$, los cuales representan el número de agentes del estrato de edad h en estado asintomático o infectado y que han migrado de la región j a la región i. Y pueden ser calculados de la siguiente forma:

$$n_{j\to i}^{m,h}(t) = n_j^h \rho_j^{m,h}(t)\left[(1-p^h)\delta(i,j) + p^h R_{j,i}\right]. \quad (13)$$

Con $\delta(i,j)$ la función delta de Kronecker. El artículo original señala en su formulación políticas de aislamiento mediante un porcentaje k_0 de población en aislamiento y se han omitido los detalles, sin embargo se recomiendan las lecturas [1, 4, 5]. El modelo formulado aquí puede ser aplicado a la dinámica espacio temporal de otros fenómenos, como la transmisión de información y congestión del tráfico.

3. Búsqueda y extracción de datos

En la Sección 2 se menciona que una de las bondades del modelo reside en la posibilidad de modificar las variables para ajustarlo a la realidad de una población particular a tratar, definiendo una macro-población adecuada.

Tabla 1. Parámetros del modelo.

Símbolo	Significado	Valor estimado en España
β_A	Infectividad de pacientes asintomáticos	$0{,}06$
β_I	Infectividad de pacientes infectados	$0{,}06$
$\langle k^g \rangle$	Número promedio de contactos	$(11{,}8, 13{,}3, 6{,}6)$
η^g	Tasa de latencia	$\frac{1}{2{,}34}$
α^g	Tasa de infección asintomática	$\left(\frac{1}{5{,}06}, \frac{1}{2{,}86}, \frac{1}{2{,}86}\right)$
μ^g	Tasa de salida	$\left(\frac{1}{1{,}0}, \frac{1}{3{,}2}, \frac{1}{3{,}2}\right)$
γ^g	Porcentaje de casos necesitados de UCI	$(0{,}002, 0{,}05, 0{,}36)$
ω^g	Tasa de fatalidad en UCI	$0{,}42$
ψ^g	Tasa de mortalidad	$\frac{1}{7{,}0}$
χ^g	Tasa de descarga de UCIs	$\frac{1}{10{,}0}$
C	Matriz de contactos	$\begin{pmatrix} 0{,}5980 & 0{,}3849 & 0{,}01711 \\ 0{,}2440 & 0{,}7210 & 0{,}0350 \\ 0{,}1919 & 0{,}5705 & 0{,}2376 \end{pmatrix}$
ξ	Factor de densidad	$0{,}01$
$\langle p^g \rangle$	Factor de movilidad	$(0{,}0, 1{,}0, 0{,}0)$
σ	Habitantes promedio por hogar	$2{,}5$
k_0	Porcentaje de confinamiento	Ajustable

La macro-población para México consiste en tomar como grupo poblacional a cada uno de los municipios de la República. Esta conceptualización se justifica porque las interacciones económicas y sociales en cada municipio obligan a los individuos a estar en constante contacto, cumpliendo la hipótesis de ser un grupo bien mezclado donde el patógeno puede ser transmitido en la misma probabilidad. En esta sección se describen los métodos de obtención de datos necesarios para esta macro-población.

3.1. Datos pandémicos de México

Los datos corresponden a los casos de COVID-19 en México, los cuales se han descargado de la página de datos abiertos del gobierno mexicano [7,8,9]. En este sitio también se consigue información proveniente de la DGE sobre las pruebas de COVID-19 realizadas a nivel nacional. La información sobre COVID-19 respecto a los pacientes, contiene la siguiente información:

– Entidad y municipio de residencia del paciente.
– Edad del paciente.
– Resultado de la prueba (negativo/positivo).

– Tipo de paciente (ambulatorio/hospitalizado).
– Si el paciente requirió de cuidados intensivos (sí/no).
– Si el paciente fue intubado (sí/no).
– Fecha de inicio de síntomas en el paciente.
– Fecha de ingreso a hospitalización (Si es que ingresó).
– Fecha de defunción (Si es que el paciente falleció).

De acuerdo con las características de estos datos, una muestra representativa del problema es tomada para el ejercicio del modelo y los datos se consideran como epidemiológicos para el modelo matemático formulado en este trabajo.

3.2. Información demográfica

La información demográfica que nutre al modelo: a) La población total presente en la región. b) La densidad poblacional de la región. c) El número promedio de habitantes por hogar. Estos parámetros se valoran a nivel nacional, estatal y municipal tal que las cifras son obtenidas del INEGI (Instituto Nacional de Estadística y Geografía). En este trabajo se utilizaron datos recientes que el INEGI reserva para el público y corresponden al censo nacional realizado en 2015 [10, 11].

3.3. Información geográfica

Los datos corresponden a la información topográfica de los municipios, en particular los polígonos que describen la aproximación territorial de estos se obtuvieron de la plataforma CARTO [12]. El formato de esta información es geoJSON y con esta pueden graficarse regiones en un mapa. Cada municipio y estado de la república tiene su propio geoJSON, y a través del módulo de manejo de datos JSON de Python pueden ser manipulados. La exposición de los mapas con los resultados que arroja el modelo se encuentra en desarrollo.

4. Implementación computacional del modelo

En esta sección se describen aspectos importantes de la implementación que se codificó en Python y adicionalmente se describen los cálculos de las variables claves que han sido modificadas a partir de la información demográfica de México. El resultado alcanzado es un software capaz de simular escenarios futuros de la pandemia en México a escala municipal. Se explican las variables manteniendo la notación del modelo de original, descritas en la Tabla 1.

4.1. Definición de variables del modelo matemático

Tres estratos de edad son necesarios: jóvenes (0 a 25 años), adultos (26 a 65 años) y personas mayores (más de 65 años). Los estados en los que cada agente en un grupo puede encontrarse se denominan como: susceptible (S), expuesto (E), asintomático

(A), infectado (I), hospitalizado en unidad de cuidados intensivos (H), recuperado (R) y muerto (D). La matriz de contactos C, así como el número promedio de contactos de un agente por estrato de edad k, se reutilizaron del modelo español [1]. Se justifica tal situación dadas las similitudes culturales entre ambas naciones. Sin embargo, el cálculo de estos valores pueden adaptarse de mejor forma e incluso añadir el estado *vacunados*, lo cual se resolverá en un trabajo futuro.

De acuerdo con los datos obtenidos en la Sección 3.2, se estimó el número promedio de habitantes en cada hogar $\sigma = 3{,}7$. La lógica para el cálculo de la matriz de movilidad de regiones R se resume en la siguiente premisa: un agente tendrá mayor probabilidad de migrar a un municipio cercano al que pertenece que de migrar a un municipio alejado.

Este planteamiento se ha inspirado en leyes físicas de fenómenos naturales, en particular en la ecuación de Coulomb, por lo que se asume que tal relación es inversamente proporcional al cuadrado de la distancia entre los municipios. Usando los datos recopilados en la Sección 3.3, se toma el polígono de aproximación territorial de cada municipio y se calcula el centroide geométrico del municipio para almacenarlo en una estructura llamada tabla de municipios. La matriz de movilidad se calcula como:

$$R_{i,j} = \frac{f(d(i,j))}{\sum_{k \neq i} f(d(i,k))}, i \neq j; \quad R_{i,i} = 0, \tag{14}$$

donde $f(x) = 1/x^2$, y $d(i,j)$ es la distancia euclidiana entre los centroides del i-ésimo municipio y el j-ésimo municipio de la tabla de municipios. Cabe señalar que $f(x)$ puede ser sustituida por otra función monótona decreciente adecuada y $d(i,j)$ por otra métrica que mejor describa la distancia entre municipios, por ejemplo la generada por la red de carreteras de México.

Finalmente, para el cálculo de la distribución inicial de la población $\rho(t_0)$, por municipio, por estrato de edad y por cada uno de los siete estados, se procesa la información epidemiológica descrita en la Sección 3.1. A cada registro de los datos se le asigna el estrato de edad al que pertenece y se clasifica en uno los estados posibles de la siguiente forma:

- Si el resultado de la prueba fue positivo, el paciente vive y han pasado más de 14 días desde la fecha de síntomas, se clasifica como paciente recuperado.
- Si el resultado de la prueba fue positivo, el paciente vive y han pasado menos de 14 días desde la fecha de síntomas, se clasifica como paciente portador.
- Si el resultado de la prueba fue positivo, el paciente vive y el paciente está hospitalizado en una Unidad de Cuidados Intensivos (UCI) o está intubado, se clasifica como paciente hospitalizado en UCI.
- Si el resultado de la prueba fue positivo, y el paciente se encontraba intubado o en una UCI, y tiene asociada una fecha de defunción, se clasifica como paciente difunto.

Posteriormente, con la información de la población de cada municipio por estratos de edad, se calcula el porcentaje de población en cada uno de los estados. El resto de la población supone se encuentra en estado susceptible.

El resto de las variables como el ritmo de mortandad, infectividad de pacientes asintomáticos e infectados, entre otros se reutilizan de la Tabla 1, por ser propiedades del patógeno y no de la región.

4.2. Implementación en Python

El modelo matemático ha sido implementado en un módulo de Python llamdo Mappand, apoyándose de la librería de cómputo Numpy. La librería se compone de la clase MMC (Micro-Markov Chain) que implementa las fórmulas expuestas en la Sección 2, las cuales son necesarias para evolucionar el esparcimiento del patógeno en el tiempo y la clase Modelo que se encarga de encapsular los métodos necesarios para leer los datos de entrada y almacenar los resultados en una base de datos. La lectura puede ser desde una base de datos o de archivos CSV. Las entradas de la clase MMC son:

- El numero de grupos poblacionales de la macropoblación, N.
- La matriz de contactos C, arreglo de numpy con forma $(3, 3)$.
- La matriz de movilidad R, arreglo de numpy con forma (N, N) .
- La matriz población por municipio y estrato de edad n, arreglo de numpy con forma $(N, 3)$.
- La densidad de población por municipio s, arreglo de numpy con froma $(N, 1)$.
- La distribución de población inicial $\rho(t_0)$, un diccionario de Python, cuyas llaves son los 8 distintos estados $\{S, E, A, I, H, D, R\}$ y cuyos valores son arreglos de numpy con forma $(N, 3)$.
- El diccionario de parámetros del modelo, que guarda la información de las variables descritas en la Tabla 1.

Una vez instanciado un objeto con estos datos, el método evolve se encarga de evolucionar en el tiempo el esparcimiento del patógeno en la macro-población. Las entradas al método son: el número de días a evolucionar en el tiempo Δt y el porcentaje de aislamiento social estimado k_0. Esta evolución se realiza de la siguiente forma:

1. Se actualizan los parámetros $\{n_i^{eff}, z^g, n_{j\to i}^{A,g}(t), n_{j\to i}^{I,g}(t), P_i^g(t), \Pi_i^g(t)\}$ al tiempo actual t, como indican las Ecuaciones 8 a 13.
2. Se calcula la evolución de la población $\rho(t + 1)$ según las fórmulas descritas de la Ecuación 1 a la Ecuación 7.
3. Se guarda la nueva población $\rho(t + 1)$ en un arreglo y se repite el procedimiento tomando $\rho(t + 1)$ como nueva población a evolucionar.

El algoritmo se repite Δt veces, y el valor de retorno es el arreglo con las poblaciones estimadas. Las fórmulas que implementa la clase MMC hacen uso de técnicas de vectorización de la librería numpy. MMC también es generalizable cambiando las entradas del programa por variables particulares de una meta población arbitraria.

Tabla 2. Simbología.

Notación	Significado
CE	Total de casos estimados
CA	Total de casos actuales
ΔC	Nuevos casos estimados
HE	Total de hospitalizaciones estimadas
HA	Total de hospitalizaciones actuales
ΔH	Nuevas hospitalizaciones estimadas
ME	Total de muertes estimadas
MA	Total de muertes actuales
ΔM	Nuevas muertes estimadas

5. Hacia la interpretación de datos en un SIG

La parte final del proyecto consiste en la interpretación de los resultados del modelo matemático. Una vez se ha calculado la distribución de la población hipotética por cada uno de los municipios, se calculan los valores:

$$\text{Casos}_i(t_0 + \Delta t) = \sum_{g=1}^{3} \left(\rho_i^{R,g}(t_0 + \Delta t) + \rho_i^{H,g}(t_0 + \Delta t) + \rho_i^{D,g}(t_0 + \Delta t) \right) n_i^g. \quad (15)$$

Esta suma encapsula los casos estimados por la vigilancia centinela pasados Δt días, suponiendo un porcentaje de aislamiento nacional k_0:

$$\text{Hosp}_i(t_0 + \Delta t) = \sum_{g=1}^{3} \rho_i^{H,g}(t_0 + \Delta t) n_i^g. \quad (16)$$

Esta ecuación reúne el número de pacientes hospitalizados en una UCI pasados Δt días, suponiendo un porcentaje de aislamiento nacional k_0:

$$\text{Muertes}_i(t_0 + \Delta t) = \sum_{g=1}^{3} \rho_i^{D,g}(t_0 + \Delta t) n_i^g. \quad (17)$$

Aquí se concentra el número de muertes acumuladas, pasados Δt días, asumiendo un porcentaje de aislamiento nacional k_0. En trabajo futuro mediante el módulo de manejo de datos JSON de Python, se añadirán estos resultados al archivo geoJSON de cada entidad y municipio como nuevas propiedades llamadas casos, hospitalizados y defunciones respectivamente.

Que posteriormente con la librería JavaScript de código abierto, Leaflet, se encargará de mostrar los datos geoJSON de los municipios y entidades aplicando una escala de colores pertinente en base a la propiedad de casos.

Tabla 3. Evolución: $\Delta t = 7, k_0 = 0{,}5$. El cuarto renglón indica que Baja California tiene 5023 casos confirmados y el modelo predice 5269 casos dentro de una semana si el 50 % de la población está en confinamiento.

Entidad	CE	CA	ΔC	HE	HA	ΔH	ME	MA	ΔM
República Mexicana	95104	87347	7757	814	312	502	10012	9767	245
Aguascalientes	912	815	97	7	2	5	36	34	2
Baja California	5269	5023	246	39	7	32	873	861	12
Baja California Sur	625	605	20	2	0	2	36	36	0
Campeche	595	563	32	9	12	-3	72	68	4
Coahuila	1174	1084	90	8	1	7	79	77	2
Colima	157	141	16	2	2	0	20	19	1
Chiapas	1982	1705	277	33	21	12	148	137	11
Chihuahua	1631	1569	62	13	16	-3	320	314	6
Ciudad de México	26181	24263	1918	223	42	181	2273	2203	70
Durango	416	365	51	6	3	3	38	37	1
Guanajuato	1712	1463	249	19	10	9	115	108	7
Guerrero	1966	1827	139	19	9	10	276	270	6
Hidalgo	1852	1670	182	16	5	11	286	284	2
Jalisco	1969	1660	309	31	17	14	148	136	12
Edo. De México	15698	14264	1434	107	31	76	1615	1588	27
Michoacán	2043	1869	174	14	1	13	163	160	3
Morelos	1523	1376	147	14	2	12	265	262	3
Nayarit	607	552	55	7	3	4	53	50	3
Nuevo León	1546	1372	174	14	6	8	93	88	5
Oaxaca	1421	1291	130	12	5	7	151	148	3
Puebla	3282	2914	368	39	23	16	357	345	12
Querétaro	973	875	98	10	5	5	94	92	2
Quintana Roo	1944	1843	101	20	19	1	354	346	8
San Luis Potosí	996	882	114	10	3	7	52	50	2
Sinaloa	3433	3261	172	26	15	11	493	484	9
Sonora	1992	1864	128	13	2	11	124	121	3
Tabasco	4336	4009	327	34	8	26	519	510	9
Tamaulipas	1688	1554	134	14	6	8	102	99	3
Tlaxcala	1084	989	95	10	12	-2	143	139	4
Veracruz	3872	3624	248	29	17	12	533	524	9
Yucatán	1920	1771	149	14	6	8	148	144	4
Zacatecas	305	284	21	0	1	-1	33	33	0

6. Discusión de resultados y conclusiones

A continuación, se muestran las predicciones de número de casos, número de hospitalizaciones en UCI (unidad de cuidados intensivos) y número de muertes en cada entidad mexicana obtenidas del modelo implementado, usando la base de datos abiertos de COVID-19 de la DGE correspondiente al día 30/05/2020 y distintos valores para las variables Δt y k_0. Estos resultados se encuentran en las Tablas 3, 4 y 5. En la Tabla 2 se describen los components.

Tabla 4. Evolución: $\Delta t = 7, k_0 = 0{,}75$.

Entidad	CE	CA	ΔC	HE	HA	ΔH	ME	MA	ΔM
República Mexicana	89817	87347	2470	726	312	414	10010	9767	243
Aguascalientes	846	815	31	6	2	4	36	34	2
Baja California	5113	5023	90	35	7	28	873	861	12
Baja California Sur	610	605	5	1	0	1	36	36	0
Campeche	573	563	10	8	12	-4	72	68	4
Coahuila	1117	1084	33	8	1	7	79	77	2
Colima	145	141	4	2	2	0	20	19	1
Chiapas	1793	1705	88	31	21	10	148	137	11
Chihuahua	1587	1569	18	12	16	-4	320	314	6
Ciudad de México	24961	24263	698	199	42	157	2272	2203	69
Durango	384	365	19	6	3	3	38	37	1
Guanajuato	1536	1463	73	17	10	7	115	108	7
Guerrero	1874	1827	47	18	9	9	276	270	6
Hidalgo	1723	1670	53	15	5	10	286	284	2
Jalisco	1754	1660	94	28	17	11	148	136	12
Edo. De México	14667	14264	403	87	31	56	1615	1588	27
Michoacán	1926	1869	57	13	1	12	163	160	3
Morelos	1412	1376	36	13	2	11	265	262	3
Nayarit	570	552	18	7	3	4	53	50	3
Nuevo León	1419	1372	47	11	6	5	93	88	5
Oaxaca	1329	1291	38	9	5	4	151	148	3
Puebla	3027	2914	113	35	23	12	357	345	12
Querétaro	903	875	28	10	5	5	94	92	2
Quintana Roo	1882	1843	39	19	19	0	354	346	8
San Luis Potosí	920	882	38	9	3	6	52	50	2
Sinaloa	3323	3261	62	24	15	9	493	484	9
Sonora	1902	1864	38	12	2	10	124	121	3
Tabasco	4124	4009	115	28	8	20	518	510	8
Tamaulipas	1599	1554	45	13	6	7	102	99	3
Tlaxcala	1010	989	21	10	12	-2	143	139	4
Veracruz	3689	3624	65	28	17	11	533	524	9
Yucatán	1813	1771	42	12	6	6	148	144	4
Zacatecas	286	284	2	0	1	-1	33	33	0

6.1. Interpretación de resultados

Según muestran las predicciones en las Tablas 3, 4 y 5, en el mejor de los casos el valor mínimo de nuevos casos pasados siete días es de 2470 casos a nivel nacional, que corresponde al escenario en el que el 75 % de la población permanece en cuarentena durante toda la semana. El hecho que el valor de Casos no disminuya en comparación a días anteriores, evidencia que la curva de casos no se está aplanando.

Además, se puede decir que partir de lo experimentado en este trabajo, la curva solo se puede aplanar si se siguen manteniendo las medidas preventivas de salud en cuanto a higiene, y disminuir el contacto con otras personas para evitar en la medida de lo posible trasmitir el virus. De otro modo, con una observación empírica, la curva no podrá ser aplanada a corto plazo.

Tabla 5. Evolución: $\Delta t = 30, k_0 = 0{,}5$. El doceavo renglón indica que Guanajuato tiene 10 pacientes en UCI y se predicen 48 nuevos pacientes en un mes si el 50 % de la población está en confinamiento.

Entidad	CE	CA	ΔC	HE	HA	ΔH	ME	MA	ΔM
República Mexicana	216707	87347	129360	1926	312	1614	11889	9767	2122
Aguascalientes	2281	815	1466	17	2	15	57	34	23
Baja California	7964	5023	2941	42	7	35	933	861	72
Baja California Sur	814	605	209	2	0	2	41	36	5
Campeche	923	563	360	4	12	-8	81	68	13
Coahuila	2420	1084	1336	17	1	16	100	77	23
Colima	438	141	297	5	2	3	23	19	4
Chiapas	6536	1705	4831	44	21	23	206	137	69
Chihuahua	2528	1569	959	10	16	-6	335	314	21
Ciudad de México	52577	24263	28314	569	42	527	2788	2203	585
Durango	1177	365	812	10	3	7	51	37	14
Guanajuato	5779	1463	4316	58	10	48	167	108	59
Guerrero	4173	1827	2346	27	9	18	312	270	42
Hidalgo	5070	1670	3400	39	5	34	330	284	46
Jalisco	7320	1660	5660	78	17	61	219	136	83
Edo. De México	41539	14264	27275	448	31	417	1955	1588	367
Michoacán	5110	1869	3241	31	1	30	194	160	34
Morelos	4106	1376	2730	46	2	44	305	262	43
Nayarit	1344	552	792	8	3	5	63	50	13
Nuevo León	4604	1372	3232	52	6	46	135	88	47
Oaxaca	4182	1291	2891	28	5	23	180	148	32
Puebla	9991	2914	7077	85	23	62	443	345	98
Querétaro	2527	875	1652	20	5	15	118	92	26
Quintana Roo	3070	1843	1227	13	19	-6	376	346	30
San Luis Potosí	2720	882	1838	20	3	17	74	50	24
Sinaloa	5378	3261	2117	29	15	14	538	484	54
Sonora	3276	1864	1412	16	2	14	147	121	26
Tabasco	8251	4009	4242	61	8	53	591	510	81
Tamaulipas	3483	1554	1929	28	6	22	137	99	38
Tlaxcala	3055	989	2066	20	12	8	169	139	30
Veracruz	9020	3624	5396	63	17	46	596	524	72
Yucatán	4188	1771	2417	32	6	26	186	144	42
Zacatecas	863	284	579	4	1	3	39	33	6

6.2. Trabajo futuro

Actualmente, se están generando archivos de documentación del módulo de Python con los resultados hasta hoy alcanzados. Se busca distinguir este proyecto de otros similares con la creación de un portal web que presente la posibilidad de realizar predicciones personalizadas de contexto particular, adaptando el software de los autores de este artículo un servicio en la nube.

Referencias

1. Arenas, A., Cota, W., Gómez-Gardeñes, J., Gómez, S., Granell, C., Matamalas, J. T., Soriano, D., Steinegger, B.: A mathematical model for the spatiotemporal epidemic spreading of COVID19. Cold Spring Harbor Laboratory (2020) doi: 10.1101/2020.03.21.20040022
2. Arranz, J., María, J.: COVID-19, SARS-CoV-2. GdT-semFYC en Enfermedades Infecciosas (2020) www.semfyc.es/wp-content/uploads/2020/03/COVID-19- semFYC.pdf
3. Bernabé-Loranca, M. B., Sarmiento-Barrios, E., Cerón-Garnica, C., Rubio-Quintero, R., Martínez-Guzmán, G.: Simulador estadístico de contagios para COVID-19 usando 13 variables del sistema de salud. Revista Pistas Educativas, vol. 42, no. 138 (2021)
4. Bernábe-Loranca, M. B., González-Velázquez, R., Granillo-Martínez, E., Ruiz-Vanoye, J. A., Canan, A. C.: Towards an approach of the contagion curve for COVID-19 in Mexico. In: Advances in Intelligent Systems and Computing. Springer International Publishing, pp. 553–566 (2021) doi: 10.1007/978-3-030-71187-0_51
5. INEGI: Datos. www.inegi.org.mx/datos/ (2020)
6. INEGI: Encuesta Intercensal 2015. www.inegi.org.mx/programas/intercensal/2015/default.html#Tabulados (2016)
7. Li, F.: Structure, function, and evolution of coronavirus spike proteins. Annual Review of Virology, vol. 3, no. 1, pp. 237–261 (2016) doi: 10.1146/annurev-virology-110615-042301
8. Secretaria de Salud: COVID-19 Tablero México (2020) coronavirus.gob.mx/datos/#DOView
9. Secretaria de Salud: Personal de Salud (2020) coronavirus.gob.mx/personal- de-salud/
10. Secretaría de Salud: Datos Abiertos - Dirección General de Epidemiología (2020) www.gob.mx/salud/documentos/datos-abiertos-152127
11. Serdán: Mapa con los polígonos de los municipios de México (2015) albertoserdan.carto.com/tables/mapa_municipios_de_mexico/public/map
12. Zhou, F., Yu, T., Du, R., Fan, G., Liu, Y., Liu, Z., Xiang, J., Wang, Y., Song, B., Gu, X., Guan, L., Wei, Y., Li, H., Wu, X., Xu, J., Tu, S., Zhang, Y., Chen, H., Cao, B.: Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: A retrospective cohort study. The Lancet, vol. 395, no. 10229, pp. 1054–1062 (2020) doi: 10.1016/s0140-6736(20)30566-3

ISSN 1870-4069

Detección de vida usando características de textura invariantes de Haralick

Oswaldo Vázquez, Ariel Alexis Placido Cabrera, Pedro Arguijo

Tecnológico Nacional de México, campus Misantla,
División de Ingeniería en Sistemas Computacionales, Veracruz,
México

{bigosvaap, arielalexisplacidocabrera}@gmail.com,
pedro_arguijo@excite.com

Resumen. La detección de vida dentro de los sistemas biométricos es una técnica cuyo objetivo es determinar si la biometría que se está capturando es una medición real de la persona viva autorizada que está presente en el momento de la captura. En este trabajo se realiza la detección de vida a través de las características de textura invariantes de Haralick en el dataset de "Spoof in the wild" sobre la región del rostro de los sujetos de prueba. Para la clasificación de las características se usaron cuatro clasificadores: arboles de decisión, random forests, support vector machine y gradient boosting tree teniendo los mejores resultados en random forests y gradient boosting tree con una exactitud general del 96% con un conjunto de entrenamiento del 20%.

Palabras clave: Detección de vida, textura, Haralick.

Liveness Detection Using Haralick Invariant Texture Features

Abstract. Life detection within biometric systems is a technique whose objective is to determine if the biometric being captured is a true measurement of the authorized living person present at the time of capture. In this work, life detection is carried out through Haralick invariant texture features in the Spoof in the wild dataset over the face region of the test subjects. Four classifiers were used for feature classification: decision trees, random forests, support vector machine and gradient boosting tree, obtaining the best results with random forests and gradient boosting tree. The overall accuracy of 96% with a training set of 20%.

Keywords: Live detection, texture, Haralick.

1. Introducción

Junto a la digitalización de diferentes servicios como la banca se ha creado una necesidad de medidas de seguridad contra los ataques de engaño. La biometría es una de estas medidas de seguridad que se ha adoptado para combatir dichos ataques. Algunas de las técnicas conocidas para identificación son el reconocimiento facial, el

reconocimiento de huellas dactilares, la verificación de la escritura, la geometría de la mano, escáner de retina e iris. Entre estas técnicas, la que se ha desarrollado rápidamente en los últimos años es la tecnología de reconocimiento facial por ser más directa, fácil de usar y conveniente en comparación con otros métodos. Por lo tanto, se ha aplicado a varios sistemas de seguridad.

Pero, en general, los algoritmos de reconocimiento facial no son capaces de diferenciar si la persona realmente se encuentra *in situ* y el rostro detectado corresponde a una persona que se encuentra en vivo en el lugar (live) o es alguna fotografía de dicha persona que intenta engañar al sistema haciendo creer que la persona se encuentra en el sitio (spoof), esto es un problema de seguridad importante. Es una forma fácil de falsificar los sistemas de reconocimiento facial por medio de imágenes del rostro. Con el fin de evitar este tipo de falsificación, un sistema seguro necesita detección de vida (*liveness detection*) [1].

La biometría es la tecnología que permite establecer la identidad de un individuo basándose en atributos físicos característicos de la persona. La importancia de la biometría en la sociedad moderna ha sido reforzada por la necesidad de sistemas de gestión de la identidad en gran escala cuya funcionalidad depende de la deducción exacta de la identidad de un individuo en el marco de varias aplicaciones.

Algunos ejemplos de estas aplicaciones son el intercambio de recursos informáticos en red, conceder acceso a las instalaciones sensibles, realizar transacciones financieras a distancia o abordar un vuelo comercial [2]. La principal tarea de un sistema de seguridad es la verificación de la identidad.

La razón principal de esto es evitar que los impostores accedan a recursos protegidos. Las técnicas generales para fines de seguridad son las contraseñas o los mecanismos de tarjetas de identificación, pero estas técnicas de identidad pueden perderse, obstaculizarse o ser robadas fácilmente, lo que perjudica a la seguridad prevista. Con la ayuda de las propiedades físicas y biológicas de los seres humanos, un sistema biométrico puede ofrecer más seguridad para un sistema de seguridad [1].

El problema de la falsificación debe ser resuelto antes de que los sistemas de reconocimiento facial puedan ser ampliamente aplicado en nuestra vida diaria. Para distintos tipos de métodos de detección de vida, en la interacción hombre-computadora es casi indispensable detectar el movimiento biológico de los usuarios.

Los movimientos más utilizados incluyen el parpadeo de los ojos [3, 4], la rotación de la cabeza [4, 5], y el movimiento de la boca [6]. Uno de los principales problemas de estos es que los usuarios deben ser altamente cooperativos y la duración de la detección de la vida es relativamente larga, lo que hacen que los usuarios se sientan incómodos al usar dichos sistemas.

En los sistemas biométricos, el objetivo de las pruebas de vida es determinar si la biometría que se está capturando es una medición real de la persona viva autorizada que está presente en el momento de la captura. Si bien los sistemas biométricos pueden tener un excelente rendimiento y mejorar la seguridad, estudios previos han demostrado que no es difícil engañar a los dispositivos biométricos mediante dedos falsos, imágenes o vídeo de alta resolución, lentes de contacto, etc.

Aunque los dispositivos biométricos utilizan información fisiológica para fines de identificación/verificación, estas mediciones rara vez indican la vida útil. La detección de la vida útil reduce el riesgo de falsificación al requerir una firma de vida útil además de la información biométrica correspondiente. Los métodos pueden incluir mediciones

médicas como la oximetría de pulso, un electrocardiograma o el olor. En unos pocos casos, la información sobre la vida es inherente a la propia biometría, es decir, esta medida no puede capturarse a menos que el usuario esté vivo, por ejemplo, el electrocardiograma es un método biométrico que solo puede usarse si la persona está viva, ya que este registra la actividad eléctrica del corazón.

Si bien el algoritmo de liveness dificulta la suplantación de identidad, es necesario considerarlo como un componente de un sistema biométrico que trae consigo características de rendimiento, así como factores como la facilidad de uso, la aceptación del usuario, la universalidad, la posibilidad de suplantación de identidad, la permanencia, etc.

Ningún sistema es perfecto en cuanto a su capacidad para prevenir los ataques de engaño. Sin embargo, los algoritmos de liveness pueden reducir esta vulnerabilidad para reducir al mínimo el riesgo de falsificación [7].

El uso del reconocimiento facial para la autenticación es cada vez más frecuente, especialmente en los dispositivos móviles. Pero entre el fácil acceso a las imágenes en los medios sociales y los avances en la resolución de imágenes digitales e impresas, los sistemas biométricos tienen lagunas de seguridad que los estafadores pueden aprovechar para falsificar con éxito un sistema de reconocimiento facial.

Para que la biometría facial pueda realmente ser adoptada por la mayoría como un mejor modo de autenticación, es esencial determinar si el rostro presentado es auténtico o si se trata de un intento de falsificar el sistema presentando una representación artificial del mismo.

Así pues, la detección automatizada de los ataques de presentación y, concretamente, la detección de la autenticidad se ha convertido en un componente necesario de cualquier sistema de autenticación que se base en la biometría facial para su verificación. El reconocimiento de vida facial ha surgido como una forma de detener el fraude y asegurar la integridad de la biometría facial como medio de autenticación. Mientras que el reconocimiento facial para la autenticación puede responder con precisión a la pregunta "¿Es esta la persona correcta?" no responde a la pregunta "¿Es esta una persona real?". Esta es la función de la detección de la vida.

La mayoría de las tecnologías actuales de detección de la vida facial son "activas", y requieren que los usuarios parpadeen, giren la cabeza o muevan el teléfono de un lado a otro. Esto da lugar a tres problemas: En primer lugar, los estafadores pueden presentar una foto recortada con agujeros en los ojos, utilizar una máscara o mostrar un vídeo para engañar al sistema. En segundo lugar, las técnicas de desafío-respuesta ponen a los atacantes en alerta de que están siendo revisados.

Y, por último, los métodos activos crean fricciones que ralentizan el proceso de autenticación, aumentan las tasas de abandono y disminuyen la experiencia general del usuario. En este trabajo proponemos determinar la detección de vida en imagenes del rostro considerando las matrices de coocurrencia de niveles de gris como una función de densidad de probabilidad discreta, tal como lo propuso Löfstedt et al. [8]. Evaluamos el desempeño de las características extraídas de la región de interés con árboles de decisión, bosques aleatorios (random forest, RF), máquinas de soporte vectorial (support vector machines, SVM) y potenciación del gradiente (gradient boosting tree, GBT).

El artículo se distribuye de la siguiente manera: en la Sección 2 describimos la metodología empleada, en la sección 3 se presentan los resultados y en la sección 4 se incluyen las conclusiones de esta investigación.

2. Trabajos previos

La detección en vivo es crucial a la hora de sistemas biométricos como la detección de rostro, permitiendo garantizar que la persona reconocida se encuentra realmente en el sitio, para la detección en vivo se han propuesto diferentes enfoques como el uso de características de textura, movimiento como parpadeo, boca u ojos, forma, reflejos, color y análisis en el dominio de la frecuencia. Los rasgos de textura se extraen de las imágenes de los rostros bajo el supuesto de que los rostros impresos producen ciertos patrones de textura que no existen en los reales.

La textura es probablemente la evidencia más fuerte de la falsificación, ya que más del 69% de las obras utilizan la textura sola o la combinan con otros descriptores en sus contramedidas [9]. El análisis de microtextura fue implementado por Jukka y otros [10]. La idea clave es enfatizar las diferencias de micro textura en el espacio de los rasgos. Los autores adoptan los patrones binarios locales (LBP), que es un poderoso operador de texturas, para describir las micro texturas y su información espacial. Los vectores en el espacio del rasgo se dan entonces como entrada a un clasificador SVM que determina si los patrones de microtextura caracterizan una imagen falsa o una imagen de persona viva.

Los descriptores de movimiento son los segundos en importancia para la detección de la suplantación de la cara, y hay dos formas diferentes de considerar el movimiento para este propósito. Una forma es detectar y describir las variaciones interfaciales, como el parpadeo de los ojos, las expresiones faciales y la rotación de la cabeza [9]. La técnica basada en el análisis del movimiento de los ojos fue introducida por Hyung-Keun Jee y otros para el sistema de reconocimiento facial incorporado [11]. Los autores propusieron un método para detectar los ojos en imágenes de entrada secuenciales y luego se calcula la variación de cada región ocular y se determina si la cara de entrada es real o no.

La suposición básica es que, debido al parpadeo y a los movimientos incontrolados de las pupilas en los ojos humanos, debería haber grandes variaciones de forma [12]. A su vez también se encuentran trabajos donde se combinan dos o más métodos como en [13], donde utilizaban un método para detectar ataques de "spoofing" facial combinando la detección de texturas de HSV (Tono, Saturación, Valor) y la detección del movimiento del parpadeo de los ojos.

El algoritmo antispoofing consiste en tres módulos principales: el detector de parpadeo, el detector de HSV y el módulo de puntuación combinada. La visión general del flujo de trabajo es que, en primer lugar, la imagen del rostro capturada por la cámara se comprueba de antemano si el rostro que se ha obtenido es el verdadero o un falso rostro resultante de un ataque de falsificación de rostro. La imagen del rostro es una captura cuadro a cuadro donde cada cuadro es el resultado de una captura de cámara de 30 fps (cuadros por segundo).

Los resultados de esta foto se analizan a través del módulo detector de parpadeo para producir un valor de puntuación de parpadeo.

(a) (b)

Fig 1. Ejemplos del dataset. a) Vivo b) Engaño. Tomado de [15].

Fig 2. RGB a Escala de grises.

3. Metodología

3.1. Dataset

El dataset considerado en este trabajo es el reportado por Liu, et. al. [14], el cual contiene 165 videos de personas reales y videos con imágenes falsas de las mismas considerando diversas variaciones. Para cada sujeto, se tienen 8 videos en vivo y hasta 20 falsos, en total 4,478 videos. Todos los videos están en 30 cuadros por segundo, aproximadamente 15 segundos de duración y resolución HD de 1080P.

Los videos en vivo se recogen en cuatro sesiones con variaciones de distancia, pose, iluminación y expresión. Los videos falsos se recopilan con varios ataques, como papel impreso y reproducción [15]. Todos los archivos de video se identifican como SubjectID_SensorID_TypeID_MediumID_SessionID .mov (o *.mp4). SubjectID varía de 001 a 165. SensorID representa el dispositivo de captura. TypeID representa el tipo de falsificación del video. MediumID y SessionID registran detalles adicionales del video, en la Fig. 1 se puede ver un ejemplo del dataset.

También se proporciona un archivo del cuadro delimitador de caras con el mismo nombre del video correspondiente (es decir, SubjectID_SensorID_Ty peID_MediumID_ SessionID.face). En cada archivo *.face, contiene una matriz de 4 por n, donde n es la longitud del marco. Cada fila registra las ubicaciones (x, y) de la

esquina superior izquierda y la esquina inferior derecha del cuadro delimitador actual, como [785 425 1070 710]. [0,0,0,0] significa que no se ha detectado ningún rostro [10].

3.2. Regiones de interés

Nuestra región de interés corresponde al área de rostro, la cual está descrita por las ubicaciones (x, y) descritas en la sección anterior. Para la extracción del rostro se itero en cada fotograma del video y posteriormente se extrajo el rostro usando las coordenadas correspondientes obtenidas del archivo *.face con el que cuenta cada video.

3.3. Matriz de coocurrencia de niveles de gris

Una vez obtenida nuestra región de interés es necesario calcular la matriz de coocurrencia de niveles de gris (GLCM, por sus siglas en inglés) para la cual se debe transformar la imagen RGB a una imagen de intensidad o de niveles de gris, para lo cual se utilizó la siguiente expresión

$$gray = 0.229R + 0.587G + 0.114B,$$

donde R, G y B corresponden a los componentes del rojo, verde y azul, respectivamente de las imágenes a color. En la Fig. 2 se muestra el efecto de transformar una imagen de RGB a escala de grises. La matriz GLCM se crea calculando la frecuencia con la que un píxel con el valor de intensidad (nivel de gris) i aparece en una relación espacial específica con un píxel con el valor j. Cada elemento (i, j) en la matriz GLCM resultante es simplemente la suma del número de veces que el píxel con valor i ocurrió en la relación espacial especificada con un píxel con valor j en la imagen de entrada.

Para calcular la matriz, es necesario definir una distancia y una dirección, además de los pares de píxeles separados esa distancia, tal como se muestra en la Fig. 3. A partir de esta matriz se pueden obtener distintas características de segundo orden: contraste (variación local de intensidad en una imagen), correlación (medida de la dependencia lineal de los niveles de gris), autocorrelación (evaluación tanto de la regularidad como de la tosquedad de la textura), cluster prominence y cluster shade (ofrecen información sobre el grado de simetría), energía (mide la repetición del píxel y expresa la regularidad de la textura), entropía (representa la irregularidad en la distribución de los valores de intensidad y es inversa a la energía), diferencia de entropía, diferencia de la varianza, solo por mencionar algunas.

3.4. Características invariantes de Haralick

Debido a la carga computacional para obtener las matrices de co-ocurrencia, los niveles de gris se cuantifican. En consecuencia, las características resultantes dependen en gran medida de la cuantificación. Si la cuantificación es idéntica, se puede garantizar la reproducibilidad de las características.

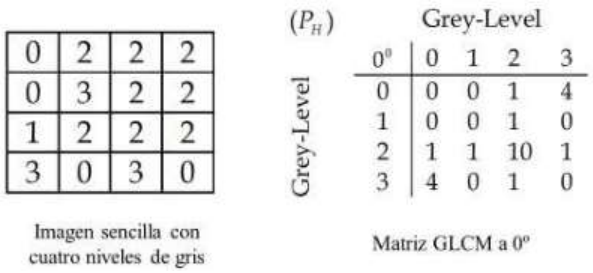


Fig. 3. Ejemplo para obtener la matriz GLCM con distancia par y dirección 0º.

Al redefinir la matriz de co-ocurrencia como una función de densidad de probabilidad discretizada [8], definió GLCM que son asintóticamente invariantes al número de niveles de gris. Con este enfoque, demostraron que las características invariantes tienen una mayor precisión, y tienen rendimientos similares incluso si las imágenes de entrenamiento y de prueba tenían una cuantización bastante diferente.

Una descripción detallada de las 21 características invariantes de textura de Haralick y su cálculo se puede encontrar en [8]. Entre estas están: autocorrelación, contraste, correlación, energía, entropía, homogeneidad, entre otras. Este conjunto de características ha mostrado un mejor desempeño en los clasificadores que las características originales.

3.5. Clasificadores

Para la clasificación se usó cuatro tipos de clasificadores Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM) y Gradient Boosted Tree (GDT). Del dataset [15] se extrajeron un total de 100817 imágenes divididas en dos clases: live and spoof. Todos los clasificadores indicados se probaron con diferentes proporciones en los datos para entrenamiento y prueba: 20/80, 30/70, 40/60, 50/50, 60/40, 70/30 y 80/20.

Para el algoritmo de DT se usó el algoritmo de CART, RF usa un conjunto de tres decisiones para predecir la clase. Cada árbol de decisión ha sido entrenado en un subconjunto aleatorio del conjunto de entrenamiento y solo usa un subconjunto aleatorio de las características, el número de árboles es de 100.

SVM separa los datos de entrenamiento en dos clases utilizando un hiperplano de margen máximo, el kernel utilizado es Radial Basis Function, en cuanto al clasificador GBT está formado por un conjunto de árboles de decisión individuales, entrenados de forma secuencial, de forma que cada nuevo árbol trata de mejorar los errores de los árboles anteriores. La predicción de una nueva observación se obtiene agregando las predicciones de todos los árboles individuales que forman el modelo.

4. Resultados

En la Tabla 1, se muestras los resultados de exactitud obtenidos con los diversos clasificadores considerando las razones de entrenamiento y pruebas mencionados anteriormente. En la tabla podemos ver los resultados con el conjunto de datos de

Tabla 1. Resultados obtenidos por diferentes conjuntos de entrenamiento y prueba, y distintos clasificadores.

Train / Test	DT		RF		SVM		GBT	
	Train	Test	Train	Test	Train	Test	Train	Test
20 / 80	0.925	0.931	0.966	0.970	0.954	0.958	0.966	0.962
30 / 70	0.945	0.943	0.975	0.974	0.951	0.957	0.966	0.962
40 / 60	0.950	0.949	0.980	0.978	0.962	0.960	0.977	0.972
50 / 50	0.952	0.952	0.982	0.981	0.962	0.961	0.974	0.974
60 / 40	0.957	0.957	0.984	0.984	0.960	0.961	0.977	0.975
70 / 30	0.966	0.961	0.987	0.987	0.962	0.961	0.975	0.976
80 / 20	0.961	0.961	0.988	0.987	0.965	0.963	0.977	0.974

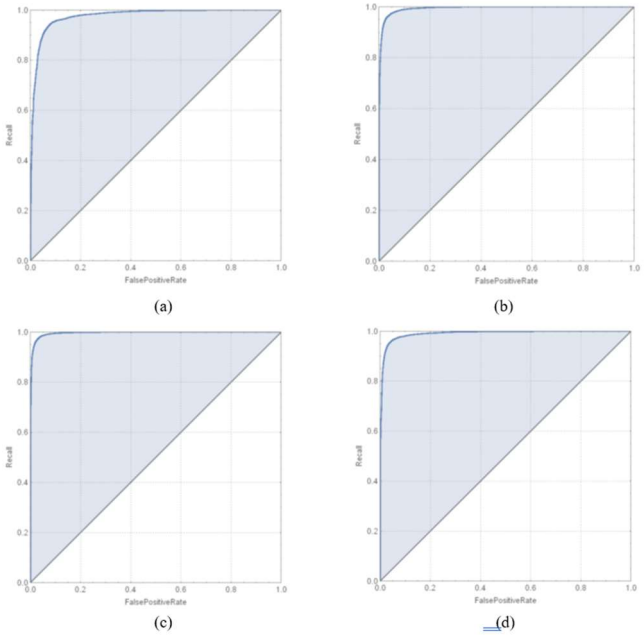


Fig 4. Curvas ROC. a) Decision Tree, b) Random Forest, c) Support Vector Machine, d) Gradient Boosting Tree.

entrenamiento y prueba de los algoritmos de Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM) y Gradient Boosting Tree (GBT), vemos que los resultados de entrenamiento son muy similares a los resultados en la parte de prueba, por lo tanto, el algoritmo no se sobre ajustan al conjunto de entrenamiento y generalizan de forma aceptable el conjunto de datos de prueba. Un ejemplo de las curvas ROC obtenidas con los clasificadores mencionados se muestra en la Fig. 4, la proporción considerada fue la de 20/80.

5. Conclusiones y trabajo futuro

En este trabajo se buscó la detección de vida en el rostro a través del análisis de textura, existen diferentes métodos que se han usado para la detección de vida como el parpadeo de los ojos, hacer que la persona haga ciertos movimientos cada vez que se

desea comprobar esto, lea algún texto para confirmar que es ella realmente el problema con estos enfoques es que solo funcionan teniendo al sujeto frente a la cámara por cierto periodo de tiempo y no son implementables en situaciones donde se necesite comprobar que una fotografía fue tomada realmente a la persona, además de que el método de análisis de textura es implementable en sistemas de video dado que un video se puede ver como una serie de fotografías a alta velocidad, el objetivo principal se cumplió al poderse comprobar que con este enfoque se pueden obtener buenos resultados. Como trabajo futuro se busca la optimización del algoritmo y creación de una librería o API para la fácil implementación en diversos sistemas.

Referencias

1. Chakraborty, S., Das, D.: An overview of face liveness detection. arXiv preprint arXiv:1405.2227 (2014) doi: 10.48550/arXiv.1405.2227
2. Jain, A. K., Flynn, P., Ross, A. A.: Handbook of biometrics. Springer (2008)
3. Pan, G., Sun, L., Wu, Z., Lao, S.: Eyeblink-based Anti-spoofing in face recognition from a genericwebcamera. In: Proceedings of 11th IEEE International Conference on Computer Vision, pp. 1–7 (2007) doi: 10.1109/ICCV.2007.4409068
4. Kollreider, K., Fronthaler, H., Bigun, J.: Verifying liveness by multiple experts in face biometrics. IEEE Computer Vision and Pattern Recognition Workshops, Anchorage, pp. 1–6 (2008) doi: 10.1109/CVPRW.2008.4563115
5. Kollreider, K., Fronthaler H., Bigun, J.: Evaluating liveness by face images and the structure tensor. Fourth IEEE Workshop on Automatic Identification Advanced Technologies, pp. 75–80 (2005) doi: 10.1109/AUTOID.2005.20
6. Chetty, G., Wagner, M.: Liveness verification in audio-video speaker authentication. In: Proceedings of 10th Australian Int. Conference on Speech Science and Technology (2004)
7. Li S. Z., Jain A. K.: Encyclopedia of biometrics. Springer, Boston (2009)
8. Löfstedt, T., Brynolfsson, P., Asklund, T., Nyholm, T., Garpebring, A.: Gray-level invariant Haralick texture features. PloS one, vol. 14, no. 2, pp. e0212110 (2019) doi: 0.1371/journal.pone.0212110
9. Souza, L., Oliveira, L., Pamplona, M., Papa, J.: How far did we get in face spoofing detection? Engineering Applications of Artificial Intelligence, vol. 72, pp. 368–381 (2018) doi: 10.1016/j.engappai.2018.04.013
10. Maatta, J., Hadid, A., Pietikainen, M.: Face spoofing detection from single images using microtexture analysis. In: Proceedings of. International Joint Conference on Biometrics, pp. 1–7 (2011) doi: 10.1109/IJCB.2011.6117510
11. Jee, H. K., Jung, S. U., Yoo, J. H.: Liveness detection for embedded face recognition system. International Journal of Biological and Medical Sciences, vol. 1, no. 4, pp. 235–238 (2006)
12. Hadiprakoso, R. B.: Face Anti-Spoofing Method with Blinking Eye and HSV Texture Analysis. In: Proceedings of IOP Conference Series: Materials Science and Engineering, IOP Publishing, vol. 1007, no. 1, pp. 012034, (2020)
13. Liu, Y., Jourabloo, A., Liu, X. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 389–398 (2018)

ISSN 1870-4069

Distancia de Levenshtein para anonimización de notas médicas y detección de comorbilidades

Alejandro Martínez-Torres[1], Helena Gómez-Adorno[2]

[1] Universidad Nacional Autónoma de México,
Facultad de Ciencias,
México

[2] Universidad Nacional Autónoma de México,
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
México

alejandromartinezt@ciencias.unam.mx,
helena.gomez@iimas.unam.mx

Resumen. En este trabajo se anonimizaron notas médicas y posteriormente se realizó un proceso de extracción de información relacionado a las comorbilidades que presentan los pacientes. Para la evaluación de los métodos desarrollados se utilizó un corpus de 98 notas médicas de pacientes diagnosticados con el nuevo SARS-CoV-2, la información fue proporcionada por la Secretaría de la Salud de la Ciudad de México. Logramos anonimizar en su totalidad la notas médicas y el método que desarrollamos extrajo el 96 % de las comorbilidades presentes en las notas médicas. Ambos resultado se lograron haciendo uso de la distancia de Levenshtein y la metodología creada puede ser usada para distintas tareas de la misma índole.

Palabras clave: Notas médicas, extracción de información, distancia de Levenshtein, anonimización de documentos, detección de comorbilidades.

Levenshtein Distance for Anonymization of Medical Notes and Detection of Comorbidities

Abstract. In this work, medical notes were anonymized and subsequently an information extraction process related to the comorbidities presented by the patients was carried out. For the evaluation of the developed methods, a corpus of 98 medical notes of patients diagnosed with the new SARS-CoV-2 was used, the information was provided by the Ministry of Health of Mexico City. We managed to completely anonymize the medical notes and the method we developed extracted 96% of the comorbidities present in the medical notes. Both results were achieved using the Levenshtein distance and the created methodology can be used for different tasks of the same nature.

Keywords: Medical notes, information extraction, Levenshtein distance, anonymization of documents, detection of comorbidities.

1. Introducción

Para la realización de este trabajo se elaboraron métodos para anonimizar los datos del paciente y extraer un listado de las posibles comorbilidades mencionadas en una nota médica.

Las notas médicas, en las que se basó el presente trabajo, fueron proporcionadas por la Secretaría de Salud de la Ciudad de México, quien solicitó la completa anonimidad de los pacientes a los que se refiere en las notas médicas. Por ello, anonimizar los datos del paciente es esencial para el uso, experimentación y etiquetado manual en notas médicas.

También es importante remarcar que al momento de anonimizar la nota médica se desea conservar todo dato que no sea parte del nombre, ya que puede poseer información relevante para futuros procesamientos.

Los expedientes médicos se almacenan en formato textual, por lo cual datos de alta relevancia no siempre se encuentran disponibles para el cómputo estadístico. En el ámbito médico, las comorbilidades son altamente relevantes, ya que predisponen a más enfermedades y son un factor de riesgo importante.

Las comorbilidades se refieren a enfermedades o trastornos secundarios que afligen al paciente, además de la enfermedad primaria por la cual se realizó la consulta médica [4]. Para el caso de la detección de comorbilidades se usó como referencia el listado del CIE-10 [3] de enfermedades y problemas relacionados con la salud, tomándolos como las posibles comorbilidades.

El presente trabajo desarrolla un método para la detección de comorbilidades sin necesidad de datos etiquetados. Se hace uso de uno de los métodos más sencillos, pero efectivo, para el etiquetado de términos específicos, la búsqueda en un texto de las ocurrencias de los términos especificados con anterioridad en una lista.

Aunque existen una gran variedad de métodos para la búsqueda de palabras o frases en un texto [8], la mayoría de estos no permiten errores en la escritura de la palabra.

La distancia de Levenshtein [2] es un método que sirve para medir la distancia entre dos palabras respecto a su escritura y es comúnmente usado para corregir errores de ortografía [1]. Sin embargo, comparar la distancia de Levenshtein de un gran número de palabras es relativamente costoso computacionalmente hablando.

Este trabajo está estructurado de la siguiente manera. En la Sección 2, se describen trabajos relacionados. En la Sección 3, se presenta una breve descripción del corpus de notas médicas utilizado para evaluar los métodos desarrollados. En la Sección 4, se presentan los métodos desarrollados para la anonimización de las notas médicas y la detección de comorbilidades. En la Sección 5 se presentan los resultados obtenidos. Y finalmente, en la Sección 6 presentamos las conclusiones y trabajo futuro.

2. Trabajo relacionado

En el reconocimiento de entidades nombradas, específicamente, la detección y etiquetado de nombres en el idioma español, el proyecto de Stanza, realizado por la universidad de Stanford [5], es uno de los que se encuentra más a la vanguardia. Sin embargo, muchos nombres comunes en México no son identificados como entidades.

Esto en su mayoría se debe a errores en la escritura del nombre, nombres poco comunes en el entrono que se entrenó el modelo y palabras que cuentan con más de un significado. Lamentablemente, para este trabajo no se cuenta con la cantidad necesaria de datos para reentrenar un modelo de este tipo. Por otro lado, dado que se cuentan con los datos del paciente de cada nota médica, pudimos usar métodos más tradicionales para la detección del nombre del paciente en la nota médica y posteriormente su anonimización.

En el área de la detección de comorbilidades, los trabajos previos cuentan normalmente con una gran cantidad de datos etiquetados, lo cual permite una variedad más amplia a procesos a realizar como lo son el uso de los vectores de palabras (más conocidos como *embeddings*, en inglés) [10].

3. Corpus

Para el presente trabajo, SEDESA nos proporcionó un corpus de 98 expedientes médicos electrónicos de pacientes diagnosticados con el nuevo coronavirus SARS-CoV-2 (COVID). Dichos datos fueron proporcionados en un formato XML el cual venía organizado por secciones de las cuales se describen a continuación:

- Nombre y apellidos del paciente.
- Edad del paciente.
- Sexo del paciente.
- Estado y alcaldía.
- Fecha de ingreso.
- Fecha alta.
- Fecha hora registro nota.
- Nota médica (XML).
- Signos vitales: contiene el resumen de los signos vitales del paciente.
- Objetivo: contiene la descripción del estado actual del paciente y motivo de la consulta o revisión hospitalaria.
- Análisis: contiene la descripción del hallazgo del médico.
- Diagnóstico: describe el diagnóstico de la enfermedad del paciente.
- Plan de manejo: describe el tratamiento recetado al paciente, tanto de medicamentos como dieta, estudios necesarios, etc.

Es importante destacar que el objeto de estudio de este trabajo es el análisis de la nota médica, por lo tanto, cada sección del XML de la nota médica fue extraído para formar un solo documento por paciente. En la Figura 1 se muestra un ejemplo del una nota médica ya anonimizado por cuestiones de confidencialidad.

Inicialmente el texto de las notas médicas no contenía ningún tipo de etiquetado, la única etiqueta que se tenía son las relacionadas con el paciente y el diagnóstico. Después de animizar las notas médicas, con la colaboración de tres expertos de SEDESA, se etiquetó de manera manual cada nota médica del corpus de pacientes COVID mediante la interfaz de una plataforma web de anotación de datos *Dataturks*[3]. A continuación se enuncian las características etiquetadas:

[3] https://docs.dataturks.com/

Nota Médica <paciente> [redacted] 06/09/1962 515351 <paciente> Mujer <doctor> HOSPITAL ABC <doctor> Signos Vitales 21/07/2020 06:52: Temperatura: 36.4 / Frecuencia cardiaca - ADL: 82.0 / Frecuencia respiratoria - ADL: 20.0 / SaO2: 93.0 / Otras constantes de hoy:Tensión Arterial Sistólica - ADL: 125.0 / Tensión Arterial Diastólica - ADL: 80.0 / Tensión Arterial Media - ADL: 95.0 / Síntomas Se trata de <paciente> de 57 años de edad, con Obesidad grado I, sin otros antecedentes de importancia para la enfermedad actual. Niega dolor torácico, Negó sintomatología urinaria digestiva. La paciente niega la presencia de disnea. Objetivo Mujer de edad aparente igual la cronológica, orientada en tiempo, persona, lugar circunstancia, alerta. Coloración normal de mucosas. Estado de hidratación adecuado, con aporte de oxígeno suplementario por puntas nasales 0.5lpm . Saturando 96%.

Fig. 1. Ejemplo de nota médica preprocesada.

1. Síntomas, se identifican las palabras que contienen referencia a síntomas presentados por el paciente.
2. Comorbilidades, se identifican las palabras que hacen referencia a enfermedades previas del paciente.
3. Medicamentos, se identifican los medicamentos recetados al paciente.
4. Medicamentos previos, se identifican los medicamentos de base que el paciente está tomando actualmente.
5. Dosis, se identifica la dosis de los medicamentos (recetados y previos).
6. Medidas (alternativas), identifica tratamientos alternativos como ozonoterapia, dieta especial, etc.
7. Signos vitales, se identifican los signos vitales como frecuencia respiratoria (FR), frecuencia cardiaca (FC), saturación de oxígeno (SATO2), tensión arterial sistólica (TS) y diastólica (TD) y temperatura.
8. Datos antropométricos, se marcan el peso y la altura del paciente.

La Figura 2 muestra el ejemplo de una nota médica etiquetada con algunas de las características descritas previamente. Es importante destacar que no todos los expediente contaban con todas las características.

Para definir las posibles comorbilidades se usó el listado del CIE-10 [3] de enfermedades y problemas relacionados a la salud. El listado se extrajo de la página oficial del CIE-10 usando técnicas de *webcrawling* [9], con el objetivo de extraer los más de 10 mil casos específicos que la CIE-10 ha especificado y cubrir toda posible comorbilidad.

En las siguientes secciones describimos los métodos desarrollados en este trabajo para la anonimización de los datos paciente y la detección de comorbilidades de la nota médica.

4. Anonimización del paciente

Como se mencionó con antelación se quiere anonimizar los datos del paciente conservando la mayor cantidad de información del texto original. El siguiente ejemplo muestra un registro del expediente médico donde se tiene los datos del paciente de forma tabular y en la nota médica se vuelve a especificar el nombre del paciente:

Fig. 2. Ejemplo de una nota médica etiquetada con características que se muestran en la figura.

- Nombre: José María.
- Apellido Paterno: Sánchez.
- Apellido Materno: de los Ángeles.
- Nota Médica: Se trata de José María Sanches de los Ángeles de 57 años de edad, con Obesidad grado I, sin otros antecedentes de importancia para la enfermedad actual. El señor José María Sanches de los Ángeles niega dolor torácico.

El objetivo de este proceso es eliminar las menciones del nombre del paciente para luego proceder a analizar la nota médica sin riesgo de violar la confidencialidad del paciente. El proceso de anonimización se describe de la siguiente manera:

1. Se realiza un preprocesamiento retirando signos de puntuación, acentos, números y mayúsculas. Se desea eliminar la mención del nombre del paciente del texto, pero conservar el resto del texto, lo cual incluye, entre otros elementos, signos de puntuación y acentos. Para no perder la información mencionada se creó una copia del texto. El resultado del preprocesamiento en el ejemplo dado es el siguiente:

   se trata de jose maria sanches de los angeles de años de edad con obesidad grado i sin otros antecedentes de importancia para la enfermedad actual el señor sanchez niega dolor toracico.

2. Una vez obtenido el texto preprocesado se utilizaron expresiones regulares [7] para reemplazar los elementos que conforman el nombre completo del paciente (nombres, apellido paterno y apellido materno) por un símbolo predefinido (#). Cada uno de los elementos del nombre puede estar constituido por una o más palabras, por ello se remplazó cada elemento del nombre por la misma cantidad de palabras marca. El resultado de este proceso en el ejemplo dado es el siguiente:

se trata de # # # # # sanches de años de edad con obesidad grado i sin otros antecedentes de importancia para la enfermedad actual el señor # niega dolor toracico.

Es importante notar aquí que no todos los elementos del nombre del paciente fueron reemplazados por el símbolo # ya que existen errores ortográficos. El error ortográfico en el apellido del paciente (*sanches* en lugar de *sanchez*) impide hacer un reemplazo directo ya que la palabra *sanches* no está en vocabulario de búsqueda.

3. Una vez remplazadas en el texto las ocurrencias bien escritas de los elementos del nombre, se buscaron y remplazaron las ocurrencias mal escritas. Para ello se separaron las palabras que conforman el nombre, se descartaron las palabras que tengan una longitud menor a tres caracteres y se calculó la distancia de Levenshtein con las palabras en el texto que tenían una longitud similar. Si la distancia entre dos palabras (ponderada con la longitud de la palabra) es menor a cierto rango, definido previamente, la palabra es remplazada con la palabra marca. El resultado de este proceso en el ejemplo dado es el siguiente:

   se trata de # # # # # # de años de edad con obesidad grado i sin otros antecedentes de importancia para la enfermedad actual el señor # niega dolor toracico.

4. Finalmente, al tener una correspondencia uno a uno entre las palabras del texto procesado y el texto original, se pudo recuperar todos los elementos eliminados en el preprocesamiento. Y a su vez se remplazó la ocurrencia de una o más palabras marca consecutivas por la etiqueta paciente. El resultado de este proceso en el ejemplo dado es el siguiente:

   Se trata de <paciente> de 57 años de edad, con Obesidad grado I, sin otros antecedentes de importancia para la enfermedad actual. El señor <paciente> niega dolor torácico.

5. Detección de comorbilidades

5.1. Preprocesamiento

Dadas las notas médicas anonimizadas se procedió a hacer un proceso de limpieza del texto en donde se removió elementos no relevantes para la tarea a realizar, tales como acentos y puntuación. El mismo proceso se realizó con el listado del CIE-10 de enfermedades y problemas relacionados a la salud.

Dado que en su mayoría cada elemento del listado del CIE-10 se conforma por más de una palabra se extrajeron las palabras más relevantes para su búsqueda en las notas médicas.

Para la extracción de palabras del listado del CIE-10 primeramente se retiraron todas las palabras vacías. Posteriormente, tomando cada elemento del listado como un documento diferente, se calculó la frecuencia inversa de documento de cada palabra. Obteniendo así una forma de valorar que tan buen discriminante es cada palabra [6].

Esta información fue usada para extraer las 3 palabras más relevantes de cada elemento del listado del CIE-10 y crear un conjunto de palabras (5,968) a buscar en las notas médicas. Para cada palabra a buscar se guardó con que elementos del listado está relacionada y la relevancia tiene en cada elemento.

5.2. Búsqueda de múltiples palabras con posibles errores en un texto

Dado un conjunto de palabras y un texto, se realizó un proceso para obtener en orden una lista de todas las palabras del conjunto que aparecen en el texto, considerando posibles errores en la escritura de la palabras en el texto. Para ello se iteró sobre cada una de las palabras que conforman el texto y se realizó lo que se presenta a continuación:

1. Filtrado de candidatos

La ocurrencia que dos palabras tengan el mismo número de letras, a lo cual se denomina como anagrama, es poco frecuente. Tal es el caso de nuestro conjunto de 5,968 palabras a buscar, de las cuales solo hubo 63 casos de anagramas.

Por ello, lo primero que se realizó fue un proceso para que dada una palabra objetivo y un conjunto de palabras, determinar un subconjunto de palabras que tenga aproximadamente las mismas letras, a lo cual denominamos como semi-anagrama.

Para realizar de forma eficiente se realizó un preprocesamiento en donde se crean subconjuntos de palabras que tienen el mismo número de veces la misma letra, a los cuales denominamos como filtros. Ejemplo creación de filtros:

Conjunto de palabras: {“ab”, “abc”, “aca”},
- (a, 0) : {},
- (a, 1) : {“ab”, “abc”},
- (a, 2) : {“aca”},
- (b, 0) : {“aca”},
- (b, 1) : {“ab”, “abc”},
- (c, 0) : {“ab”},
- (c, 1) : {“abc”, “aca”}.

Dada la información anteriormente procesada, para encontrar un subconjunto de las palabras que sean anagramas de una palabra objetivo, solo se necesita calcular el número que tiene la palabra objetivo de cada letra y sacar la intersección de los filtros correspondientes.

Sin embargo, seleccionar palabras que sean anagramas de otra solo permite considerar errores en el orden de la escritura de la palabra, no en las letras que la conforman. Para ampliar el número de candidatos es importante tener un umbral de cuantas letras se pueden errar.

Dado x un porcentaje que se puede errar en una palabra, previamente definido, y una palabra p de longitud l, se calcula u que corresponde a un número entero del número de letras en p que se pueden errar; donde errar es cambiar una letra por otra, agregar una letra o eliminar una letra. Una vez calculado u y teniendo los filtros correspondientes a p, se puede calcular un subconjunto de palabras que sean semi-anagramas.

Lo primero a denotar es que toda palabra que sea semi-anagrama de p, puede no estar en a lo más $2u$ filtros. Esto se debe a que un cambio del tipo agregación o eliminación de una letra retira la palabra de exactamente un filtro (en la letra agregada o eliminada), sin embargo un cambio tipo remplazo elimina la palabra en dos filtros (en la letra agregada y en la eliminada).

Tabla 1. Ejemplo de comparación de dos palabras con $u = 1$.

ejemplo	eje	jem	emp	mpl	plo
eejemplo	eej	eje	jem	emp	mpl
Elementos en común	{e,e,j}	{e,j}	{e,m}	{m,p}	{p,l}
Puntaje	3	2	2	2	2

Dadas estas observaciones, podemos eliminar, de un conjunto de palabras candidato, las palabras que no son semi-anagramas. Esto se realiza con un sistema de *strikes*, donde un *strike* a una palabra es no estar presente en un filtro.

Para calcular el conjunto de palabras candidato para p basta tomar la unión de cualesquiera $2u + 1$ filtros de p, ya que cualquier palabra que sea semi-anagrama de p está en al menos uno de esos filtros. Posteriormente, se calcula la intersección entre el conjunto de candidatos y de cada filtro.

Si un candidato no se encuentra en un filtro, acumula un *strike*. En el dado caso que un candidato acumule $2u + 1$ *strikes*, se retira del conjunto de candidatos.

Finalmente, si la suma de los *strikes* acumulados y la diferencia de longitud entre una palabra candidato y objetivo es mayor a $2u$, también se retira del conjunto de candidatos. Esto se debe a que los errores de tipo remplazo de letra no afectan en la longitud de la palabra, pero son los que afectan en dos filtros; y los errores de tipo eliminación y agregación de letra afectan en a lo más un filtro, pero también afectan en una unidad la longitud de la palabra.

2. Filtrado individual

Una vez obtenido un conjunto de candidatos que son semi-anagramas de la palabra objetivo, se realiza un proceso que hace una comparación lineal entre cada palabra candidato y la palabra objetivo. Primero se compara si la palabras son iguales, de no ser así se realiza otro proceso con el objetivo de descartar las palabras que tienen un orden considerablemente diferente al de la palabra objetivo.

Este proceso se realiza obteniendo todas las $2u+1$ secuencias de letras consecutivas de cada palabra y comparando, en orden de aparición, la intersección de sus elementos, un ejemplo se presenta en la Tabla 1.

3. Decisión final

Una vez obtenido un conjunto menor de palabras a comparar con la palabra objetivo, se calcula la distancia de Levenshtein de esta con cada una de las palabras del conjunto. Si la distancia (ponderando con la longitud de la palabra) de una palabra candidato a la objetivo es menor a cierto rango definido previamente, se establece que la palabra apareció en el texto y se puede agregar a la lista de palabras.

5.3. Selección de enfermedades por lista de palabras

Una vez obtenido una lista, en orden, de las palabras del conjunto que aparecen en el texto, se realiza un proceso de selección para ver que enfermedades hacen referencia a las palabras en la lista.

Usando la relación entre las palabras y las enfermedades que se guardo previamente, se implementó un algoritmo glotón. En el cual, para definir la primera enfermedad, se toma desde la primera palabra en la lista la mayor secuencia de palabras consecutivas que hagan referencia a una o más enfermedades (en el dado caso que las mismas palabras aparezcan en más de una enfermedad).

Dada esta secuencia de palabras se calcula la relación que tienen con la enfermedad en la que aparecen, sumando el valor individual de las relaciones de cada palabra. Si el valor calculado es mayor a cierto umbral definido previamente, se define que la enfermedad aparece en el texto. Se eliminan las palabras de la lista y se vuelve a repetir el proceso hasta que no queden elementos en la lista.

6. Resultados

A continuación se presentan los resultados de los métodos de anonimización y detección de comorbilidades.

6.1. Anonimización

La metodología usada para anonimizar las notas médicas fue bastante efectiva cubriendo todas las menciones del nombre del paciente, incluyendo las instancias en donde hay errores en la escritura de este, sin embargo a su vez puede presentar problemas. Un problema es la sustitución de palabras que tienen una similitud con alguno de los elementos que compone el nombre.

Si bien experimentando con el umbral de la distancia de Levenshtein se puede evitar el remplazo de varias palabras, hay casos en donde dos palabras son demasiado parecidas para evitar que sean confundidas por la misma palabra con un error en su escritura. Ejemplo: Daniel y Daniela. La Figura 1 muestra una nota médica que ya pasó por el proceso de anonimización, donde el nombre del paciente fue reemplazado por $<paciente>$.

6.2. Detección de comorbilidades

La metodología que se desarrolló fue pensada para buscar en notas médicas los elementos del listado del CIE-10 de enfermedades y problemas relacionados a la salud, sin embargo al realizarse el trabajo se procuró que pudiera funcionar para cualquier tipo de texto y cualquier lista con una cantidad extensa de elementos que consten de una o más palabras.

Para la evaluación de esta tarea se contó con la ayuda de personal médico que etiquetó las comorbilidades que aparecieron en las notas médicas, como se menciono en la Sección de Corpus.

La comparación uno a uno de términos médicos del listado del CIE-10 con los términos usados en las notas médicas haría complicada la evaluación. Por ello, para la evaluación se optó por usar una lista de todas las comorbilidades encontradas en las 98 notas médicas en lugar del listado del CIE-10.

Tabla 2. Evaluación detección de comorbilidades por umbral de similitud entre palabras.

Umbral	**70-100**	**85-100**	**99-100**
Recall	92.63	95.78	81.05
Precisión	20.37	32.15	33.62
F1	33.39	48.14	47.52

Tabla 3. Comparación de comorbilidades en una nota médica y de resultados con listado CIE-10 y total de comorbilidades usada para la evaluación.

Comorbilidades etiquetadas	***Detección listado CIE-10***	***Detección listado etiquetas***
DIABETES MELLITUS 2	DIABETES MELLITUS	DIABETES MELLITUS 2
HIPERTENSION ARTERIAL SISTEMICA		HIPERTENSION ARTERIAL SISTEMICA, HIPERTENSION ARTERIAL SISTEMATICA
EXFUMADOR	OTRAS ENFERMEDADES CARDIOPULMONARES	EXFUMADOR
		COLECISTECTOMIA, HIPOKALEMIA
	AISLAMIENTO, DISNEA, TOS, CEFALEA	

Se tomaron aleatoriamente 30 notas médicas para la evaluación y se usaron 3 distintos umbrales para la similitud entre palabras.

El umbral de 99 a 100 de similitud entre palabras es equivalente a no aceptar errores o alteraciones en una palabra (sin considerar acentos).

Se compararon los resultados entre el etiquetado y los resultado de la metodología presentada en este documento, los resultado se muestran en la Tabla 2. El mejor desempeño se logra con el umbral entre 85 a 100 de similitud entre palabras, equivalente a permitir errores de una o dos letras en las palabras.

La metodología desarrollada sirve para la búsqueda de múltiples elementos de más de una palabra en un texto, donde se considera que las palabras pueden tener errores en su escritura. Alta importancia tiene en su efectividad la lista de elementos a buscar.

En la Tabla 3 se presenta la variación en los resultado obtenidos al usar diferentes listas y se compara con el etiquetado real de la nota médica.

En el caso de comorbilidades y el uso del listado del CIE-10 cabe destacar dos problemas que se presentaron. El primero es que la distinción entre comorbilidades y enfermedades o problemas de la salud puede llegar a ser bastante considerable, abarcando estos últimos síntomas.

El segundo problema proviene de las particularidades que puede llegar a tener a cada uno de las enfermedades o problemas de la salud, no mencionadas en la descripción

directa. Ejemplo: I10 Hipertensión esencial (primaria) cubre el caso de Hipertensión arterial sistémica.

7. Conclusiones

Los métodos propuestos para la realización de este trabajo son bastante útiles cuando no se cuenta con una cantidad muy grande de datos para entrenar modelos de aprendizaje supervisado.

Particularmente, el uso de la distancia de Levenshtein para la anonimización puede cubrir múltiples omisiones de usarse otro método. Por el lado de la detección de comorbilidades, el trabajo realizado puede servir para múltiples tareas de la misma índole (detección de una colección amplia de términos en un texto).

Como trabajo a futuro se sugiere el uso de un mejor listado de posibles comorbilidades, como lo sería uno realizado por doctores, enfocado en las comorbilidades más relevantes. Al igual se propone en un futuro realizar las tareas realizadas con métodos de aprendizaje de máquina.

Referencias

1. Allen, J.: Natural Language Understanding. Benjamin Cummings (1987)
2. Backurs, A., Indyk, P.: Edit distance cannot be computed in strongly subquadratic time (unless seth is false). In: Proceedings of the forty-seventh annual ACM symposium on Theory of computing. pp. 51–58 (2015)
3. Martín-Vegue, A., Vázquez-Barquero, J., Castanedo, S. H.: Cie-10 (i): Introducción, historia y estructura general. Papeles medicos, vol. 11, no. 1, pp. 24–35 (2002)
4. Plasencia-Urizarri, T. M., Aguilera-Rodríguez, R., Almaguer-Mederos, L. E.: Comorbilidades y gravedad clínica de la COVID-19: Revisión sistemática y meta-análisis. Revista Habanera de Ciencias Médicas, vol. 19 (2020)
5. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C. D.: Stanza: A python natural language processing toolkit for many human languages, (2020)
6. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information processing & management, vol. 24, no. 5, pp. 513–523 (1988)
7. Stubblebine, T.: Regular expression pocket reference: Regular expressions for perl, ruby, PHP, python, C, java and .NET (2007)
8. Thabit, K., AL-Ghuribi, S. M.: A new search algorithm for documents using blocks and words prefixes. Scientific Research and Essays, vol. 8, no. 16, pp. 640–648 (2013)
9. Viveros-Jiménez, F., Sanchez-Perez, M. A., Gómez-Adorno, H., Posadas-Durán, J. P., Sidorov, G., Gelbukh, A.: Improving the boilerpipe algorithm for boilerplate removal in news articles using html tree structure. Computación y Sistemas, vol. 22, no. 2, pp. 483–489 (2018)
10. Zhang, Y., Ma, X., Song, G.: Chinese medical concept normalization by using text and comorbidity network embedding. In: 2018 IEEE International Conference on Data Mining (ICDM). pp. 777–786 (2018)

ISSN 1870-4069

Aprendizaje automático para la detección de cáncer de mama

María de la Luz Escobar, José I. De la Rosa,
Carlos E. Galván-Tejada, Jorge I. Galvan-Tejada,
Hamurabi Gamboa-Rosales, Jose M. Celaya-Padilla

Universidad Autónoma de Zacatecas,
Departamento de Ingenería Eléctrica,
México

{escobarmaria50, jose.celaya}@uaz.edu.mx

Resumen. A nivel mundial, el cáncer de mama es uno de los tipos de cáncer que ha provocado mayor número de decesos entre la población femenina. Un diagnóstico temprano de la enfermedad permite que las pacientes logren aumentar la expectativa de vida, o en el mejor de casos superar la enfermedad. Recientemente, los sistemas asistidos por computadora se han enfocado en el estudio de imágenes mastográficas, como herramienta de apoyo para al análisis de cáncer. El estudio de la textura, la forma, el color, así como descriptores estadísticos presentes en la imagen contribuyen a la detección y al diagnóstico de la enfermedad; por ejemplo, la densidad de la masa se correlaciona con los tumores. Las investigaciones actuales que hacen uso de técnicas de aprendizaje automático, se centran en el mejoramiento del proceso de extracción de características de imágenes, permitiendo así la disminución de falsos positivos y negativos en la detección de cáncer. El propósito de este trabajo es presentar un marco de trabajo, para un proceso de mejora de la extracción de características imágenes. La metodología implementada describe un modelo basado en los datos de salida proporcionados por cuatro filtros para la reducción de ruido cuyo objetivo es disminuir los falsos positivos en la detección del cáncer de mama. El mejor resultado de los modelos de reducción de ruido propuestos para las métricas de precisión, sensibilidad y especificidad de 79,3 %, 92,5 % 55,5 %. respectivamente para el caso de la prueba de entrenamiento.

Palabras clave: Biomarcadores, ruido Gaussiano, falsos positivos, análisis de características.

Machine Learning Approach for Breast Cancer Detection

Abstract. The Breast cancer in women is the most common worldwide type of cancer and the leading cause of cancer death. An early disease diagnosis has caused that patients have been increasing life expectation, and in the best case they overcome illness. Recently, the CAD systems have been focused on

the study of the mammography, like supporting tools in the analysis of cancer. Image texture, shape, color, and statistical feature descriptors are used to obtain some image information. For example, how the density of masses is correlated with tumors. Actually, research in machine learning techniques is focused on the improvement in giving a significant process of extraction of features of mammography , the reduction in false-positive and false-negative in the detection of cancer. The purpose of this paper is to present a framework to obtain an improvement of the extraction of features. The implemented methodology describes a model based in the output data provided by four filters, which are based on noise reduction where the main goal is to reduce the false positive in detecting breast cancer. The best result of the proposed reducing noise models for accuracy, sensitivity, and specificity metric was 79.3 %, 92.5 % 55.5 % respectively for the training test.

Keywords: Biomarkers, Gaussian noise, false-positive, characteristics analysis.

1. Introducción

El nombre de cáncer refiere a un conjunto de enfermedades que se presentan en el cuerpo. El cáncer de mama ocupa el segundo lugar en decesos entre la población femenina a nivel mundial, solo superado por el cáncer de pulmón.

Las estimaciones del año de 2019 presentan 2,088,849 de nuevos casos y 629,679 millones de defunciones [1, 2]. En México se diagnosticaron 27,283, y se presentaron 6,884 muertes seguidas por el cáncer cervicouterino [1].

Sin embargo, algunas investigaciones refieren, a que mediante un diagnóstico temprano existe la posibilidad de un descenso en la letalidad femenina, ya que ésta sería más vulnerable a un tratamiento curativo, aumentando la esperanza de vida [2, 4].

1.1. Trabajos relacionados

El cáncer de mama es un tipo de cáncer mortal entre las mujeres en todo el mundo, alcanzando 9,6 millones de muertes y 2 millones de casos en 2019 y cantidades similares para 2019 [1, 2].

Las Mamografías se utilizan como una herramienta para el diagnóstico de cáncer de mama, y cuando se encuentra en sus primeras etapas (diagnóstico precoz), los tratamientos son más eficientes, contribuyendo a reducir el número de muertes por esta enfermedad [3, 5].

El análisis asistido por computadora (En sus siglas en ingles CAD) permite ayudar a visualizar la forma, el contorno, la densidad y el perímetro de la masa, y cuya observación permite realizar una estimación de la clasificación de la lesión de cáncer de mama [6, 10].

Las diferentes propuestas en la literatura se sustentan en el desarrollo de técnicas informáticas para el mejoramiento de características, para un diagnóstico óptimo de la enfermedad.

Por ejemplo, Galván et al. [11], proponen un modelo multivariado para la clasificación de lesión de tumores benignos o malignos, mediante un algoritmo genético realizando un análisis de características morfológicas de las lesiones y obteniendo una clasificación. Otros estudios proponen modelos cuyo foco principal es la reducción de falsos positivos, utilizando técnicas de optimización en imágenes para clasificar las lesiones [12, 14].

Por otro lado, Hernández et al. proponen una reducción de falsos positivos mediante la clasificación grasa y grasa glandular, mediante un conjunto de características determinadas por micro-clasificaciones[15]. Otros enfoques de análisis multivalente han demostrado que la información de un pronóstico y factores predictivos se puede obtener al identificar el cáncer de mama en sus primeras etapas [16].

Entre las diferentes técnicas de procesamiento de imágenes digitales y reconocimiento de patrones que se han aplicado en la literatura en la detección de cáncer de mama, se encuentra el uso de información mutua y una selectividad para el diagnóstico, utilizada cuando la información está uniformemente distribuida [17].

Estudios anteriores relacionaron de descriptores de imágenes combinados con datos clínicos para el diagnóstico de cáncer de mama [18]. Un estudio a partir de las distribuciones de valores de intensidad, logra extraer en diferentes escalas cinco sub regiones, y cuya optimización se realiza a partir de una máquina de vector (SVM) [19].

El estudio de un esquema de CAD basado en casos utilizando un conjunto de características de densidad mamográfica global, textura, espiculación y similitud estructural seleccionadas de forma óptima de la imagen [20]. La extracción de características basadas en operaciones morfologicas y utilizando filtros lógicos de coordenadas, segmentación y mapas autorganizados son otra aportación para el mejoramiento de características para la predicción de calcificaciones [21] .

Otro enfoque basado la máquina de vectores de soporte (SVM) y el algoritmo de búsqueda gravitacional mixta (MGSA) se utilizado para detectar los tumores de cáncer de mama en imágenes de mamografía [22].

Tang diseño del sistema implementado las transformadas wavelet discretas y la transformada coseno de Fourier para analizar las imágenes de mamografía y extraer las características estadísticas [23].

1.2. Contribución y organización de artículo

Por tanto, este artículo propone una metodología novedosa para análisis de características obtenidas dentro de una imagen mamografía, utilizando un enfoque de clasificación de características y un modelo multivalente como clasificador de los tumores benignos o malignos en donde cada características proviene de un filtro con diferente PSNR.

El artículo se organizado en cinco partes. La primera sección comienza con una breve introducción sobre el cáncer de mama. En esta sección se presenta una visión general del estado del arte. En la sección 2 se describen el los materiales y métodos implementados en la metodología. Finalmente, las conclusiones se realizan en la sección 4 y la bibliografía en la sección 5.

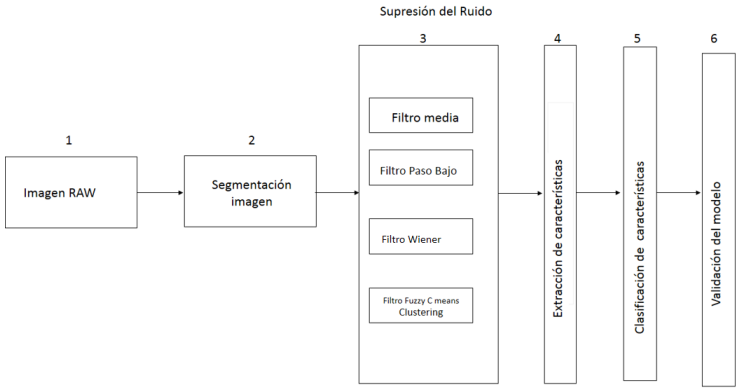


Fig. 1: Diagrama de Bloques de la metodología propuesta.

2. Materiales y métodos

Esta investigación propone un enfoque para el desarrollo de un biomarcador informático, para el análisis y extracción de características en una imagen mastográfica. Mediante técnicas de filtrado para la supresión del ruido Gaussiano, se genera un modelo multivalente, y a través de técnicas de inteligencia artificial se realiza la clasificación de imágenes de tumores benignos y malignos. Las pruebas de eficiencia y efectividad permiten evaluar el rendimiento del modelo propuesto en este trabajo (como se muestra en la Figura 1).

El desarrollo de biomarcador informático se efectúa en seis etapas. En la primera etapa se realiza la adquisición de una imagen con un formato JLPEG, el cual es convertido a un formato PGM. (2) La etapa de segmentación permite abstraer la mama o zona de estudio (ROI), convirtiendo toda la información fuera de ella en ceros. La inferencia del ruido se suprime mediante filtros mediana, promedio, Winer y Fuzzy C Means, pero coadyuvando a resaltar, suprimir u ocultar información contenida en la imagen.

En la etapa (4), se extrae un conjunto de características estadísticas para utilizarlas como medio de caracterización de los senos. Posteriormente, (5) mediante la implementación de una red de retro- propagación se realiza la clasificación entre lesiones benignas o malignas. Finalmente, la validación del modelo propuesto (6), se calcula utilizando el ROC (Área bajo la curva), cuyo término se refiere a la capacidad del modelo para predecir correctamente la clase benigna o maligna.

2.1. Adquisición y segmentación de la imagen

Las imagenes utilizadas en esta investigación son datos recolectados de una base de datos DDSM (Digital Data base for Screening Mammograph), provenientes del Hospital General de Massachusetts, de la Escuela de Medicina de la universidad del Sur Florida, de laboratorio nacional Sandia y de la Facultad de Medicina de la Universidad de Washington.

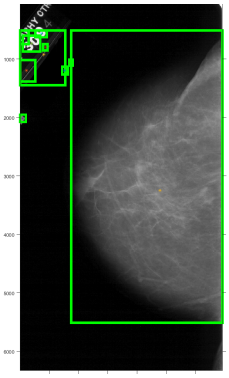


Fig. 2: Imagen Segmentada.

Esta base de datos tiene el propósito del desarrollo de nuevas técnicas para el diágnostico y detección de cáncer. En este proyecto, se han seleccionado un conjunto de 120 imágenes, de las cuales 60 imágenes son benignas y 60 imágenes malignas.

Antes de la segmentación mamaria, se realiza la conversión de formato jpg sin pérdidas (LJPEG formato antiguo), a un formato PNM (Portable Any Map) sin péridas de resolución, para posteriormente ser leídas por funciones de Matlab. [24]

La etapa de segmentación, se centra en la caracterización de la imagen de fondo para crear una imagen binaria que se utiliza como máscara de segmentación. Está técnica valida a un grupo de píxeles basados en un umbral global. El método consiste en encontrar la región objetivo que puede ser una aproximación a la zona de la mama (Eq. 1):

$$\text{máx}(ROI) = \begin{cases} P(I(R_i)_{i,j}) > 50, & 1, \\ \text{otra manera}, & 0. \end{cases} \tag{1}$$

La segmentación valida un grupo de píxeles en diferentes áreas de la imagen. Esta técnica se basa en un umbral global. Por lo tanto, los objetivos seleccionados tienen un valor de umbral superior a 50, por el contrario cero. El enfoque de la segmentación es seleccionar la mayor área que representada la anatomía de la mama o ROI (Fig. 2).

2.2. Filtrado de la señal

La tarea fundamental de la etapa de filtrado, es la mejora de la calidad de las características extraídas de la imagen, proceso fundamental para una predicción correcta en la clasificación de tumores malignos y benignos.

En la fase de adquisición de la imagen, las imágenes son contaminadas por ruido (señal aleatoria) y en las imágenes Médicas, el ruido puede modelarse con una distribución gaussiana.

La eliminación de ruido de las imágenes es una importante tarea de procesamiento de imágenes. Existen muchas formas de eliminar el ruido de una imagen o de un conjunto de datos. Tradicionalmente, se han utilizado modelos lineales y no lineales, los cuales hemos implementados en esta investigación:

Tabla 1: Ecuaciones que representa forma física de la ROI en la imagen.

Características	**Descripción**
areaq,p	$\sum_u \sum_v u^q v^p(i,j)$
Perímetro	$\sum_{t=1}^{2} 55 \parallel x_{t-1} - x_t \parallel + \parallel x_n - x_1 \parallel$
Center de masas	$\sum_u \sum_v (u-\bar{x})^p (v-\bar{y})^q I(u,v)$
Orientación	$\theta = 1/2tan^{-1} \frac{\mu_{11}}{\mu_{20}-\mu_{02}}$
Excentricidad	$\epsilon = \frac{(\mu_{20}-\mu_{02})^2+4\mu_{11}}{area}$
Solido	$\frac{area}{H}$
Extentención	$\frac{area}{B}$
Circularida	$4\pi \frac{area}{perimeter^2}$
Número de Euler	$E = N - H$
Longitud de eje Menor	$\sqrt{(x_2 - X_x 1)^2 + (y_2 - Y_1)^2}$
Redondez	$\frac{4\pi area}{(convexPerimeter)^2}$
Regularidad	$max(F_k)$

Filtro promediador se considera filtro como un filtro paso bajas, y se basa en remplazar cada píxel central por el promedio de nivel de gris de cada vecino. Este tipo sirve para resaltar componentes a gran escala eliminando la variabilidad local.

Sin embargo, el efecto de suavizado en los bordes y detalles de contraste son desventajas al implementar este este tipo de filtros [25].

Filtro de la Mediana se utiliza en la reducción de desenfoque de los bordes, sustituyendo del valor de gris del píxel por la mediana de los valores vecinos. El problema de la implementación de este tipo de filtros es determinar el valor central de cada píxel incluidos de los incluidos en la ventana [25].

El Filtro de Wiener es tipo de filtros es utilizado para la reducción de ruido aditivo, minimizando el error medio cuadrático entre la señal estimada y la señal deseada. La desventaja de este método es que no da una imagen clara [25].

Finalmente, el filtro robusto fuzzy C Means es un tipo de filtro que se utiliza para obtener más robustez en la reducción de cualquier tipo de ruido o en la combinación de ellos[26].

2.3. Extracción de características

La máscara de segmentación y los filtros de imagen se utilizan para caracterizar los tejidos mamarios, mediante descriptores estadísticos, de forma y de intensidad los cuales son extraídos de la imagen como se muestran en la ecuación (2):

$$I(i,j) = \begin{cases} 1\ (u,v) & \epsilon \quad C, \\ 0\ (u,v) & \notin \quad C. \end{cases} \tag{2}$$

Las características geométricas del borde frente a la región en imágenes binarias se definen en la Tabla 1.

Tabla 2: Ecuaciones que representan la intensity de la imagen.

Características	**Descripción**
Entropía	$e = (z_i)log\rho(z_i),$
Contraste	$\sum_{i=1}^{n}\sum_{j=1}^{m} y_{i,j}$
Correlacción	$\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{(i-\mu_i)(j-\mu_j)}{\sigma_i\sigma_j},$
Energía o uniformidad	$\sum_{i=1}^{n}\sum_{j=1}^{m} y_{i,j}^2$
Homogeneidad	$\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{y_{i,j}}{1+\lvert i-j\rvert}$

Tabla 3: Descriptores de intensidad.

Característica	**Descripción**
Media	$\frac{1}{n}\sum_{i=1}^{n} X_i,$
Desviación estandar	$\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{x})^2}$
Oblicuidad	$\frac{\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{x})^3}{(\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}X_i-\bar{x})^2})^3}$
Curtosis	$\frac{\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{x})^4}{(\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}X_i-\bar{x})^2})^2} - 3$
Mínimo	Mínimo intensidad de la ROI
Máximo	Máximo intensidad de la ROI

Las Tablas a continuación representan la matriz de concurrencia, representada por la textura de la ROI:

$$C = \frac{1}{n}A. \tag{3}$$

La Ecuación (3) muestra la matriz de Concurrencia donde n representa el operador de posición y A es una matriz del tamaño $N * N$, representa los puntos con niveles de gris, Tabla (2).

Los momentos estadísticos de una imagen son representados mediante medidas: como las media, desviación estándar e histogramas basados en la textura (Tabla 3).

2.4. Generación del modelo

Con cada salida de los filtros, se genera un vector de características Xi , y la unión de todas las características de salida de los filtros proporciona un modelo multivariante (Eq. 4 y 5):

$$Y_i = X_{f1} \cup X_{f2} \cup X_{f3} \cup X_{f4}, \tag{4}$$

$$Y_i = X_{f1,1} + X_{f1,2} + X_{f1,3} + X_{f1,4} + ... + X_{f4,147}. \tag{5}$$

En primer lugar, los vectores Yi de Y se normalizan, la media es cero, y la desviación estándar 1. El proceso para generar un modelo utilizando una red neuronal es representado en la Figura 3. La red neuronal artificial implementada con la herramienta MatLab proporciona la clasificación de tumores malignos o benignos.

El algoritmo de inteligencia artifical utilizado en esta red neuronal es, topología "backpropagation". Esta red se implementaron 10 capas ocultas, dos capas de salida binarias y el tipo aprendizaje utilizado es supervisado.

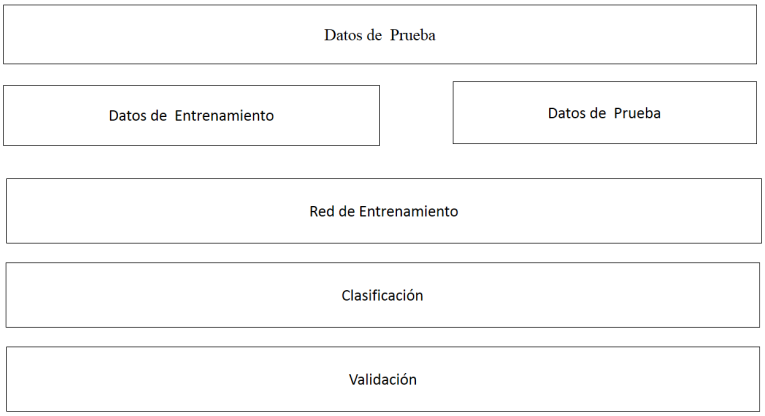


Fig. 3: Imagen Segmentada.

La característica de la red neuronal es la velocidad de convergencia debido al algoritmo de gradiente conjugado, con una velocidad de convergencia superior al descenso del gradiente [27].

La función de activación, con la cual trabaja esta red es la función sigmoidal. Para esta investigación se utilizaron un total de 118 sujetos, agrupados en 75 casos benignos y 75 malignos. El conjunto de datos se dividió en un 70 % para el entrenamiento y un 15 % para la prueba, y el 15 % para la validación de los datos (Fig. 3).

La precisión se utilizó como métrica para valorar el rendimiento de la red neuronal, y el error medio cuadrático (MSE) se utilizó como métrica para optimizar los pesos de la red neuronal.

3. Resultados

Una vez entrenada la red neuronal, en la matriz de confusión representada en la Figura 5a , podemos ver que el modelo alcanzó una precisión de entrenamiento del 75,6 %, una sensibilidad del 87,3 % y una especificidad del 53.6 %. La tasa de falsos positivos fue del 13 % de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 46.4 %.

En la prueba 4 (B), el modelo alcanzó una precisión de prueba del 66.6 %, una sensibilidad del 83.3 % y una especificidad del 33.3 %.

La tasa de falsos positivos fue del 16.7 % de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 33.3 %.

En la prueba 4 (C), el modelo alcanzó una precisión de prueba del 61.6 %, una sensibilidad del 81.1 % y una especificidad del 28.6 %.

La tasa de falsos positivos fue del 18.2 % a de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 71.4 %.

En la prueba 4 (d), el modelo alcanzó una precisión de prueba del 72.2 %, una sensibilidad del 87.5 % y una especificidad del 46.3 %.

(a) Modelo generado por una Imagen Cruda. (b) Modelo generado por una Imagen filtrada.

Fig. 4: Comparación de métrica de Presición, sensitividad y especificidad de los dos modelos porpuestos mediante Matriz de confusión.

La tasa de falsos positivos fue del 14.3 % de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 46.3 %.

Figura 5b, podemos ver que el modelo alcanzó una precisión de entrenamiento del 75,6 %, una sensibilidad del 87,3 % y una especificidad del 53.6 %. La tasa de falsos positivos fue del 13 % a de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 46.4 %. En la prueba 4 (B), el modelo alcanzó una precisión de prueba del 66.6 %, una sensibilidad del 83.3 % y una especificidad del 33.3 %. La tasa de falsos positivos fue del 16.7 % de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 33.3 %.

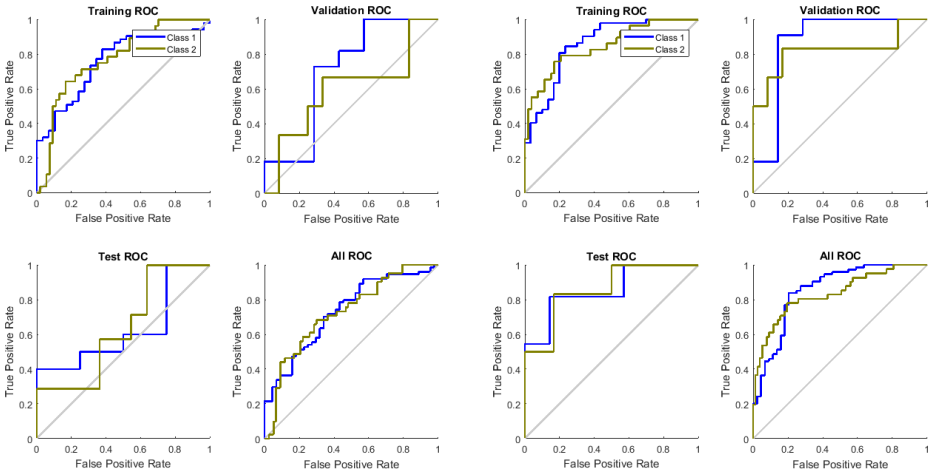
En la prueba 4 (c), el modelo alcanzó una precisión de prueba del 61.6 %, una sensibilidad del 81.1 % y una especificidad del 28.6 %. La tasa de falsos positivos fue del 18.2 % a de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 71.4 %.

En la prueba 4 (d), el modelo alcanzó una precisión de prueba del 72.2 %, una sensibilidad del 87.5 % y una especificidad del 46.3 %. La tasa de falsos positivos fue del 14.3 % de clasificaciones correctas, y la tasa de clasificaciones erróneas fue del 53.7 %.

El gráfico ROC de la Figura 5 muestra que la sensibilidad y la especificidad, muestran una clasificación perfecta entre imágenes benigna o maligna.

El modelo en presencia de ruido obtuvo un Área Bajo la Curva (AUC) de entrenamiento de = 0.78, en la prueba y la validación el modelo alcanzó un AUC de 0.68 y 0.78 respectivamente.

El modelo con reducción ruido obtuvo un Área Bajo la Curva (AUC) de entrenamiento de = 0.61, en la prueba y la validación el modelo alcanzó un AUC de 0.3 y 0.6 respectivamente.



(b) Modelo generado por una Imagen filtrada. (a) Modelo generado por una Imagen filtrada.

Fig. 5: Comparación de métrica de Presición por los dos modelos porpuestos mediante tablas ROC.

La comparativa de las matrices de confusión, así como las tablas ROC muestran un mejor rendimiento con el enfoque propuesto en esta investigación. Basado en la combinación de características se genera en un modelo a partir de la implementación de filtros para suprimir el ruido Gaussiano (4b y 5b).

La extracción de características con la imagen cruda, presenta un menor rendimiento acorde a las métricas de precisión, efectividad y sensibilidad (4a y 5a). Cabe hacer mención que los filtros implementados individuales no tendrían el mejor rendimiento, debido desventajas propias de los mismos, como el suvizado de los bordes, el bajo contraste, las imágenes que no son claras etc.

La Figura 6 muestra los errores entre los valores objetivos y los valores predichos después del entrenamiento los cuales son cercanos a cero. Lo que significa que el 70 % de los datos predichos por la red neuronal ha realizado con éxito la predicción. Sin embargo, un mayor número de conjuntos de datos provoca un desequilibrio en la red neuronal.

Este problema se puede resolver reduciendo el número de características en el conjunto de datos. En este caso, el error cero 0.001502 y los datos de entrenamiento están entre 27 y 10, los de validación y prueba 33 y 10, divididos en 20 bins. El rendimiento de la red neuronal se midió en términos del error cuadrático medio Fig. 7.

El gráfico presenta perturbaciones debido a la gran cantidad de descriptores, generados por los filtros de salida; por consecuencia la tasa de aprendizaje de la red será lenta. Esto se puede mejorar a tráves de técnicas de selección de características, buscando solo las aquella caracterísitcas que proporcionen información relevante.

El mejor rendimiento de la red se obtiene en la época veintidós para los datos de validación, prueba y ensayo respectivamente. En la Tabla 4 se hace referencia algunos de los descriptores que presentaron mejor rendimiento individualmente.

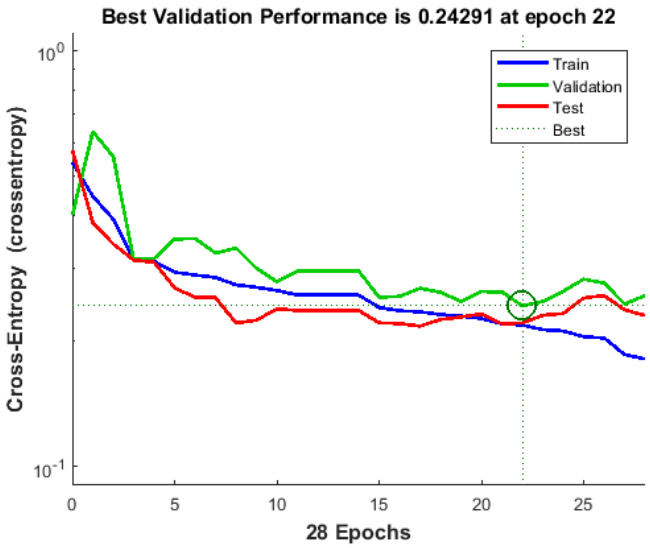


Fig. 6: Gráfica de ROC (A) Entrenamiento, (B) Pruebas, (C) Validación, (D) Conjunto de datos completo.

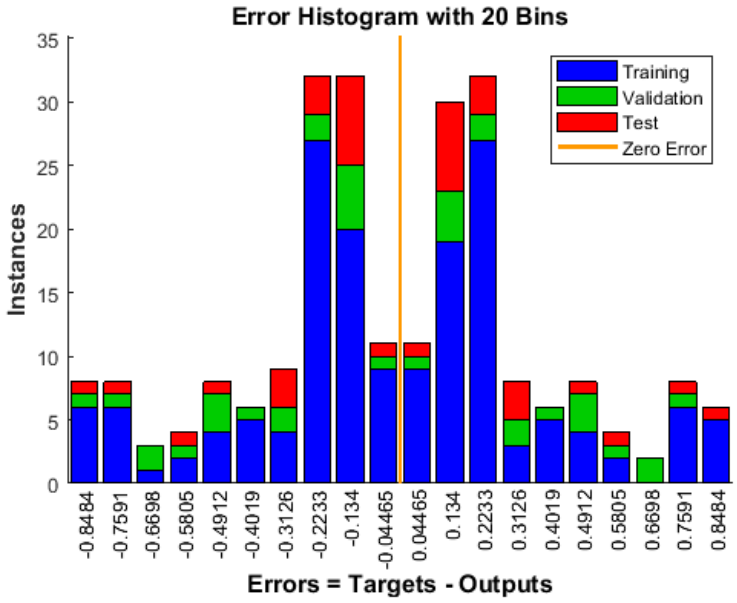


Fig. 7: Funcionamiento de la Red Neuronal.

Sin embargo, se puede notar que los descriptores de intensidad, proporciona un factor rendimiento, mucho mayor en la clasificación de imágenes en presencia del cáncer y tejido sano.

Tabla 4: Características individuales con el mejor desempeño en la clasificación.

Características extraídas de cada filtro	**Precisión**
Brillo (Filtro de media)	85 %
Perímetro	80 %
Uniformidad(FuzzyCMeans)	75 %
Entropía (FuzyCMeans)	74 %
Contraste (FuzyCMeans)	74 %
Energía (Filtro Winner))	73 %
Entropía (Filtro Winner)	72 %
Diámetro	72 %
Máximo Intensidad (Filtro Winner)	72 %
Oblicuidad(Filter Winner)	70 %
Contraste (Filtro Mediana)	70 %
Brillo (Filtro Media)	70 %
Área	70 %
Orientación	70 %
Entropía (filter paso bajo)	70 %
Correlación (Filter Media)	70
Media Intensidad (Fuzzy C means clustering)	70 %
Uniformidad (Filtro Media)	70 %
Correlación (Filter Media)	70 %
Orientación	70 %

4. Conclusiones

El presente estudio es un intento preliminar de mejorar la clasificación de tumores benignos y malignos estudiando los efectos del ruido Gaussiano para obtener un modelo que presente mejores predicciones incorporandolos al trabajo de los radiólogos para ayudar como segunda opinión, y además, puede utilizarse para evitar biopsias innecesarias.

La inclusión de filtros relacionados con el ruido produjo al menos cuatro características con una precisión de hasta 79.3 % para predecir los tumores. Sin embargo, el elevado número de características provocó un bajo rendimiento. Se sugiere que una selección de características más robusta y en combinación con características derivadas de dichos filtros podría ser una buena opción para mejorar la predicción la etapa de clasificación de tumores.

Para trabajos futuros se estudiarán los efectos de ruido Gaussiano y Cuántico con la finalidad los mejorar el modelos para las clasificaciones de tumores benignos y malignos.

Referencias

1. World Health Organization: Cancer burden rises to 18.1 million new cases and 9.6 million cancer deaths in 2018. International Agency for Research on Cancer (2018)

2. Ferlay, J., Colombet, M., Soerjomataram, I., Mathers, C., Parkin, D. M., Piñeros, M., Znaor, A., Bray, F.: Estimating the global cancer incidence and mortality in 2018: GLOBOCAN sources and methods. International Journal of Cancer, vol. 144, no. 8, pp. 1941–1953 (2019) doi: 10.1002/ijc.31937
3. Kalager, M., Zelen, M., Langmark, F., Adami, H. O.: Effect of screening mammography on breast-cancer mortality in Norway. The New England Journal of Medicine, vol. 363, no. 13, pp. 1203–1210 (2010) doi: 10.1056/NEJMoa1000727
4. Nelson, H. D., Fu, R., Cantor, A., Pappas, M., Daeges, M., Humphrey, L.: Effectiveness of breast cancer screening: Systematic review and meta-analysis to update the 2009 U.S. Preventive services task force recommendation. Annals of internal medicine, vol. 164, no. 4, pp. 244–255 (2016) doi: 10.7326/M15-0969
5. Marmot M., Altman, D. G., Cameron, D. A., Dewar, J. A., Thompson, S. G., Wilcox, M.: The benefits and harms of breast cancer screening: An independent review. British Journal of Cancer, vol. 108, no. 11, pp. 2205–2240 (2012) doi: 10.1038/bjc.2013.177
6. Freer, T. W., Ulissey, M. J.: Screening mammography with computer aided detection: Prospective study of 12,860 patients in a community breast center. Radiology, vol. 220, no. 3, pp. 781–786 (2001) doi: 10.1148/radiol.2203001282
7. Ng, K. H., Muttarak, M.: Advances in mammography have improved early detection of breast cancer. Journal-Hong Kong College of Radiologists, vol. 6, pp. 126–131 (2003)
8. Leung, J. W., Margolin, F. R., Dee, K. E., Jacobs, R. P., Denny, S. R., Schrumpf, J. D.: Performance parameters for screening and diagnostic mammography in a community practice: Are there differences between specialists and general radiologists? American Journal of Roentgenology, vol. 188, no. 1, pp. 236–241 (2007) doi: 10.2214/AJR.05.1581
9. Oeffinger, K. C., Fontham, E. T. H., Etzioni, R., Herzig, A., Michaelson, J. S., Shih, Y. C. T., Walter, L. C., Church, T. R., Flowers, C. R., LaMonte, S. J.: Breast cancer screening for women at average risk: 2015 Guideline update from the American cancer society. Journal of the American Medical Association, vol. 314, no. 15, pp. 1599–1614 (2015) doi: 10.1001/jama.2015.12783
10. Gardezi, S. J. S., Elazab, A., Lei, B., Wang, T.: Breast cancer detection and diagnosis using mammographic data: Systematic review. Journal of Medical Internet Research, vol. 21, no. 7 (2019) doi: 10.2196/14464
11. Galván-Tejada, C. E., Zanella-Calzada, L. A., Galván-Tejada, J., Celaya-Padilla, J. M., Gamboa-Rosales, H., Garza-Veloz, I., Martinez-Fierro, M. L.: Multivariate feature selection of image descriptors data for breast cancer with computer-assisted diagnosis. Diagnostics, vol. 7, no. 1, pp. 9 (2017) doi: 10.3390/diagnostics7010009
12. Li, Y., Chen, H., Rohde, G. K., Yao, C., Cheng, L.: Texton analysis for mass classification in mammograms. Pattern Recognition Letters, vol. 52, pp. 87–93 (2015) doi: 10.1016/j.patrec.2014.10.008
13. Wu, Y., Giger, M. L., Doi, K., Vyborny, C. J., Schmidt, R. A., Metz, C. E.: Artificial neural networks in mammography: Application to decision making in the diagnosis of breast cancer. Radiology, vol. 187, no. 1, pp. 81–87 (1993) doi: 10.1148/radiology.187.1.8451441
14. Eltoukhy, M. M., Faye, I.: An adaptive threshold method for mass detection in mammographic images. In: IEEE International Conference on Signal and Image Processing Applications, pp. 374–378 (2013) doi:10.1109/ICSIPA.2013.6708036
15. Hernández-Capistrán, J., Martínez-Carballido, J. F., Rosas-Romero, R.: False positive reduction by an annular model as a set of few features for micro-calcification detection to assist early diagnosis of breast cancer. Journal of Medical Systems, vol. 42, no. 8, pp. 1–9 (2018)
16. Domínguez, M. A., Marcos, M., Meirino, R., Villa-Franca, E., Dueñas, M. T., Arias, F., Martínez, E.: Prognostic and predictive factors in early breast cancer (2001)

17. Khaire, U. M., Dhanalakshmi, R.: Stability of feature selection algorithm: A review. Journal of King Saud University-Computer and Information Sciences, pp. 1060–1073 (2019) doi: 10.1016/j.jksuci.2019.06.012
18. Moura, D. C., Guevara López, M. A.: An evaluation of image descriptors combined with clinical data for breast cancer diagnosis. International journal of Computer Assisted Radiology and Surgery, vol. 8, no. 4, pp. 561–574 (2013) doi: 10.1007/s11548-013-0838-2
19. Sun, W., Tseng, T. L., Qian, W., Zhang, J., Saltzstein, E. C., Zheng, B., Lure, F., Yu, H., Zhou, S.: Using multiscale texture and density features for near-term breast cancer risk analysis. Medical physics, vol. 42, no. 6, pp. 2853–2862 (2015) doi:10.1118/1.4919772
20. Tan, M., Aghaei, F., Wang, Y., Zheng, B.: Developing a new case based computer-aided detection scheme and an adaptive cueing method to improve performance in detecting mammographic lesions. Physics in Medicine and Biology, vol. 62, no. 2, pp. 358 (2017) doi: 10.1088/1361-6560/aa5081
21. Quintanilla-Domínguez, J., Ruiz-Pinales, J., Barrón-Adame, J. M., Guzmán-Cabrera, R.: Microcalcifications detection using image processing. Computacion y Sistemas, vol. 22, no. 1, pp. 291–300 (2018) doi: 10.13053/cys-22-1-2560
22. Melekoodappattu, J. G., Subbian, P. S. : A hybridized ELM for automatic micro calcification detection in mammogram images based on multi-scale features. Journal of Medical Systems, vol. 43, no. 7, pp. 1–12 (2019) doi: 10.1007/s10916-019-1316-3
23. Tang, X., Zhang, L., Zhang, W., Huang, X., Iosifidis, V., Liu, Z., Zhang, M., Messina, E., Zhang, J.: Using machine learning to automate mammogram images analysis. In: IEEE International Conference on Bioinformatics and Biomedicine, pp. 757–764 (2020) doi: 10.1109/BIBM49941.2020.9313247
24. University of South Florida: DDSM: Digital database for screening mammography (2006) http://www.eng.usf.edu/cvprg/Mammography/Database.html
25. Woods, R. E., González, R. C.: Digital image processing using Matlab. Education (2004)
26. Ahmed, M. N., Yamany, S. M., Mohamed, N., Farag, A. A., Moriarty, T.: A modified fuzzy c-means algorithm for bias field estimation and Segmentation of MRI data. IEEE Transactions on Medical Imaging, vol. 3, no. 21, pp. 193–199 (2002) doi: 10.1109/42.996338
27. Hagan, M. T., Demuth, H. B., Beale, M. H.: Neural Network Design (1995)

ISSN 1870-4069

Análisis de representaciones vectoriales en bitácoras de mantenimiento en la Industria: Hacia un sistema de recuperación de información

Jesús Roberto Enrique León Carmona[1], Samuel González-López[1], Esaú Villatoro-Tello[2,3], Jesús Miguel García-Gorrostieta[4]

[1] Instituto Tecnológico de Nogales,
Sonora,
México

[2] Universidad Autónoma Metropolitana,
Unidad Cuajimalpa,
México

[3] Idiap Research Institute,
Martigny,
Switzerland

[4] Universidad de la Sierra,
Sonora,
México

17341003@itnogales.edu.mx, samuel.gl@nogales.tecnm.mx,
evillatoro@cua.uam.mx, jgarcia@unisierra.edu.mx

Resumen. La identificación de información útil en textos a través de aplicaciones con diferentes técnicas de minería de datos es poco utilizada en el contexto industrial [1]. En este artículo se analizan representación como Word2Vec, Doc2Vec y TF-IDF para determinar la más adecuada para la tarea de recuperación de información en bitácoras de mantenimiento. Además, se propone una metodología para la recuperación de información la cual brinde ayuda en el área de producción analizando el texto de los mantenimientos previos de esa área. La metodología propuesta ayudará a la toma de decisiones dándole resultados al técnico de mantenimientos posibles soluciones previas con el mismo problema. Se observaron resultados alentadores por parte del modelo Word2Vec Skip-Gram para representar los documentos.

Palabras clave: Modelos de representación textual, líneas de producción, procesamiento del lenguaje natural.

Analysis of Vector Representations in Maintenance Logs in the Industry: Towards an Information Retrieval System

Abstract. The identification of useful information in texts through applications with different data mining techniques is little used in the industrial context [1].

In this article, representations such as Word2Vec, Doc2Vec and TF-IDF are analyzed to determine the most suitable for the task of retrieving information in maintenance logs. In addition, a methodology for the recovery of information is proposed which provides help in the maintenance area by analyzing the text of the previous maintenance in the production area. The proposed methodology will help decision-making, giving the maintenance technician results possible previous solutions with the same problem. Encouraging results were seen from the Word2Vec Skip-Gram model to represent the documents.

Keywords: Textual representation models, production lines, natural language processing.

1. Introducción

La identificación de información útil en textos a través de aplicaciones con diferentes técnicas de minería de datos es poco utilizada en el contexto industrial. Esto abre una brecha para poder explorar la información y transformarla en conocimiento útil [1].

La recuperación de información (RI) se ha desarrollado desde finales de la década de 1950. Actualmente adquiere un rol más importante por el valor que tiene la información, disponer o no de la información en tiempo y forma puede resultar en el éxito o fracaso de una operación. RI es el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información [2].

La recuperación de información es un área amplia, donde se abarcan diferentes temas, algunos computacionales como el almacenamiento y la organización; y otros relacionados con el lenguaje y los usuarios como la representación, la recuperación y la interpretación [3]. En los últimos años el crecimiento de las tecnologías junto con el procesamiento del lenguaje natural abre las puertas a las empresas donde se pueda extraer una cantidad considerable de texto para poder utilizar técnicas que nos ayude a tomar decisiones. La incertidumbre de cuándo ocurrirán tiempos caídos, fallas en alguna máquina o un mantenimiento preventivo es frecuente en las líneas de producción.

Una línea de producción involucra estaciones de trabajo que pueden automatizar los procesos de ensamble de algún producto. Estas fallas pueden suceder en cualquier momento de un proceso de producción. En este artículo proponemos una primera etapa de la metodología para la recuperación de información la cual brinde ayuda en el área de mantenimiento analizando el texto de los mantenimientos previos en el área de producción. El método incluye técnicas de procesamiento del lenguaje natural para analizar textos de mantenimiento, la construcción de diferentes modelos con los datos textuales y un método para analizar la similitud entre documentos.

Buscamos relacionar las causas raíz de un problema de una maquina con la solución que el técnico le dio a dicho problema. La metodología ayudará a la toma de decisiones dándole resultados al técnico de mantenimientos posibles soluciones previas con el mismo problema y brindando la posibilidad de elegir la solución se ajuste a la situación, ayudando a agilizar el proceso de encontrar la falla y reducir tiempos caídos [1][4].

[1] Corresponde al tiempo que la estación de trabajo no realiza su actividad debido alguna falla

2. Trabajos relacionados

El procesamiento de lenguaje natural (PLN) es una opción viable para resolver problemas dentro de la industria. Nuestro trabajo ha recurrido a técnicas para calcular representaciones vectoriales continuas de palabras a partir de conjuntos de datos muy grandes. La calidad de estas representaciones se mide en una tarea de similitud de palabras y los resultados se comparan con las técnicas de mejor rendimiento basadas en diferentes tipos de redes neuronales

Los modelos vectoriales de palabras o Word Embedding (en inglés) son un recurso que puede ser utilizado como insumo para la resolución de varios de los problemas del área de PLN. Un modelo vectorial consiste en la codificación de las palabras y/o frases en un vector numérico de grandes dimensiones. Esta codificación permite tener un mapeo de "(palabra, vector)" que permite identificar cada palabra con su correspondiente vector. Contar con este mapeo es importante ya que la gran mayoría de los algoritmos utilizados están pensados para trabajar con números (sobre todo las redes neuronales) [5].

El proceso comienza con la recolección de los datos, así las empresas pueden usar sensores de bajo costo, conectividad inalámbrica y herramientas de procesamiento de bigdata para que sea más barato y fácil recopilar datos de rendimiento real y monitorear el estado del equipo. Esto mediante el uso de algoritmos basados en datos que analizan la información recopilada de una máquina determinada y su entorno ambiental, y luego la procesa de regreso a la máquina para un control adaptativo para una planificación de producción eficaz, eficiente y la programación de mantenimiento a tiempo [6]. In [7] se presenta un método para predecir fallas en el proceso de manufactura, utilizando atributos no categorizados.

El primer paso del método es el agrupamiento de datos de aquellos procesos similares, posteriormente aplicaron técnicas de aprendizaje, construyendo por cada grupo formado un clasificador. Para la predicción primero buscan clasificar el dato en un grupo y después es clasificado. Los autores reportan una AUC (área bajo la curva ROC) de 0.69, lo que revela la complejidad del problema. El uso de una red Bayesiana para predecir fallas ha sido estudiada en [8]. En este trabajo presentan una metodología que incluye 4 pasos.

El primero refiere a la recolección de los datos, el segundo se enfoca al aprendizaje y optimización de la red Bayesiana. En el paso 3 se extraen patrones de todos los tipos de fallas por separado y en el último paso se realiza la predicción de fallas recurriendo a un conjunto de reglas. Reportaron diferentes desempeños por cada una de las reglas establecidas y usaron el índice de predictibilidad (PI) para medir el rendimiento.

Por otro lado, el enfoque de las empresas que usan datos de éxito/falla (Ground Truth), puede utilizar un marco de referencia sobre un aprendizaje supervisado, se pueden crear modelos basados en los datos de exitosos o no para estudiar el comportamiento de las predicciones midiéndolos, utilizando métricas de clasificación como precisión, recuerdo, f-score, exactitud. El utilizar modelos que comparen diferentes técnicas da una perspectiva más amplia de cómo se comportan los datos dentro de una empresa y dar una visión diferente [9].

Algunos de los datos de mantenimiento, alejándonos de los sensores, están escritos por el personal experto al realizar servicio a las instalaciones y equipos de manufactura en Excel como una cadena de palabras. Una de las estrategias para revisar estos datos

Tabla 1. Ejemplo de fallas y soluciones del corpus.

Diagnóstico del Operador	Diagnóstico del Técnico	Solución
Problemas con cortos	Programa movido	Se ajustó programa
Problemas con cortos	Boquilla no tira flux	Se ajustó flux y limpio boquilla
Por cortos	Turno anterior	Turno anterior
Cortos e insuficiencias en ws2	No sale fluxer suficiente	Se purgo fluxer
Se está apagando la maquina cada rato	Precalentador	Se revisó precalentador y reseteo equipo
Problemas con la flaxeadora	Bomba pierde presión	Válvula dañada
Baja temperatura en la olla	Maquina bloqueada	Se reseteo maquina

es proporcionar un análisis de las palabras claves para facilitar la identificación de tendencias, así el aprendizaje automático sirve como puente entre los datos textuales extraídos de los equipos y el personal ya que se utilizarán en el mapeo para clasificar los textos [10].

3. Métodos y materiales

3.1. Corpus

El corpus proviene de 6853 registros escritos por el técnico del área, el cual se almacena en una hoja de Excel, en él se encuentra el diagnóstico del operador "la observación del operador del problema", la descripción del problema por el técnico y la solución que este mismo le dio, la mayoría de estos casos contienen la descripción de dichos mantenimientos otros contienen casillas en blanco y otros con palabras o símbolos sin sentido. La Tabla 1 muestra algunos ejemplos de los registros escritos en mayúsculas.

Después de observar los datos provenientes de Excel en formato. xlsx se observan inconsistencias, posteriormente se lleva a un formato csv para ajustarlo para leer las palabras separadas por comas y no por casillas esto facilita a word2vec a vectorizar los datos. Donde cada coma "," representa una celda y cada salto de línea representa una nueva fila.

Salida del archivo csv:

- Fallas con el conveyor, conveyor se atora, se ajustó tornillos.
- Pallets atorados en sello, cambio de turno, cambio de turno.
- Problemas con la flaxciadora, problemas con la flaxer, se ajustó sensor.
- Los registros de mantenimiento a menudo se registran de manera desordenada y no estandarizada: Estos registros se pueden escribir a mano y luego transcribir o ingresar directamente desde el entorno operativo a través de dispositivos de entrada

limitados, computadoras. Los datos de entrada subsiguientes, aunque nominalmente son lenguaje natural, exhiben muchas características que hacen difícil el procesamiento de la entrada.

- El vocabulario utilizado en las descripciones es inconsistente: por lo general, no existe un conjunto estándar de términos utilizados para los nombres de las piezas mecánicas o las actividades que se realizan en ellas. Por ejemplo, un registro puede contener "cambiar el aceite" mientras que otro puede contener "reemplazar el fluido", ambos se refieren a la misma acción.
- El vocabulario puede no corresponder directamente a sistemas o componentes de interés: el personal de reparación puede referirse a un componente o parte de un sistema sin mencionar explícitamente el sistema que la empresa está interesada en monitorear. Por ejemplo, “problemas con cortos" en realidad, puede ser "pre-calentador no calienta" que debe clasificarse como una falla del sistema de coordenadas y no como una falla de cortos en sí.
- La entrada no está bien formada: Debido a las limitaciones de longitud de los caracteres y al probable tratamiento de las entradas del registro como tarea secundaria, las descripciones del texto se limitan a frases cortas o fragmentos de oraciones. Además, normalmente no se presta atención a la ortografía y otras reglas del lenguaje y, por lo tanto, los datos exhiben errores gramaticales y ortográficos. Por ejemplo, “tiempo caído no funciona”, contiene un error tipográfico, mientras que “se corrió pzas. Durante el tiquetse ajusto rpm altura y flux. Tiene 30 min run”, No menciona el tópico de la falla ni el porqué de ella.
- Son comunes la jerga y las abreviaturas: Debido al espacio de entrada limitado y las presiones de tiempo de los técnicos de mantenimiento, las entradas de registro suelen estar acompañadas de abreviaturas y contienen una gran cantidad de jerga. Por ejemplo, " limpieza de pot t/c 20 " podría ser de “tiempo caído”. El uso de la jerga puede variar desde términos generalmente conocidos hasta términos muy locales para máquinas o herramientas particulares.
- No se dispone de una gran cantidad de datos: Se trabaja con datos proporcionados de una empresa maquiladora, estos datos se han recopilado por unos meses de los registros de mantenimiento de alrededor de 6853 registros para una sola clase de máquina.

Está limitada la cantidad de datos provenientes de informes de mantenimientos en el área de producción, están en formato XLSX: se trata de un libro de Excel que no cuenta con ningún tipo de macros, y para mantener una integridad de los datos se pasan a formato CSV estos (del inglés comma-separated values) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas (o punto y coma en donde la coma es el separador decimal como en “se, corrió, pzas.

Durante, el, tiquetse, ajusto, rpm, altura, y, flux., Tiene, 30, min, run”,) y las filas por saltos de línea. El formato CSV es muy sencillo y no indica un juego de caracteres concreto, ni cómo van situados los bytes, ni el formato para el salto de línea. Estos puntos deben indicarse muchas veces al abrir el archivo, por ejemplo, con una hoja de cálculo.

La idea básica de separar los campos con una coma es muy clara, pero se vuelve complicada cuando los valores del campo también contienen comillas dobles o saltos de línea. Las implementaciones de CSV pueden no manejar esos datos, o usar comillas

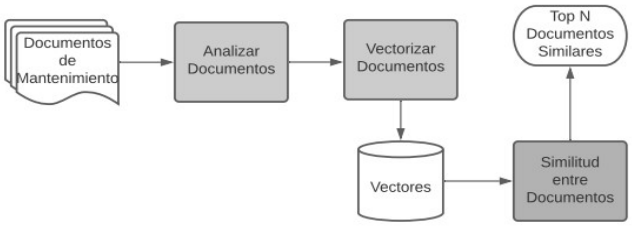


Fig. 1. Metodología propuesta.

de otra clase para envolver el campo. Algunos campos también necesitan embeber estas comillas, así que las implementaciones de CSV pueden incluir caracteres o secuencias de escape. Además, el término "CSV" también denota otros formatos de valores separados por delimitadores que usan delimitadores diferentes a la coma (como los valores separados por tabuladores).

Un delimitador que no está presente en los valores de los campos (como un tabulador) mantiene el formato simple. Es importante indicar que por el momento el corpus contiene información sensible de la empresa por la que no puede ser compartido en su estado actual.

3.2. Metodología

Se realizará un análisis utilizando técnicas de procesamiento de lenguaje natural, el primer paso es procesar los datos que provienen de los mantenimientos correctivos del área de producción que los técnicos realizan, para resolver dicho problema se planteara utilizar técnicas como Word2vec, entre otras y analizar dichos modelos para obtener el modelo con mejores resultados.

Se propone una comparación con cuatro técnicas de procesamiento de lenguaje para atender el problema desde diferentes puntos de vista. En la Fig. 1 se observa la metodología propuesta en la cual primeramente se analizan los documentos del corpus eliminamos palabras vacías y convertimos a minúsculas. Posteriormente vectorizamos los documentos para generar un vector para cada documento, para esto se probaron técnicas tradicionales como TF-IDF y técnicas basadas en Word embeddings como Word2vec y Doc2vec.

Una vez construidos los vectores de los documentos se utilizó la distancia coseno para determinar la similitud coseno entre la consulta y los documentos buscados. En esta primera etapa del estudio se busco identificar cual modelo de representación se adecuaba mejor a los datos. En una etapa futura del estudio se busca presentar y evaluar los Top N documentos similares encontrados.

4. Diseño experimental

Los experimentos realizados consisten en analizar representaciones como Word2vec, Doc2vec, TF-IDF para determinar cuál modelo de representación se adecua mejor a los datos. Para ello dividimos nuestro corpus en 90% datos de entrenamiento

Tabla 2. Resultados Similitud Coseno.

Modelo	Resultado
TF-IDF	0.54499435
Word2Vec-CBOW	0.89598442
Word2Vec-SKIP-GRAM	**0.94781154**
Doc2Vec-DBOW	0.80774255
Doc2Vec-PV-DM	0.78620946

(5,900) y 10% para datos para prueba (590), con el fin de comparar los vectores promedio resultantes del conjunto de entrenamiento contra el conjunto de prueba.

Finalmente analizamos la similitud coseno entre el vector promedio de entrenamiento y el vector promedio de prueba de los modelos creados. Para el modelo con Word2vec se obtiene una representación vectorial densa de palabras que capturan algo sobre su significado utilizando redes neuronales. La construcción del modelo se logra con la librería Gensim obteniendo vectores de tamaño 150 por cada palabra. Para obtener la representación vectorial de cada consulta se calcula el promedio de los Word embeddings. Finalmente se calcula la distancia coseno de vector promedio de entrenamiento y el vector promedio de prueba utilizando la herramienta sklearn. Para el modelo de Doc2Vec el cual genera una representación vectorizada de un grupo de palabras tomadas como una sola unidad.

El modelo modifica el algoritmo de word2vec para el aprendizaje no supervisado de representaciones continuas para bloques de texto más grandes, como oraciones, párrafos o documentos completos. Para la construcción de los vectores para cada documento utilizamos la librería de gensim con vectores de tamaño 150. Finalmente se calcula la distancia coseno de vector promedio de entrenamiento y el vector promedio de prueba utilizando la herramienta sklearn. El modelo de frecuencia de términos-frecuencia inversa de documentos TF-IDF es un modelo de bolsa de palabras con pesado.

El tamaño de los vectores generados está en función del tamaño del vocabulario del corpus. La generación de los vectores TF-IDF para cada documento se construyó utilizando la herramienta gensim. Finalmente se calcula la distancia coseno de vector promedio de entrenamiento y el vector promedio de prueba utilizando la herramienta sklearn.

5. Resultados

Los resultados de los diferentes modelos son comparados para determinar cuál modelo de representación se adecua mejor a los datos de los registros de mantenimientos correctivos.

Para ello utilizamos calculamos la distancia coseno entre el vector promedio del conjunto de entrenamiento contra el vector promedio del conjunto de prueba. En la

Tabla 2 observamos los resultados de la similitud coseno para cada modelo de representación. Al analizar los resultados notamos que algunos de los modelos son más adecuados para capturar la información textual y semántica de los registros de mantenimiento.

Suponemos que la similitud coseno entre los vectores promedio del conjunto de entrenamiento y prueba nos brinda una noción de que una representación que captura mejor la información textual de los registros logra una mayor cercanía. En base a esto al revisar los resultados de los diferentes modelos, notamos que no están tan alejados unos de otros, esto podría interpretarse de que se está reteniendo algo de información y que el mejor modelo para representar los documentos es Word2Vec Skip-Gram al tener una mayor puntuación.

6. Conclusiones y trabajo a futuro

Con las representaciones utilizadas pudimos observar que tan similares son dos documentos como si fuera un buscador tradicional al hacer búsqueda de concatenación de palabras, ya que nuestro corpus no nos permite trabajar con otras métricas por la falta de datos y su escasez en el léxico con los que los técnicos describen los mantenimientos realizados.

Los técnicos expertos realizan estos tickets de mantenimientos con su propia experiencia y los describen con sus palabras dejando por un lado la estandarización con la que se podrían escribir los tickets y así obtener un mejor análisis y utilizar otras métricas que ayuden a comprender el comportamiento de los datos textuales descritos por el personal de producción y los técnicos.

Esta investigación se utiliza la métrica de coseno inverso para mantener la integridad de los datos redactados por el personal intentando encontrar una relación entre lo descrito por los técnicos y las búsquedas que el usuario realiza al querer buscar una solución a un mantenimiento en común, al buscar una concatenación de palabras se espera obtener respuestas cercanas a lo descrito ya que la consulta realizada estará cercana "vectorialmente cercana" a la respuesta deseada intentando mantener las jergas, mal escrituras, mal formuladas, signos y diferentes formas de describir problemas y soluciones , como material para nutrir las búsquedas.

Con respecto a los resultados de Word2Vec skip-grams, este nos parece el más adecuado comparado con los demás modelos creados con Gensim por el hecho de realizar consultas de concatenación de palabras. Este modelo intentara predecir las palabras "vectores" vecinas de esta búsqueda, ya que estas palabras se estarán asociando con sus palabras vecinas y dar una respuesta más cercana a la buscada. En este modelo no se interesa del todo las entradas y salidas de la red, sino que el objetivo es simplemente aprender los pesos de la capa oculta que son en realidad los vectores de las palabras que se está intentando aprender de ellas.

La tarea para el modelo skip-gram seria, dada una palabra intentar predecir las palabras vecinas, y esto está definido por la ventana que en nuestro caso fue de 10. Podemos concluir que son viables las técnicas utilizadas en este trabajo para aplicarse en diferentes áreas dentro de las industrias tales como en los datos textuales de mantenimiento. En trabajos futuros se planea completar el proceso de recuperación de información al identificar los tops N mejores resultados para ser presentados al usuario.

Además, se pretende experimentar con técnicas de aprendizaje profundo como seq2sec y BERT. Finalmente se pretende también crear una aplicación amigable con el usuario que muestre las respuestas de las consultas, con el propósito de suministrar al personal de mantenimiento una herramienta más visual y entendible para el personal de esa área.

Agradecimientos. Durante la realización de este trabajo, Esaú Villatoro-Tello fue apoyado parcialmente por Idiap Research Institue, la UAM-Cuajimalpa, y el SNI-CONACyT México. Samuel González-López y Jesús Miguel Garcia-Gorrostieta fueron apoyados parcialmente por el SNI-CONACyT México.

Referencias

1. Ur-Rahman, N., Harding, J. A.: Textual data mining for industrial knowledge management and text classification: A business oriented approach. Expert Systems with Applications, vol. 39, NO. 5, pp. 4729–4739 (2012) doi: 10.1016/j.eswa.2011.09.124
2. Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I. H., Trigg, L.: Weka-a machine learning workbench for data mining. In Data mining and knowledge discovery handbook Springer, Boston, pp. 1269–1277 (2009) doi: 10.1007/978-0-387-09823-4_66
3. Tolosa, G. H., Bordignon, F. R.: Introducción a la recuperación de información (2008)
4. Chakraborty, G., Krishna, M.: Analysis of unstructured data: Applications of text analytics and sentiment mining. In SAS global forum, pp. 1288–2014 (2014)
5. Claassen, M., Grill, P.: Aprendizaje semisupervisado de rasgos de temporalidad en el léxico del español (2017)
6. Al-Abassi, A., Karimipour, H., HaddadPajouh, H., Dehghantanha, A., Parizi, R. M.: Industrial big data analytics: Challenges and opportunities. In Handbook of Big Data Privacy, Springer, Cham. pp. 37–61 (2020) doi: 10.1007/978-3-030-38557-6_3
7. Zhang, D., Xu, B., Wood, J.: Predict failures in production lines: A two-stage approach with clustering and supervised learning. In: Proceedings of IEEE International Conference on Big Data (Big Data), Washington, DC, pp. 2070–2074 (2016) doi: 10.1109/Big Data.2016.7840832
8. Abu-Samah, A., Shahzad, M. K., Zamai, E., Said A.: Failure prediction methodology for improved proactive maintenance using Bayesian approach. IFAC-PapersOnLine, vol. 48, no. 21, pp. 844–851 (2015) doi: 10.1016/j.ifacol.2015.09.632
9. Brito, J. H., Pereira, J. M., da Silva, A. F., Angélico, M. J., Abreu, A., Teixeira, S.: Machine learning for prediction of business company failure in hospitality sector. In Advances in Tourism, Technology and Smart Systems Springer, Singapore, pp. 307–317 (2020) doi: 10.1007/978-981-15-2024-2_28
10. Cadavid, J. P. U., Lamouri, S., Grabot, B., Pellerin, R., Fortin, A.: Machine learning applied in production planning and control: A state-of-the-art in the era of industry 4.0. Journal of Intelligent Manufacturing, vol. 38, pp. 1531–1558 (2020) doi: 10.1007/s10845-019-01531- 7

ISSN 1870-4069

Extracción de signos vitales, medidas antropométricas y fechas en expedientes médicos

Rodrigo Diaz-Moreno, Helena Gómez-Adorno,
Alejandro Martínez-Torres

Universidad Nacional Autónoma de México,
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
México

{rodrigo.diaz, helena.gomez}@iimas.unam.mx,
alejandromartinezt@ciencias.unam.mx

Resumen. La extracción de información de notas médicas es una tarea importante en el área de procesamiento de lenguaje natural. En este artículo presentamos métodos de extracción de información basado en reglas para la identificación de datos relevantes en las notas médicas como son: los signos vitales, los datos antropométricos y las fechas. Para la evaluación de los métodos desarrollados utilizamos un corpus de 198 notas médicas de pacientes de la Secretaría de Salud de la Ciudad de México. Los métodos desarrollados para la extracción de signos vitales y antropométricos logran extraer datos relevantes en la mayoría de las notas médicas del corpus. El método de extracción de fechas obtiene una precisión del 99 % en el corpus de evaluación.

Palabras clave: Notas médicas, extracción de información, signos vitales, datos antropométricos, identificación de fechas.

Extraction of Vital Signs, Anthropometric Measurements and Dates in Medical Records

Abstract. The extraction of information from medical notes is an important task in the area of natural language processing. In this article we present rule-based information extraction methods for the identification of relevant data in medical notes such as: vital signs, anthropometric data, and dates. For the evaluation of the methods developed, we used a corpus of 198 medical notes from patients from the Ministry of Health of Mexico City. The methods developed for the extraction of vital and anthropometric signs manage to extract relevant data in most of the medical notes of the corpus. The date extraction method obtains an accuracy of 99% in the evaluation corpus.

Keywords: Medical notes, extraction of information, vital signs, anthropometric data, identification of dates.

1. Introducción

La extracción de información de las notas médicas es importante debido a que estas poseen una gran cantidad de datos valiosos para análisis estadísticos [6, 2]. Esta tarea resulta complicada debido a que la forma en como está escrito el texto puede ser ambiguo y sobre todo que los autores tienen distintos estilos de escritura, por lo que los datos pueden variar en donde se encuentran.

En este trabajo no enfocaremos en identificar los signos vitales y los datos antropométricos. Los signos vitales son parámetros clínicos que reflejan el estado fisiológico del organismo humano, los cuales son:

- Temperatura: grado de calor conservado por el equilibrio entre el calor generado (termogésis) y el calor perdido (termólisis) por el organismo.
- Frecuencia respiratoria: el número de veces que una persona respira por minuto.
- Frecuencia cardiaca: el número de latidos por minuto.
- Presión arterial: es la fuerza que la sangre ejerce contra las paredes arteriales.
- Saturación de oxígeno: mide el porcentaje de hemoglobina encuadernada del oxígeno en la sangre.

Los datos antropométricos son mediciones técnicas sistematizadas que expresan, cuantitativamente, las dimensiones del cuerpo humano, las cuales son:

- Edad.
- Peso.
- Talla.
- Índice de Masa Corporal (IMC).

Por otro lado, la detección de fechas en notas médicas es de utilidad para ubicar en el tiempo los elementos detectados en la misma. Por ejemplo, los datos antropométricos y signos vitales pueden cambiar respecto al tiempo. De la misma manera, la identificación de fechas en las notas médicas servirá para ubicar en el tiempo otros datos del historial médico como: enfermedades, síntomas y medicamentos recetados.

Existen varias herramientas para la detección de fechas, sobre todo en el idioma inglés [7], sin embargo, al trabajar con notas médicas hechas en México (escritas en español), muchas de ellas necesitan ajustarse para detectar los formatos que fueron usados. Otro problema que se tuvo durante la extracción de fechas fue diferenciar entre un año y una cifra de 4 dígitos en un contexto médico.

Este trabajo está estructurado de la siguiente manera. En la Sección 2, describimos trabajos relacionados. En la Sección 3, presentamos una breve descripción del corpus de notas médicas utilizado para evaluar los métodos desarrollados. En la Sección 4, introducimos los métodos de extracción de signos vitales, datos antropométricos y fechas; así como las métricas de evaluación. Finalmente, en la Sección 5 presentamos los resultados obtenidos y en la Sección 6 presentamos las conclusiones y direcciones de trabajo futuro.

2. Trabajo relacionado

En la extracción de información en el dominio del procesamiento de lenguaje natural, se ha trabajado sobre todo en métodos de aprendizaje automático, como las maquinas de soporte vectorial (SVM) [2] el cual es un método que básicamente construye un clasificador binario utilizando muestras de entrenamiento intentando encontrar el hiperplano óptimo, que maximiza la distancia entre clases de la muestra de entrenamiento (llamados vectores de soporte) y de esta manera poder predecir si una palabra es la entidad de interés.

Así como redes neuronales convolucionales [4], que ocupan filtros para extraer características de los vectores de palabra y de esa manera poder realizar un reconocimiento de entidades nombradas. Con la ventaja de que procesan efectivamente grandes cantidades de datos de entrenamiento y ocupan menos parámetros para la tarea por lo que tienden a tener un menor sobreajuste. Consiguiendo resultados en el estado del arte.

A pesar del buen rendimiento que tienen estos sistemas tienen el inconveniente de que estos modelos no son interpretables, por lo que es muy difícil que se puedan mejorar con un humano con su conocimiento de ese dominio. El otro inconveniente es que se requiere una gran cantidad de recursos computacionales y datos etiquetados, es debido a eso que nuestro sistema considera expresiones regulares [1], los cuales son modelos basados en reglas, que tienen la ventaja de no requerir tantos datos y al ser un modelo más simple, es más fácil de ajustar para obtener un mejor rendimiento.

Adicionalmente, existen varias herramientas en la literatura que permiten la extracción de fechas en documentos escritos en lenguaje natural. Se han desarrollado bibliotecas, como date-detector basado en expresiones regulares, y otras herramientas en el lenguaje de programación Python, sin embargo fueron desarrolladas para textos en el idioma inglés [3].

También existen herramientas para el reconocimiento de entidades nombradas que pueden ser utilizadas para la extracción de fechas en textos escritos en español, como Stanza desarrollado por la universidad de Stanford [5].

El inconveniente de herramientas como Stanza es que no reconocen fechas en una variedad de formatos, muy probablemente debido a que estos formatos no se encontraban entre los datos con los que se realizó el entrenamiento de la herramienta. Es por esto que se desarrolló el presente trabajo basado en expresiones regulares que captura una amplia variedad de fechas en distintos formatos usados en el idioma español.

3. Corpus

Para el presente trabajo, SEDESA nos proporcionó un corpus de 98 expedientes médicos electrónicos de pacientes diagnosticados con el nuevo coronavirus SARS-CoV-2 (COVID). Por otro lado, se nos proporcionaron 100 expedientes médicos adicionales de pacientes con distintas enfermedades, en el mismo formato al de los expedientes médicos de pacientes con SARS-CoV-2. Dichos datos fueron proporcionados en un formato XML el cual venía organizado por secciones de las cuales se describen a continuación:

Table 1. Corpus de notas médicas COVID vs No COVID.

Tipo de Nota Médica	Número de Notas
COVID	98
No COVID	100

– Nombre y apellidos del paciente.
– Edad del paciente.
– Sexo del paciente.
– Estado y alcaldía.
– Fecha de ingreso.
– Fecha alta.
– Fecha hora registro nota.
– Nota médica (XML).
  Signos vitales: contiene el resumen de los signos vitales del paciente.
  Objetivo: contiene la descripción del estado actual del paciente y motivo de la consulta o revisión hospitalaria.
  Análisis: contiene la descripción del hallazgo del médico.
  Diagnóstico: describe el diagnóstico de la enfermedad del paciente.
  Plan de manejo: describe el tratamiento recetado al paciente, tanto de medicamentos como dieta, estudios necesarios, etc.

Es importante destacar que el objeto de estudio de este trabajo es el análisis de la nota médica, por lo tanto, cada sección del XML de la nota médica fue extraído para formar un solo documento por paciente. La tabla 1 muestra la cantidad de notas médicas existentes en el corpus por tipo de pacientes (COVID y No COVID).

Inicialmente el texto de las notas médicas no contenía ningún tipo de etiquetado, la única etiqueta que se tenía son las relacionadas con el paciente y el diagnóstico. Con la colaboración de tres expertos de SEDESA, se etiquetó de manera manual cada nota médica del corpus de pacientes COVID mediante la interfaz de una plataforma web de anotación de datos Dataturks[1]. A continuación se enuncian las características etiquetadas:

1. Síntomas, se identifican las palabras que contienen referencia a síntomas presentados por el paciente.
2. Comorbilidades, se identifican las palabras que hacen referencia a enfermedades previas del paciente.
3. Medicamentos, se identifican los medicamentos recetados al paciente.
4. Medicamentos previos, se identifican los medicamentos de base que el paciente está tomando actualmente .
5. Dosis, se identifica la dosis de los medicamentos (recetados y previos).

[1] https://docs.dataturks.com/

Fig. 1. Ejemplo de una nota médica etiquetada con características que se muestran en la figura.

6. Medidas (alternativas), identifica tratamientos alternativos como ozonoterapia, dieta especial, etc.
7. Signos vitales, se identifican los signos vitales como frecuencia respiratoria (FR), frecuencia. cardiaca (FC), saturación de oxígeno (SATO2), tensión arterial sistólica (TS) y diastólica (TD) y temperatura.
8. Datos antropométricos, se marcan el peso y la altura del paciente.

La Figura 1 muestra el ejemplo de una nota médica etiquetada con algunas de las características descritas previamente. Es importante destacar que no todos los expediente contaban con todas las características.

4. Metodología

Para resolver la extracción de signos vitales, datos antropométricos y fechas, se desarrolla un sistema basado en expresiones regulares implementado en el lenguaje de programación python, en donde en cada signo vital, en cada dato antropométrico y en cada formato específico de fecha se busca una expresión regular que contemple todas sus posibles variantes.

4.1. Pre-procesamiento

Los datos de las notas clínicas vienen en un formato XML, el cual tiene algunas inconsistencias en las etiquetas, por lo que se tuvo que corregir una vez identificando la estructura del XML y de esta manera evitar en lo posible pérdidas de información.

Nota Médica <paciente> 7000000000486834 06/09/1962 515351 <paciente> Mujer <doctor> HOSPITAL ABC <doctor> Signos Vitales 21/07/2020 06:52: Temperatura: 36.4 / Frecuencia cardiaca - ADL: 82.0 / Frecuencia respiratoria - ADL: 20.0 / SaO2: 93.0 / Otras constantes de hoy:Tensión Arterial Sistólica - ADL: 125.0 / Tensión Arterial Diastólica - ADL: 80.0 / Tensión Arterial Media - ADL: 95.0 / Síntomas Se trata de <paciente> de 57 años de edad, con Obesidad grado I, sin otros antecedentes de importancia para la enfermedad actual. Niega dolor torácico, Negó sintomatología urinaria digestiva. La paciente niega la presencia de disnea. Objetivo Mujer de edad aparente igual la cronológica, orientada en tiempo, persona, lugar circunstancia, alerta. Coloración normal de mucosas. Estado de hidratación adecuado, con aporte de oxígeno suplementario por puntas nasales 0.5lpm . Saturando 96%.

Fig. 2. Ejemplo de nota médica preprocesada.

Una vez hecho esto se procedió a estructurar el XML a un formato de renglones y columnas para la facilidad de la extracción de datos antropométricos y signos vitales.

Posteriormente, se hizo una limpieza de los datos, ya que estos tenían errores de codificación por lo que acentos o caracteres especiales en las notas médicas venían representados de manera distinta por lo que generaban ruido en el texto. Se extrajeron los datos de la creación de la nota médica para el uso posterior de detección de fechas y el resto de la información de cada nota se agrupo en un solo texto.

Finalmente, el texto de la nota médica fue anonimizado para el etiquetado mencionado previamente en la sección Corpus, el cual se usa para evaluar la extracción de signos vitales y medidas antropométricas. En la Figura 2 se muestra un ejemplo del texto que con el que se trabaja para la extracción de signos vitales, medidas antropométricas y fechas.

4.2. Signos vitales

Temperatura: Se extrae la temperatura de la nota médica a partir de una expresión regular que busca una palabra que empiece por 'temp' seguida de hasta 7 caracteres que puede seguirle de la palabra 'axilar', o la palabra fiebre, seguida de un número que contempla decimales de hasta dos cifras. Quedando la expresión regular siguiente:

”(?:temp\w{,7}(?: axilar)?|fiebre \w{,2}) (\d{,2}(?:\.\d{,2})?)”.

Frecuencia cardiaca: Para la frecuencia cardiaca se ocupa una expresión regular que busca el patrón: Una palabra que empiece por 'cardiaca' seguida de hasta 7 caracteres o la palabra 'fc', seguida de un número que contempla decimales de hasta dos cifras. Quedando la expresión regular siguiente:

”(?:cardiaca.{,6}|fc) (\d{,3}(?:\.\d{,2})?)”.

Frecuencia respiratoria: Se extrae la frecuencia respiratoria de la nota médica a partir de una expresión regular que busca una palabra que empiece por 'respiratoria' seguida de hasta 6 caracteres o la palabra 'fr', seguida de un número que contempla decimales de hasta dos cifras. Quedando la expresión regular siguiente:

”(?:fr|respiratoria.{,6}) (\d{,3}(?:\.\d{,2})?)”.

Saturación de oxígeno: Para el caso de la extracción de la saturación de oxígeno se busca una palabra que empiece por 'sat' seguida de hasta 6 caracteres o la palabra 'sao2', seguida de un número que contempla decimales de hasta dos cifras. Este es el único caso que se presenta en las notas médicas en donde hay más de una ocurrencia por nota, por lo que se consideran todas las coincidencias y se guarda en una lista. Quedando la expresión regular siguiente:

"(?:sao2|sat\w{0,7}) (\d{,3}(?:\.\d{,2})?)".

Presión sistólica: Se extrae la presión sistólica de la nota médica a partir de una expresión regular que busca el patrón: Una palabra que empiece por 'sist' seguida de hasta 7 caracteres o la palabra 'ta', seguida de un número que contempla decimales de hasta dos cifras. Quedando la expresión regular siguiente:

"(?:sist.{,7} .{,6}|ta) (\d{,3}(?:\.\d{,2})?)".

Presión diastólica: Para la extracción de la presión diastólica se busca el patrón: Una palabra que empiece por 'diast' seguida de hasta 7 caracteres, seguida de un número o una palabra que empiece por 'ta' seguida de un número y una diagonal seguida de otro número para quedarnos con este segundo número que corresponderá a la presión diastólica. Quedando la expresión regular siguiente:

"diast.{,7} .{,6} (\d{,3}(?:\.\d{,2})?)|ta \d{,3}(?:\d{,2})?/(\d{,3}(?:\.\d{,2})?)".

4.3. Datos antropométricos

Peso: Se extrae el peso de la nota médica a partir de una expresión regular que busca el patrón: Un número que contempla decimales de hasta dos cifras seguidas de la palabra 'kg', o una palabra que empiece por 'kg' con o sin paréntesis seguida de un número que contempla decimales con hasta dos cifras. Quedando la expresión regular siguiente:

"(\d{,3}(?:\.\d{,2})?) kg|\(?kg\)? (\d{,3}(?:\.\d{,2})?)".

Talla: Para la talla se busca a partir de una expresión regular el patrón: Un número que contempla decimales de hasta dos cifras seguidas de la palabra 'cm' o una palabra que empiece por 'cm' con o sin paréntesis seguida de un número que contempla decimales con hasta dos cifras. Cuidando que cuando haya una ocurrencia que sea menor a 2, esta se multiplique por 100 debido a los casos escritos en metros, dejando todo en centímetros. Quedando la expresión regular siguiente:

"(\d{,3}(?:\.\d{,2})?) cm|\(?cm\)? (\d{,3}(?:\.\d{,2})?)".

IMC: Para el IMC se extrae con una expresión regular el patrón: Una palabra que empiece por 'imc' seguida de un número que contempla decimales de hasta 2 dígitos. Para las notas médicas donde no haya una ocurrencia y se tenga la información tanto de la talla como del peso, se calcula manualmente el índice de masa corporal. Siendo la fórmula:

$$IMC = \frac{Peso}{Estatura^2}.$$

Quedando la expresión regular siguiente:

"imc (\d{,3}(?:\.\d{,2})?)".

4.4. Fechas

Primero, para obtener un conjunto de fechas como objetivo a extraer de las notas médicas, se extrajo todas las posibles fechas detectadas con herramientas previamente existentes como date-detector y Stanza, además de expresiones regulares que atrapan toda expresión que contienen el nombre de un mes.

Una vez obtenido el conjunto de posibles fechas se descartaron manualmente todas las que no lo eran. Teniendo un total de 428 fechas que se deseaban detectar. Posteriormente se agruparon las fechas respecto a los diferentes formatos que presentan.

Los formatos que se detectaron fueron:

- DD/MM/AAAA (con variaciones de longitud y separadores).
- Mes Día Año (ej. Septiembre 24, 2020).
- Día Mes Año (ej. 24 de septiembre del 2020).
- Mes Año (ej. septiembre 2020).
- Año (ej. 2020).
- Referencias al año (ej. el año pasado).

Definiciones y separadores: Dado los formatos mencionados, hay espacio para variaciones en la escritura de los días, mese y años. Un día, al igual que un mes, puede variar entre uno y dos dígitos. Un año entre 2 y 4 dígitos. Y un mes, además de ser un número, puede ser una palabra, una palabra con errores ortográficos o una palabra abreviada.

De igual forma estos datos son distinguidos unos de otros por separadores, conformados por una variedad de símbolos y palabras, espacios o signos de puntuación. Las expresiones regulares deben ser lo suficientemente robustas para cubrir todos estos casos.

Diferenciación entre un año y un número de 4 dígitos: Obtener un número de 4 dígitos no es suficiente para juzgar si el número en cuestión es un año o no. Verificar si un número se encuentra en cierto rango no es suficiente, ya que se encontraron ejemplos, dentro de las notas médicas, donde un mismo número (2009) es usado como una cantidad y como un año.

Por ello se identificaron palabras, expresiones o signos de puntuación que preceden a la mención de un año. Se encontró en el contexto de notas médicas que seguido de un signo de puntuación se puede presentar un año, a diferencia de una cantidad. También se encontró que las palabras que anteceden a un año, pero no una cantidad son: en, el, del, de o año.

Referencias al año: Dentro del corpus presentado se encontraron fechas que no presentaban el año, ya que se asumía que era el 2020, año de escritura de la nota. A su vez, se encontraron referencias informales al año, tales como "este año" o "el año pasado". Para poder obtener la fecha de referencias informales se capturaron las expresiones usadas y se relacionaron con la fecha de escritura de la nota médica.

Extracción de fechas y sus componentes: Se crean las expresiones regulares para definir los distintos componentes de una fecha: día, mes dígito, mes palabra, año, separadores símbolos, separadores palabras, expresión informal para año y antecedentes al año. Posteriormente, se crean las expresiones regulares que juntan los componentes en los distintos formatos.

Al capturar la fecha, también se obtienen los datos del día, el mes y el año. En caso que la fecha no cuente con el dato del día, se asume que ocurrió el primero del mes. Si la fecha no cuenta con mes, se asume el primer mes del año, enero. Y dado el caso en que no se cuente con año, se usa el año de la creación de la nota médica como se mencionó anteriormente. Esta información se guarda para un futuro uso y también se crea una fecha en un formato estándar.

5. Resultados

5.1. Evaluación de los métodos

En la Tabla 2 presentamos un resumen de la extracción en notas médicas de pacientes COVID y no COVID, siendo la primera columna la entidad que extrajimos, la segunda y tercera columna su correspondiente número de notas médicas en donde hubo una ocurrencia para pacientes COVID y no COVID. Es importante destacar que las notas médicas en las que no hubo una ocurrencia de una entidad es debido a que esta no contaba con la entidad a extraer.

Se puede observar en la tercera columna que la cantidad de datos faltantes es mucho mayor a la de pacientes COVID, esto debido a que las notas son más variadas y no todas cuentan con todas las entidades. Los casos más extremos son la presión sistólica y diastólica, en donde no hubo una sola ocurrencia en estas notas.

En la Tabla 3 se presenta el puntaje de la extracción de los signos vitales, obtenido a partir de un etiquetado manual hecha por expertos, notando que todas las entidades tienen un buen puntaje.

Siendo solo el caso de la saturación de oxígeno el más bajo, esto debido a que en las notas médicas se hace una mención de que se tiene que monitorear que la saturación de oxígeno no tiene que bajar de un valor y en el etiquetado no consideran ese valor debido a que no es el que presenta el paciente. No pudiéndose evaluar los datos antropométricos debido a que no se contaba con datos etiquetado para ello.

Table 2. Resumen de la extracción notas médicas.

Dato a extraer	Datos pacientes COVID	Datos pacientes no COVID
Temperatura	96	54
F. Cardíaca	95	52
F. Respiratoria	95	50
Saturación de Oxígeno	91	53
P. Sistólica	81	0
P. Diastólica	81	0
Peso	20	64
Talla	13	54
IMC	14	54

Table 3. Puntaje de la extracción de los signos vitales.

Entidad	Precisión	Recall
Temperatura	1.0	1.0
F. Cardíaca	1.0	1.0
F. Respiratoria	1.0	1.0
Saturación de Oxígeno	0.9565	1.0
P. Sistólica	1.0	1.0
P. Diastólica	1.0	1.0

5.2. Signos vitales

La Figura 3 presenta los datos que se extrajeron de los signos vitales en forma de diagrama de caja, esto se hace para poder visualizar que los datos estas dentro del rango que es de esperarse y que se encuentran en la misma escala.

Notando que la temperatura tuvo un mínimo de 36°C y máximo de 37.75°C, lo cual se encuentra dentro del rango de la temperatura corporal que va de 36 a 40°C.

De igual forma se observa que la frecuencia cardiaca tuvo un mínimo de 28 y un máximo de 143 latidos por minuto.

Para la frecuencia respiratoria se observa que su rango va de 14 a un máximo de 40 latidos por minuto, notandose que hay una gran cantidad de datos atípicos debido a que las notas médicas de pacientes no COVID son en su mayoría niños, los cuales tienen una frecuencia respiratoria mayor.

Continuamos notando que la saturación de oxígeno se encuentra dentro del rango que es de esperarse, siendo el mínimo 83% y el máximo de 100%. Para la presión arterial se observa que tanto la presión sistólica como la diastólica se encuentran dentro del rango que es de esperarse siendo el mínimo de 81 y 50 mm Hg y máximo de 142 y 91 mm Hg respectivamente.

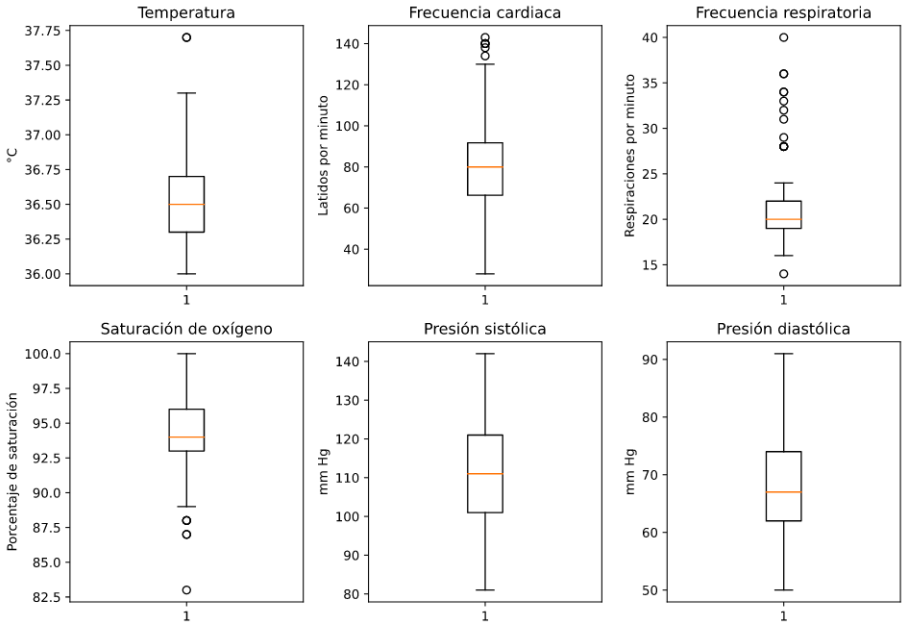


Fig. 3. Signos vitales.

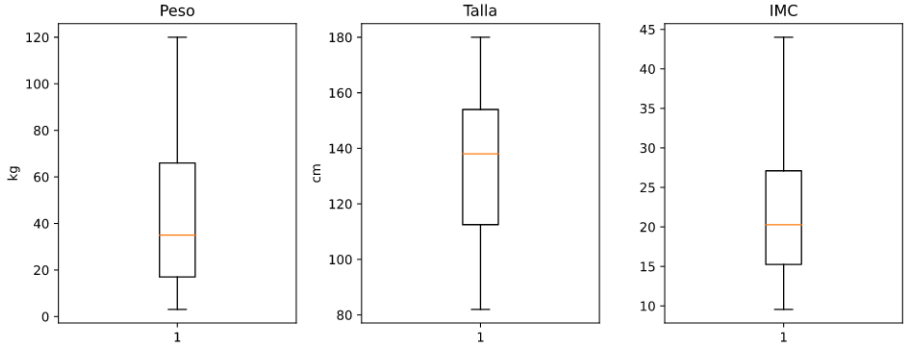


Fig. 4. Datos antropométricos.

5.3. Datos antropométricos

En la Figura 4, se hace un diagrama de caja para los datos antropométricos extraídos, de igual manera para visualizar que los datos extraídos estén en el rango que es de esperarse, observando que el peso tiene un mínimo de 3 kg y máximo de 120 kg, siendo importante mencionar que el mínimo se debe a que las notas médicas de pacientes no COVID son en su mayoría niños.

De igual manera para la talla se observa un mínimo de 82 y un máximo de 180 cm. Por ultimo el IMC tuvo un mínimo de 9.56 y un máximo de 44.

Table 4. Ejemplo de diferentes formatos de fechas extraídos.

Ejemplos de fechas extraídas
17.07.2020
01/03/1976
19.07.20
18 de junio de este año
19 de julio de 2020
abril 2019
julio de 2020
en 1994
: 2009

5.4. Fechas

Con el método propuesto se lograron extraer diferentes tipos de formatos de fechas, en la Tabla 4 se pueden apreciar algunos ejemplos de fechas extraídas.

Las fechas detectadas por nuestro método cubren por completo el conjunto objetivo de 428 fechas, además detecta otras fechas que no estaban contempladas.

Logramos extraer un total de 852 posibles fechas, de las cuales solo hubo 7 falsos positivos, es decir 7 elementos que no eran fechas. Esto se traduce a una precisión del 99.17% y un puntaje F1 de 99.49%.

6. Conclusiones

Los signos vitales y datos antropométricos tienen ciertos patrones en las notas médicas que pueden ser encontrados con una correcta exploración y teniendo la suficiente variedad de notas médicas, por lo que nuestro sistema basado en reglas funciona adecuadamente. Pudiendo extraer los datos correspondientes y validar que efectivamente están correctos.

Este sistema al estructurar la información de los signos vitales y datos antropométricos de las notas médicas, puede ser ocupado de base para una posterior explotación de la información encontrada.

Para la correcta extracción de fechas es necesario detectar todo posible formato utilizado para denotar una fecha.

La variedad de fechas presentadas y su clasificación posterior fue lo que hizo posible la creación de expresiones regulares capaces de realizar la tarea.

Como trabajo futuro se puede probar este sistema con un etiquetado que permita automatizar los resultados de los signos vitales y datos antropométricos, así como con nuevas notas médicas para volver más robusta las expresiones regulares.

En un futuro, el trabajo realizado puede ser utilizado para definir el alcance de cada fecha en la nota médica. De tal forma que se pueda ubicar en el tiempo otros datos tales como signos vitales y datos antropométricos.

References

1. Aguirre-Ojea, F., Manzotti, M., Díaz-Maffini, M.: Extracción automática de signos vitales en las evoluciones. In: Proceedings of IX Reunión Red Latinoamericana y del Caribe para el Fortalecimiento de los Sistemas de Información de Salud, pp. 1 (2019)
2. Ananiadou, S., Kell, D. B., Tsujii, J. I.: Text mining and its potential applications in systems biology. Trends in Biotechnology, vol. 24, no. 12, pp. 571–579 (2006) doi: 10.1016/j.tibtech.2006.10.002
3. Bird, S., Klein, E., Loper, E.: Natural language processing with Python: Analyzing text with the natural language toolkit (2009)
4. Gavrilov, D., Gusev, A., Korsakov, I., Novitsky, R., Serova, L.: Feature extraction method from electronic health records in Russia. In: Proceedings of Conference of Open Innovations Association, FRUCT, pp. 497–500. No. 26 (2020)
5. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C. D.: Stanza: A Python natural language processing toolkit for many human languages (2020) doi: 10.48550/arXiv.2003.07082
6. Wang, Y., Wang, L., Rastegar-Mojarad, M., Moon, S., Shen, F., Afzal, N., Liu, S., Zeng, Y., Mehrabi, S., Sohn, S., Liu, H.: Clinical information extraction applications: A literature review. Journal of biomedical informatics, vol. 77, pp. 34–49 (2018) doi: 10.1016/j.jbi.2017.11.011
7. Zhou, G., Su, J.: Named entity recognition using an HMM-based chunk tagger. In: Proceedings of 40th Annual Meeting of the Association for Computational Linguistics, pp. 473–480 (2002)

ISSN 1870-4069

Minería de secuencias de ADN para identificación de bacterias asociadas con Vaginosis Bacteriana

Freddy Garcia-Fuentes, Juana Canul-Reich,
Erick De-la-Cruz-Hernández, Betania Hernández-Ocaña,
Oscar Chávez-Bosquez

Universidad Juárez Autónoma de Tabasco,
División Académica de Ciencias y Tecnologías de la Información,
México.

ffreddy.mx@gmail.com, {juana.canul, erick.delacruz,
betania.hernandez, oscar.chavez}@ujat.mx

Resumen. El presente trabajo clasifica documentos con secuencias de ADN para identificar microorganismos presentes en la vaginosis bacteriana. Se aplica el método probabilístico de Latent Dirichlet Allocation (LDA) para llevar a cabo un análisis de secuencia sin realizar ninguna técnica de alineación de secuencias. El proceso consiste en fragmentar secuencias de ADN en subsecuencias cortas llamadas k-mer. Con la colección de k-mers contenidas en los documentos se crea el corpus de documentos y se importa en LDA para generar la matriz de términos y tópicos. De la matriz de términos se mide la similitud coseno de los temas resultantes con el gen codificante ARNr 16S de los microorganismos: *(Gardnerella vaginalis, Atopobium vaginae, Prevotella sp, Lactobacillus crispatus y Lactobacillus inners).* Si el resultado de similitud coseno es mayor al 40 % se etiquetan los temas desconocidos con el nombre de la bacteria de mayor puntuación. De la matriz de tópicos, se observan las probabilidades con una puntuación de 99 % y se toman los documentos con las secuencias completas de ADN para validar manualmente en el banco de secuencias RDP disponible en la web. Los resultados hallaron los microorganismos presentes en la vaginosis bacteriana del conjunto de secuencias de ADN mediante las técnicas de minería de texto.

Palabras clave: ADN, ARNr 16S, LDA, k-mer, vaginosis bacteriana.

DNA Sequence Mining for Identification of Bacteria Associated with Bacterial Vaginosis

Abstract. The present work classifies documents with DNA sequences to identify microorganisms present in bacterial vaginosis. The probabilistic method of Latent Dirichlet Allocation (LDA) is applied to conduct sequence analysis without performing any sequence alignment technique. The process consists of

fragmenting DNA sequences into short subsequences called *(k-mer)*. With the collection of k-mers contained in the documents, the document corpus is created and imported into LDA to generate the matrix of terms and topics. From the matrix of terms, the cosine similarity of the resulting subjects with the 16S rRNA coding gene of the microorganisms is measured: *(Gardnerella vaginalis, Atopobium vaginae, Prevotella sp, Lactobacillus crispatus and Lactobacillus inners)*. If the cosine similarity result is greater than 40%, the unknown subjects are labeled with the name of the bacteria with the highest score. From the matrix of topics, probabilities with a score of 99% are identified and the documents with the entire DNA sequences are taken for manual validation in the RDP sequence bank available on the web. Results show that microorganisms present in bacterial vaginosis were found in the set of DNA sequences using text mining techniques.

Keywords: DNA, 16S rRNA, LDA, k-mer, bacterial vaginosis.

1. Introducción

La capacidad computacional en los últimos años ha incrementado exponencialmente, lo que permite procesar enormes cantidades de información en poco tiempo. Áreas como la biología se han beneficiado de estos avances, lo que ha permitido generar una gran cantidad de datos de secuencias de ADN.

Estas secuencias son piezas fundamentales en el estudio de la filogenética, que es una de las principales áreas de investigación de la ciencia biológica. Hay muchas investigaciones en el área de las ciencias de la vida para las que la Informática propone emplear técnicas de inteligencia artificial, específicamente del dominio de la minería de datos y comparar datos de secuencias de ADN sin necesidad de emplear el alineamiento.

Otras bacterias se alojan en la vagina humana y actuan como la primera línea de defensa contra las infecciones vaginales [10] y es importante en la salud integral de la mujer [5].

La vaginosis bacteriana (VB) se diagnostica mayoritariamente en mujeres en edad fértil, hasta ahora no se conoce con certeza la causa de su aparición, aunque se cree que implica la pérdida de lactobacilos vaginales y la adquisición de comunidades bacterianas que incluyen muchas bacterias asociadas a la VB [16].

La VB es una causa común de vaginitis y aumenta el riesgo de enfermedades inflamatorias pélvicas, resultados adversos del embarazo, riesgo de infección por inmunodeficiencia humana y problemas de infertilidad [16].

La clasificación de las secuencias genómicas para su identificación ha desempeñado un papel muy importante en el campo de la medicina y en el análisis de la biodiversidad [24]. La de clasificación de las secuencias presentes en un meta-genoma se lleva a cabo de la siguiente manera:

1. Comparando las secuencias contra genomas conocidos,
2. Extrayendo las secuencias de genes ribosomales (16S y 18S) y comparándo contra una base de datos [2].

La bioinformática emplea técnicas de análisis basadas en el alineamiento de secuencias, que buscan la similitud comparando una secuencia desconocida con otra conocida para identificar si existe una relación entre ambas secuencias. Estas técnicas de alineamiento son muy fiables. Sin embargo involucran operaciones con matrices y éstas tienden a ser demasiado grandes, lo que ocasiona que los algoritmos de alineamiento tomen mucho tiempo para presentar resultados.

Otro descubrimiento importante es la descomposición de cadenas de ADN en subcadenas cortas llamadas k-mer. Estos avances han permitido explotar las técnicas de minería de datos reduciendo la complejidad computacional de los problemas del tipo NP *(tiempo polinomial no determinista)* [1] para el tratamiento de secuencias genómicas con métodos de minería de textos [13].

2. Trabajos relacionados

En el campo de la bioinformática, el alineamiento de las secuencias genómicas [34] es el método utilizado para identificar regiones similares en las secuencias [19]. Actualmente, existen aproximaciones en el alineamiento empleando algoritmos de programación dinámica que garantizan un alineamiento óptimo pero requiere mucho tiempo computacional [33, 1].

Los algoritmos para el alineamiento de múltiples secuencias se consideran problemas NP-duros [12]. Por otro lado, se han propuesto algoritmos que se basan en la teoría de que no es necesario el alineamiento [34].

Los algoritmos sin alineamiento son conocidos por el uso de métodos de comparación para cuantificar la similitud o disimilitud entre una o más secuencias biológicas en lugar del proceso de alineamiento [1].

La distancia genética puede considerarse como uno de los mejores criterios para comparar diferentes especies, teniendo en cuenta sus características [12]. Los modelos probabilísticos se aplican principalmente en el campo de la minería de texto, para organizar un corpus de documentos de acuerdo a un conjunto de temas que representa la ocurrencia de temas identificados a partir de esos documentos [8, 36].

El primer estudio computacional con ADN basado en las similitudes de la secuencia por comparación, fue a través de las distancias evolutivas de un conjunto de datos genómicos públicos [25].

Los modelos de temas se propusieron originalmente para el procesamiento de palabras, luego se aplicaron al procesamiento de imágenes y de audio [27, 32], así como al procesamiento de música [31].

Recientemente, algunos investigadores aplicaron el modelado de temas para el procesamiento de datos biológicos, como la extracción de relaciones proteína-proteína a partir de resúmenes científicos de la literatura de MEDLINE [35].

3. Materiales y métodos

3.1. Secuencias de ADN a analizar

Las secuencias de ADN se descargaron en formato fastq del banco de secuencias (ENA) [30].

Estas secuencias de ADN a analizar están compuestas de 155 pares de secuencias crudas para el análisis de vaginosis bacteriana. Las secuencias se tomaron de 48 mujeres embarazadas, de raza caucásica, con inicio espontáneo de parto prematuro con o sin RPM ($\leq 36^{6/7}$ semanas)(casos) y 107 mujeres embarazadas con inicio espontáneo de parto con o sin ruptura prematura (RPM) o cesáreas planificadas a término ($\geq 38^{0/7}$) (controles) [20].

Del banco de secuencias GenBank [6] se tomaron las secuencias válidas del gen ARNr 16S de la región vaginal V2 y V6: *Gardnerella vaginalis* (GenBank, NR118377), *Atopobium vaginae* (GenBank, AF325325), *Prevotella sp* (GenBank, KF007172), *Lactobacillus crispatus* (GenBank, AF243150) y *Lactobacillus inners* (GenBank, AY526083).

Para la revisión manual y comparar los documentos de las secuencias completas de ADN se usa la herramienta RDP [28], como la base de datos principal. RDP es una base de datos en linea que contiene herramientas para el análisis de la secuencia del gen 16S ARN ribosomal [15].

3.2. Gen ARNr 16S

El gen ARNr 16S es considerado una buena opción para la clasificación de las bacterias [26], es como un estándar para identificar microorganismos y se considera la diana universal para la identificación bacteriana a partir del ADN [3, 23].

En este estudio, se consideran las secuencias del gen ARNr 16S de ADN de cinco bacterias presentes en la vaginosis bacteriana y se obtienen del GenBank: *Gardnerella vaginalis, Atopobium vaginae, Prevotella sp, Lactobacillus inner y Lactobacillus crispatus*.

3.3. Modelado de temas (LDA)

El método LDA está basado en el modelado probabilístico [8] Asignación Latente de Dirichlet (LDA) [9] de aprendizaje no supervisado. Es un modelo estadístico aplicado inicialmente a documentos de texto con el objetivo de descubrir los temas de una colección de documentos denominado corpus. Es un modelo que obtiene información de los diferentes temas tratados con la distribución de probabilidad sobre el conjunto de las palabras.

La distribución de probabilidad [24] se expresa como:

$$P\left(w_i\right) = \sum_{j=1}^{T} P\left(w_i | z = z_j\right) P\left(z = z_j\right), \tag{1}$$

donde $P\left(w_i\right)$ es la probabilidad de que la palabra w_i este en un documento dado; $P\left(z = z_j\right)$ es la probabilidad de elegir una palabra del tema z_j para el documento actual; $P\left(w_i | z = z_j\right)$ es la probabilidad de muestreo de la palabra w_i, dado el tema zj y T es el numero de temas.

1 La distribución φ para cada tópico, representa la probabilidad de las ocurrencias de palabras en cada tópico dado, como el conjunto de:

$$\varphi \approx Dirichlet\left(\delta\right), \tag{2}$$

donde $\approx$ quiere decir "está distribuido como".

2 La proporción de Θ es la distribución del tópico para el documento d como el conjunto de:

$$\Theta \approx Dirichlet\,(\alpha)\,. \tag{3}$$

Los modelos de temas LDA, tienen la característica de seleccionar la cantidad de temas que se desean generar para el estudio de la investigación.

3.4. Muestreo de Gibbs

Gibbs es un algoritmo que genera una muestra aleatoria a partir de distribuciones de probabilidades de datos completos o incompletos [21]. Se trata de un algoritmo Metropolis-Hastings y es un método MCMC (Monte Carlo Markov Chain) y gracias a los avances computacionales en la actualidad se aplica en áreas como la biología [22].

La ecuación para encontrar la distribución de probabilidad de la asignación de una sola palabra w en un documento d de pertenecer al tema k esta dada por [18]:

$$p(Z_{d,n} = k|\overrightarrow{Z}_{-d,n}\,, \overrightarrow{W}, \alpha, \lambda) = \frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{^V k, W_{d,n} + ^\lambda W_{d,n}}{\sum_i V_{K,i} + \lambda_i}, \tag{4}$$

donde:

$n_{d,k}$: Veces que el documento d usa el tema k

$V_{k,W}$: Veces que el tema k usa la palabra dada w

αk: Parámetro de Dirichlet para la distribución de documentos a temas

λw: Parámetro de Dirichlet para la distribución de tema a palabra

El muestreo Gibbs se configura con valor α = 0.1, puesto que da mejores resultados [24].

3.5. Similitud coseno

Es una medida de similitud y describe el grado de semejanza o disimilitud de los objetos comparados. Su formula esta dada en (5):

$$cos(\theta) = \frac{A.B}{||\ A\ ||||\ B\ ||}. \tag{5}$$

Los objetos se consideran vectores de frecuencia de términos de los documentos y su índice de similitud se mide a partir de la multiplicación de dos vectores (A, B) dividida por el producto de sus longitudes. Este resultado representa el ángulo coseno de los dos vectores. Su valor cae en el rango [0, 1] donde: si el ángulo es cero, su similitud es uno, y cuanto mayor sea el ángulo menor será su similitud [11].

Del resultado LDA, se mide la similitud coseno de cada tema conformado por las subsecuencias k-mers para encontrar una relación entre el conjunto de temas y las secuencias ARNr 16S.

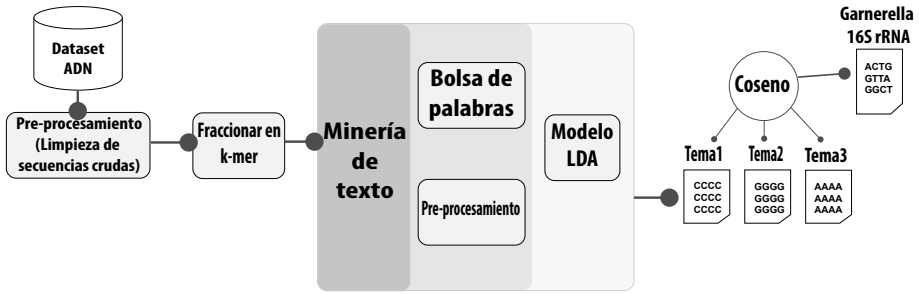


Fig. 1. Flujo de trabajo para el análisis de secuencias de ADN con técnicas de minería de texto.

4. Estudio experimental

En esta investigación se plantea estudiar secuencias de ADN sin métodos tradicionales de alineación para la identificación de vaginosis bacteriana; en su lugar se aplican técnicas de minería de texto. Cada una de las secuencias limpias de ADN representan los documentos y los k-mers las palabras. El desarrollo del trabajo experimental se ilustra en la Figura 1.

4.1. Preparación de las secuencias de ADN

Los procesos de secuenciación, a través de los cuales se generan las secuencias de ADN introducen errores de mala calidad en los datos secuenciados [7]. El conjunto de secuencias de ADN utilizado en este estudio descrito en la sección 3 son secuencias crudas que requirieron un pre procesamiento para identificar y eliminar los posibles errores de secuenciación.

Este pre procesamiento se realizó con el algoritmo DADA2 [7] en el lenguaje de programación estadístico R [20]. DADA2 es un paquete disponible en R y proporciona un conjunto de herramientas para medir la calidad de las secuencias duplicadas y de mala calidad DADA2.

El algoritmo DADA2 [7] recibe como entrada una o muchas muestras de secuencias crudas. Estas muestras están compuestas de dos lecturas separadas $forward$ y $reverse$ realizadas por el secuenciador. DADA2 realiza el pre procesamiento de las secuencias crudas y genera una salida compuesta de un conjunto de secuencias únicas.

La cantidad de secuencias únicas puede variar dependiendo de la configuración de los parámetros en DADA2.

Los valores de los parámetros en este estudio para la función DADA2 son: truncLen=c(260,240) se eliminan los extremos de las secuencias de baja calidad, trimRight = 5 número de nucleótidos que se eliminarán al final de cada lectura de secuencias después de truncLen, maxEE=c(2,3) elimina las lecturas de secuencias con los “errores esperados”, truncQ=2 para leer la primera instancia con el puntaje dado.

DADA2 fusiona las secuencias únicas y se obtienen secuencias completas sin ruido. Estas secuencias fusionadas se almacenan por separado en archivos de texto plano en formato txt. Cada una de las secuencias contienen las lecturas de nucleótidos (A, C, G, T).

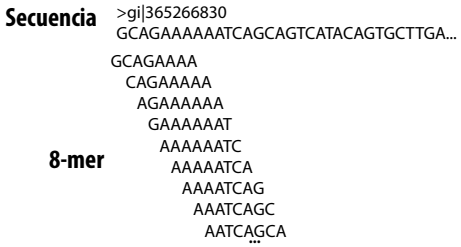


Fig. 2. K-mer de longitud k = 8 nucleotidos.

Cada una de las secuencias limpias representa un documento, por lo tanto el conjunto de secuencias almacenadas en archivos txt puede considerarse como el corpus de los documentos.

Un k-mer es una subsecuencia de ADN, un pequeño fragmento de secuencia definida por un tamaño k, donde k representa la cantidad de nucleótidos que conformara el k-mer. Ver Figura 2.

Las secuencias limpias de ADN obtenidas del pre procesamiento de nucleótidos descrita en esta sección, esta compuesta de una sola cadena de texto definida por los nucleótidos (A, C, G, T). Para extraer las palabras de las secuencias de ADN, se toma cada una de la secuencia almacenada en los documentos y se fragmenta en subcadenas de nucleótidos con la función substring() en R y cada secuencia fragmentada se almacena en un nuevo documento de texto plano txt.

Para este análisis, la longitud de la cadena k-mer es k = 8 nucleótidos. Para decidir el valor de k nucleótidos, se realizaron tres pruebas con k = 8, k = 9 y k = 10, dando mejor resultado k = 8 con mayor puntuación en la similitud del tema generado y la secuencia 16S. En la práctica, la longitud k-mer (k) puede establecerse con seguridad entre 8 y 10 para el gen ARNr 16S, como se describe en [4].

Con los nuevos documentos que contienen la colección de k-mers se crea el corpus de documentos.

Este mismo procedimiento de fragmentado se aplica a las secuencias ARNr 16S de la sección 3.2, utilizadas para medir la similitud coseno con los temas generado por LDA.

4.2. Latent Dirichlet Allocation (LDA)

Con LDA se procesa el corpus de documentos compuesto por la colección de k-mer y genera como resultado una matriz de términos y tópicos como se observa en la Figura 3.

Los términos son la distribución de los k-mers con las probabilidad de pertenecer a uno de los temas generados y los tópicos son los documentos marcados con la probabilidad de pertenecer a un tema.

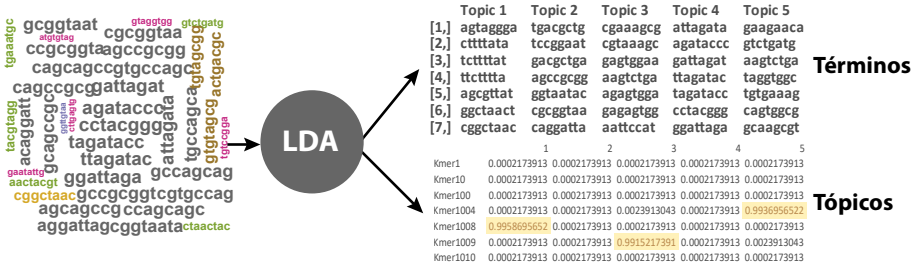


Fig. 3. Términos y Tópicos.

4.3. Interpretación de los temas

A partir del resultado LDA de la Figura 3 descrita en la sección 4.2, LDA solo genera los temas (tema1, tema2, tema3, ...) y no tiene la cualidad de identificar el significado de cada tema generado.

Para identificar y clasificar cada uno de los temas generados por LDA se toman los primeros k-mers con las probabilidades más altas de pertenecer al tema, cuanto más cerca este la probabilidad al valor 1 el k-mer pertenecerá al tópico asignado.

Para asociar los temas desconocidos, se les asigna una etiqueta de la bacteria midiendo la similitud coseno contra el conjunto de bacterias del gen ARNr 16S que conforman la vaginosis bacteriana descrito en la sección 3. Si los temas logran alcanzar una similitud coseno con una puntuación mayor a 40 %, se etiqueta el tema con el nombre del gen comparado, tal como se sugiere en [14]. Ver las tablas de resultados del apartado 5.

El valor del 40 % de la similitud coseno se tomo como umbral debido a que todas las secuencias que cumplieron con este valor como mínimo, se ingresaron al sitio RDP para realizar su búsqueda en la bases de datos de secuencias conocidas correspondiente al gen codificante ARNr 16S. En todos los casos estas secuencias dieron un resultado positivo, es decir la secuencia introducida coincide con la bacteria localizada en el gen ARNr 16S.

Por ejemplo al introducir una secuencia de la bacteria *Atopobium vaginae* se encontró que coincide con la bacteria *Atopobium vaginae* ya conocida en el gen ARNr 16S. Por otro lado, se realizó este mismo procedimiento con secuencias cuyos valores de similitud eran menores a 40 %, y en todos los casos el resultado obtenido de la búsqueda en el sitio RDP fue negativo.

De la matriz de tópicos, a los archivos que obtienen una probabilidad de 99 %, se toma la secuencia completa libre de impureza y se valida manualmente en el banco de secuencias ribosómica RDP 16S disponible en la web [28]. Así, se confirma si la similitud coseno del tema generado por LDA y el gen ARNr 16S etiquetan correctamente a la bacteria del conjunto de vaginosis bacteriana.

Si los resultados en RDP no confirman la existencia de la bacteria etiquetada con la similitud coseno, es un indicativo para cambiar la cantidad de temas a generar en LDA o también los parámetros de limpieza en DADA2.

Tabla 1. Muestra 1.

Temas	**Similitud**	**ARNr 16S**
Tema 1	0.41742100	Prevotella sp
Tema 2	0.51097878	Gardnerella vaginalis
Tema 3	0.19383758	Atopobium vaginae
Tema 5	0.41727181	Lactobacillus Crispatus
Tema 5	0.41524543	Lactobacillus Inner

5. Resultados

Con la finalidad de observar la eficiencia y ajustar los temas idóneos a generar en el algoritmo LDA. Se analizan las variaciones de los resultados de LDA en diferentes situaciones, aumentando la cantidad de documentos de secuencias en el corpus y los temas generados en LDA.

Se toma una secuencia de ADN de las 155 tomadas a las pacientes y posteriormente se realiza una limpieza con DADA2. Se generan 174 documentos de secuencias de ADN limpias de impurezas. Las 174 secuencias de ADN, se fragmentan por separado en secuencias cortas k-mer, generando 174 nuevos documentos que almacenan la colección de k-mer. En total se obtienen $6,913$-mer.

Se crea el corpus de documentos con los documentos k-mer y se generan 5 temas en LDA, la misma cantidad de microorganismo que se desean identificar de la vaginosis bacteriana.

En la Tabla 1 están registrados los valores obtenidos al medir la similitud coseno de los temas generados por LDA y las bacterias del gen ARNr 16S correspondientes a vaginosis bacteriana.

Con excepción del Tema 3 cuya similitud es inferior al 40 %, el resto de los temas superaron este porcentaje. Vea sección 4.3 para la interpretación de este umbral del 40 %.

El Tema 2 con una similitud coseno = 0.51097878 es un indicativo de la presencia del microorganismo *Gardnerella vaginalis*, mientras que el Tema 1 indica la presencia de *Prevotella sp* y en el Tema 5 están presente dos bacterias *Lactobacillus Crispatus y Lactobacillus Inner*.

En la siguiente ejecución se toman 25 pares de secuencia de las 155 pacientes y se obtiene en total 1,651 documentos de secuencias de ADN limpias con un corpus de documentos de 21,889-mers. Se generan 5 temas en LDA. El Tema 2 clasifica positivo para *Gardnerella vaginalis* con una similitud coseno = 0.52801140.

Sin embargo, se hallaron en el mismo tema, documentos que coinciden con el mismo umbral de probabilidad del 99 % y al comprobar en la bases de datos online RDP clasificó para *Gardnerella vaginalis* y otras bacterias tales como: *Actinobaculum massiliense*, *Mobiluncus*, *Corynebacterium* todas pertenecientes a la clase *Actinobacteria*. Con este hallazgo, se modifica la cantidad de temas de 5 a 10 y se ejecuta nuevamente el algoritmo LDA.

Los resultados en esta ejecución para 10 temas, el tema clasificado como *Gardnerella* continua con una similitud coseno = 0.52801140 y ahora comparte características similares a *Bifidobacterium* puesto que aun se tienen documentos clasificados con probabilidades del 99 % de pertenecer al mismo tema.

Tabla 2. 25 pares de secuencias con 15 temas.

Temas	Similitud	ARNr 16S
Tema 7	0.44190317	Prevotella sp
Tema 2	0.52801140	Gardnerella vaginalis
Tema 11	0.36626978	Atopobium vaginae
Tema 12	0.53305327	Lactobacillus Crispatus
Tema 13	0.53621329	Lactobacillus Inner

Tabla 3. 25 pares de secuencias con 20 temas.

Temas	Similitud	ARNr 16S
Tema 6	0.51253985	Atopobium vaginae

Tabla 4. Similitud con 155 muestras y 20 temas.

Temas	Similitud	ARNr 16S
Tema 4	0.45783722	Gardnerella vaginalis
Tema 8	0.52961422	Atopobium vaginae
Tema 10	0.54504403	Prevotella sp
Tema 14	0.45047193	Lactobacillus Crispatus
Tema 16	0.52807122	Lactobacillus Inner

Se incrementa la cantidad de temas a 15 y se ejecuta LDA. *Gardnerella vaginalis* continua con una similitud coseno = 0.52801140 y al revisar los documentos con las probabilidades de 99 % de pertenecer al Tema 2 como se muestra en la Tabla 2, se observo que los documentos marcados con esta probabilidad solo pertenecían a *Gardnerella vaginalis*.

Con 15 tópicos, la similitud coseno=0.36626978 para el Tema 11 de la Tabla 2 es muy baja para *Atopobium* y no se puede determinar si existe o no documentos de secuencias con presencia de la bacteria. Se ejecuta nuevamente LDA generando 20 temas y la similitud para *Atopobium vaginae* incrementa como se muestra en la Tabla 3.

Si la cantidad de temas generados por LDA son insuficientes para un corpus de documentos demasiado grande, los términos contenidos en los documentos se mezclan en un mismo tema cuando los términos comparten características similares.

Después de este análisis previo en los diferentes resultados del algoritmo LDA variando la cantidad de temas a generar y documentos en el corpus, se toman las 155 pares de secuencias de ADN de las muestras tomadas a las pacientes, del resultado de la preparación y limpieza de las secuencias se obtiene un total de 5,933 documentos de secuencias limpias y se generan 35,446-mers en total.

En esta ejecución se generan 20 temas, siendo un indicador razonable considerar [29]. En esta ejecución se atribuyen los mayores pesos probabilísticos a los documentos donde las palabras están principalmente correlacionadas y clasifican para los microorganismos presentes en la vaginosis bacteriana, estos resultados se muestran en la Tabla 4.

Con el aumento en la cantidad de temas, la distribución de los k-mers se organizan correctamente en los temas correspondientes. Al observar los resultados de la Tabla 4, la similitud coseno de 0.45783722 calculada entre el Tema 4 y el gen codificante ARNr 16S de las bacterias que componen la vaginosis bacteriana, este valor es suficiente tal

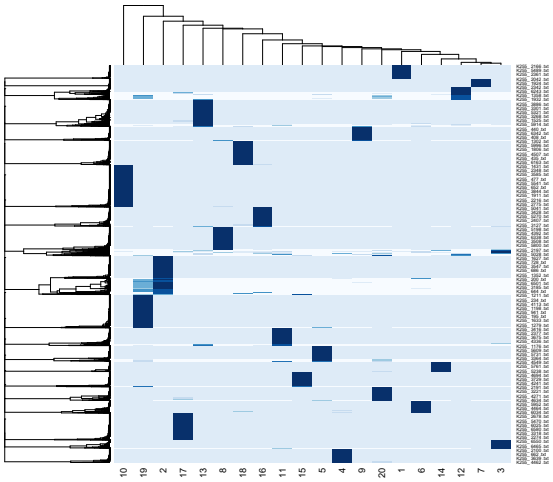


Fig. 4. LDA, extracción de los temas contenidos de la colección de k-mer. Cada rectángulo de color azul representa un tema.

como se describe en la sección 4.3 y en [14] para etiquetar el Tema 4 con la bacteria correspondiente a *Gardnerella vaginalis*. De la misma forma se etiquetan los Temas 8, 10, 14 y 16 con sus bacterias correspondiente.

En la Figura 4 se visualizan los 20 temas generados de los 155 pares de secuencias de ADN. El eje x representa la diversidad cualitativa de los microorganismos presentes en el microbioma vaginal.

En el eje y están los documentos con las probabilidades del 99 % y representan la diversidad cuantitativa en orden según la relevancia del tema. Cada bloque azul marino concentra los documentos con mayor probabilidad de estar asociados a un tema.

6. Conclusiones

Gracias a la información disponible en los bancos de secuencias del GenBank es posible descargar secuencias parciales específicas del gen ARNr 16S.

Los métodos probabilísticos son eficaces en el tratamiento de los datos. Aunque clasifican muy bien, es importante generar la cantidad óptima de temas que pueden estar presentes en un corpus de documentos.

El desarrollo de esta investigación como primer paso se tomó en cuenta la calidad de las secuencias de ADN, se buscó la calidad óptima en el filtrado y corte de error de las secuencias crudas con el pre procesamiento de secuencias de la herramienta DADA2.

El segundo paso consistió en transformar las secuencias limpias en secuencias cortas llamadas k-mer. Estas secuencias cortas tomaron el papel de palabras y se almacenaron en documentos para ser importados en los modelados de temas probabilísticos.

LDA es una técnica de minería de texto basada en modelos probabilísticos que identifica el contenido de cada documento y los clasifica en temas.

Un punto importante del algoritmo LDA, es que no identifica cada uno de los temas generados, se puede inferir de acuerdo al contenido de las palabra asociadas a cada tema.

La interpretación de los tópicos, puede llegar a ser compleja y gracias al gen ARNr 16S en nuestro análisis la interpretación de la composición de cada tópico se puedo identificar midiendo la distancia coseno. Estos nuevos métodos de comparación de secuencias sin alineación, mejoran el rendimiento computacional [1]. Los métodos tradicionales de alineamiento de secuencias, tienen la cualidad de procesar la información con mayor tiempo y recurso computacional.

Como línea de investigación futura, se busca complementar todos los microorganismos que conforman la vaginosis bacteriana, en conjunto con las líneas de investigación clínica para un pronóstico confiable y su diagnostico oportuno.

Referencias

1. Zielezinski, A., Vinga, S., Almeida, J., Karlowski, W. M.: Alignment-free sequence comparison: Benefits, applications, and tools. Genome biology, vol. 18, no. 1, pp. 1–17 (2017) doi: 10.1186/s13059-017-1319-7
2. CIAD: Ecología microbiana, secuenciación masiva y bioinformática (2016)
3. Moya, A. S.: Microbioma y secuenciación masiva. Revista Espsñola de Quimioterapia, vol. 30, no. 5, pp. 305–390 (2017)
4. Chan, C. X., Bernard, G., Poirion, O., Hogan, J. M., Ragan, M. A.: Inferring phylogenies of evolving sequences without multiple sequence alignment. Scientific Reports, vol. 4, no. 1, pp. 1–9 (2014) doi: 10.1038/srep06504
5. Lledó-Bosch, B.: Efecto del microbioma vaginal en la tasa de embarazo en pacientes que se someten a técnicas de reproducción asistida (2018)
6. Ouellette, B. F., Rapp, B. A., Wheeler, D. L., Benson, D. A., Boguski, M. S., Lipman, D. J., Ostell, J.: Genblack. Nucleic Acids Research, vol. 27, no. 1, pp. 12–17 (1999) doi: 10.1093/nar/27.1.12
7. Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J. A., Holmes, S. P.: Dada2: High-resolution sample inference from illumina amplicon data. Nature Methods, vol. 13, no. 7, pp. 581–583 (2016) doi: 10.1038/nmeth.3869
8. Blei, D. M.: Probabilistic topic models. Communications of the ACM, vol. 55, no. 4, pp. 77–84 (2012) doi: 10.1145/2133806.2133826
9. Blei, D. M., Ng, A. Y., Jordan, M. I.: Latent dirichlet allocation. Journal of Machine Learning Research, vol. 3, pp. 993–1022 (2003)
10. Financiera, D.: El microbioma vaginal y su relación con el comportamiento, la salud sexual y las enfermedades de transmisión sexual. Obstetrics & Gynecology, vol. 129, pp. 643–54 (2017) doi: 10.1097/AOG.0000000000001932
11. Liu, D., Chen, X., Peng, D.: Some cosine similarity measures and distance measures between q-rung orthopair fuzzy sets. International Journal of Intelligent Systems, vol. 34, no. 7, pp. 1572–1587 (2019) doi: 10.1002/int.22108
12. Gamage, G., Gimhana, N., Wickramarachchi, A., Mallawaarachchi, V., Perera, I.: Alignment-free whole genome comparison using k-mer forests. In: 19th International Conference on Advances in ICT for Emerging Regions, vol. 250, pp. 1–7 (2019) doi: 10.1109/ICTer48817.2019.9023714
13. Fan, H., Ives, A. R., Surget-Groba, Y., Cannon, C. H.: An assembly and alignment-free method of phylogeny reconstruction from next-generation sequencing data. BMC genomics, vol. 16, no. 1, pp. 1–18 (2015)

14. Choi, I., Ponsero, A. J., Bomhoff, M., Youens-Clark, K., Hartman, J. H., Hurwitz, B. L.: Libra: Scalable k-mer–based tool for massive all-vs-all metagenome comparisons. GigaScience, vol. 8, no. 2 (2019) doi: 10.1093/gigascience/giy165
15. Cole, J. R., Chai, B., Farris, R. J., Wang, Q., Kulam, S. A., McGarrell, D. M., Garrity, G. M., Tiedje, J. M.: The ribosomal database project (RDP-II): Sequences and tools for high-throughput rRNA analysis. Nucleic Acids Research, vol. 33, no. 1, pp. D294–D296 (2005) doi: 10.1093/nar/gki038
16. Marrazzo. J. M.: Interpreting the epidemiology and natural history of bacterial vaginosis: Are we still confused? Anaerobe, vol. 17, no. 4, pp. 186–190 (2011) doi: 10.1016/j.anaerobe.2011.03.016
17. Gamboa-Unsihuay, J. E.: Topic modeling en twitter: Determinación de la agenda política peruana en el periodo de enero a setiembre del 2018. Anales Científicos, vol. 80, no. 2, pp. 308–327 (2019)
18. Boyd-Graber, J., Blei, D., Zhu, X.: A topic model for word sense disambiguation. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1024–1033 (2007)
19. Nieto, J. J.: Comparación de secuencias (2005)
20. Hočevar, K., Maver, A., Vidmar-Šimic, M., Hodžić, A., Haslberger, A., Premru-Seršen, T., Peterlin, B.: Vaginal microbiome signature is associated with spontaneous preterm delivery. Frontiers in Medicine, vol. 6, pp. 201 (2019) doi: 10.3389/fmed.2019.00201
21. Angelone, L.: Análisis del muestreo Gibbs para detección de motivos en secuencias biológicas (2005)
22. Moreno-Arévalo, M. A.: Descripción e implementación del muestreador de Gibbs en versión bivariada (2016)
23. del-Rosario-Rodicio, M., del Carmen-Mendoza, M.: Identificación bacteriana mediante secuenciación del ARNr 16s: Fundamento, metodología y aplicaciones en microbiología clínica. Enfermedades infecciosas y microbiología clínica, vol. 22, no. 4, pp. 238–245 (2004) doi: 10.1016/S0213-005X(04)73073-6
24. la Rosa, M., Fiannaca, A., Rizzo, R., Urso, A.: Probabilistic topic modeling for the analysis and classification of genomic sequences. BMC Bioinformatics, vol. 16, no. 6, pp. 1–9 (2015) doi: 10.1186/1471-2105-16-S6-S2
25. Nei, M., Kumar, S.: Molecular evolution and phylogenetics. Oxford University Press (2000)
26. Pace, N. R., Stahl, D. A., Lane, D. J., Olsen, G. J.: The analysis of natural microbial populations by ribosomal RNA sequences. Advances in microbial ecology, vol. 9, pp. 1–55 (1986) doi: 10.1007/978-1-4757-0611-6_1
27. Elango, P. K., Jayaraman, K.: Clustering images using the latent dirichlet allocation model. University of Wisconsin, pp. 1–18 (2005)
28. Wang, Q., Garrity, G. M., Tiedje, J. M., Cole, J. R.: Naïve bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Applied and Environmental Microbiology, vol. 73, no. 16, pp. 5261–5267 (2007) doi: 10.1128/AEM.00062-07
29. Arun, R., Suresh, V., Veni Madhavan, C. E., Murthy, N.: On finding the natural number of topics with latent dirichlet allocation: Some observations. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, vol. 6118, pp. 391–402 (2010) doi: 10.1007/978-3-642-13657-3_43
30. Leinonen, R., Akhtar, R., Birney, E., Bower, L., Cerdeno-Tárraga, A., Cheng, Y., Cleland, I., Faruque, N., Goodgame, N., Gibson, R., Hoad, G., Jang, M., Pakseresht, N., Plaister, S., Radhakrishnan, R., Reddy, K., Sobhany, S., Hoopen, P. T., Vaughan, R., Zalunin, V., et al.: The european nucleotide archive. Nucleic Acids Research, vol. 39, pp. D28–D31 (2010) doi: 10.1093/nar/gkq967

31. Zhang, R., Cheng, Z., Guan, J., Zhou, S.: Exploiting topic modeling to boost metagenomic reads binning. BMC Bioinformatics, vol. 16, no. 5, pp. 1–10 (2015) doi: 10.1186/1471-2105-16-S5-S2
32. Kim, S., Narayanan, S., Sundaram, S.: Acoustic topic model for audio information retrieval. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 37–40 (2009) doi: 10.1109/ASPAA.2009.5346483
33. Shehab, S. A., Keshk, A., Mahgoub, H.: Fast dynamic algorithm for sequence alignment based on bioinformatics. International Journal of Computer Applications, vol. 37, no. 7, pp. 54–61 (2012)
34. Vinga, S., Almeida, J.: Alignment-free sequence comparison—a review. Bioinformatics, vol. 19, no. 4, pp. 513–523 (2003) doi: 10.1093/bioinformatics/btg005
35. Aso, T., Eguchi, K.: Predicting protein-protein relationships from literature using latent topics. Genome Informatics 2009: Genome Informatics Series, vol. 23, pp. 3–12 (2009) doi: 10.1142/9781848165632_0001
36. Griffiths, T. L., Steyvers, M.: Finding scientific topics. In: Proceedings of the National Academy of Sciences, vol. 101, pp. 5228–5235 (2004) doi: 10.1073/pnas.0307752101

ISSN 1870-4069

Extracción de síntomas en notas médicas escritas en español

Dalia Cruz-Aguirre, Helena Gómez-Adorno, Armando Rios-Lastiri

Universidad Nacional Autónoma de México,
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
México

dalia.cruz@comunidad.unam.mx,
{armando.rios, helena.gomez}@iimas.unam.mx

Resumen. La extracción de información de notas médicas es una tarea importante en el área de procesamiento de lenguaje natural. En este artículo presentamos métodos de extracción basados en aprendizaje automático y en diccionarios para la identificación de datos relevantes en las notas médicas como son los síntomas. Para la evaluación de los métodos desarrollados utilizamos un corpus de 98 notas médicas de pacientes de la Secretaría de Salud de la Ciudad de México. Nuestro método basado en redes neuronales recurrentes para la extracción de síntomas de las notas médicas obtuvo un F1-score de 96.02 %.

Palabras clave: Notas médicas, extracción de información, síntomas, redes neuronales recurrentes.

Extraction of Symptoms in Medical Notes Written in Spanish

Abstract. The extraction of information from medical notes is an important task in the area of natural language proceesing. In this article we present extraction methods based on machine learning and dictionaries for the indentification of relevant data in medical notes such as symptoms. For the evaluation of the methods developed, we used a corpus of 98 medical notes from patients from the Ministry of Health of Mexico City. Our method based on recurrent neural networks for extracting symptoms from medical notes obtained an F1-score of 96.02%.

Keywords: Medical notes, information extraction, symptoms, recurrent neural networks.

1. Introducción

La pandemia del SARS-CoV-2 del 2020 ha provocado la necesidad de emplear diferentes tratamientos para una enfermedad infecciosa de la cual no se tenían

registrados tratamientos efectivos, el uso de machine learning para el procesamiento de lenguaje natural en expedientes clínicos es una herramienta para la búsqueda de patrones en pacientes infectados por COVID.

El propósito principal de este trabajo es extraer los síntomas que presentan los pacientes e identificar cuales de ellos están negados. Los síntomas se deben extraer directamente de notas médicas proporcionadas por la Secretaría de Salud de la Ciudad de México (SEDESA).

El primer método que se probó para poder reconocer los síntomas fue mediante búsquedas en un diccionario de síntomas en español. Para el segundo método se ocupó el Named Entity Recognition (NER) mediante el uso de uso de redes neuronales recurrentes (RNN).

Este trabajo está estructurado de la siguiente manera. En la Sección 2, describimos el trabajo relacionado. En la Sección 3, presentamos una breve descripción del corpus de notas médicas utilizado para evaluar los métodos desarrollados.

En la sección 4, introducimos los métodos de extracción de síntomas; así como las métricas de evaluación. Finalmente, en la Sección 5 presentamos los resultados obtenidos y en la Sección 6 presentamos las conclusiones y direcciones de trabajo futuro.

2. Trabajo relacionado

Los métodos existentes para el reconocimiento y extracción de entidades nombradas están sustancialmente divididos en dos categorías:

1. Métodos basados en reglas.
2. Métodos basados en el aprendizaje automático.

Los métodos de extracción de entidades basados en reglas se basan principalmente en diccionarios. Expertos manualmente seleccionan características, palabras clave, etc. para construir plantillas de reglas, usan patrones de coincidencia y coincidencia de cadenas para completar la extracción de entidades.

Los métodos basados en el aprendizaje automático entrenan modelos de aprendizaje a través de un corpus etiquetado.

Los modelos de aprendizaje comunes incluyen: Modelo ocultos de Markov (HMM), que han sido utilizados para la extracción e identificación de reacciones adversas [3], el modelo de máxima entropía (ME), los campos condicionales aleatorios (CRF) [5] y el modelo bidireccional de memoria de corto y largo plazo (Bi-LSTM) que en los últimos años ha mostrado mejoras en comparación con los modelos ocultos de Markov y las máquinas vector soporte para la anotación de registros médicos en el idioma chino [7].

Además existen varios sistemas de extracción de entidades que se han convertido en herramientas ampliamente utilizadas en dominios biomédicos, como cTAKES [4], que es un sistema de procesamiento de lenguaje natural de código abierto que extrae información clínica de texto no estructurado de historias clínicas electrónicas y MetaMap. Tanto cTAKES como MetaMap manipulan el sistema de lenguaje médico unificado (UMLS) para estandarizar conceptos y extraer entidades médicas.

En el caso de la extracción de síntomas a partir de notas médicas, se ha presentado una síntesis de la literatura sobre el uso del procesamiento de lenguaje natural (PLN) para procesar o analizar la información de los síntomas documentada en las narrativas de texto libre [2] los enfoques de PLN incluían herramientas de PLN desarrolladas previamente, métodos de clasificación y procesamiento basado en reglas manual.

En el presente trabajo se propone un nuevo método de extracción de síntomas mediante redes neuronales recurrentes y se comparará con un método basado en diccionario con similitud en los trabajos previos pues utiliza como base el sistema de lenguaje médico unificado (UMLS).

3. Corpus

Para el presente trabajo, SEDESA nos proporcionó un corpus de 98 expedientes médicos electrónicos de pacientes diagnosticados con el nuevo coronavirus SARS-CoV-2 (COVID).

Por otro lado, se nos proporcionaron 100 expedientes médicos adicionales de pacientes con distintas enfermedades, en el mismo formato al de los expedientes médicos de pacientes con SARS-CoV-2. Dichos datos fueron proporcionados en un formato XML el cual venía organizado por secciones las cuales se describen a continuación:

- Nombre y apellidos del paciente.
- Edad del paciente.
- Sexo del paciente.
- Estado y alcaldía.
- Fecha de ingreso.
- Fecha alta.
- Fecha hora registro nota.
- Nota médica (XML).
- Signos vitales: contiene el resumen de los signos vitales del paciente.
- Objetivo: contiene la descripción del estado actual del paciente y motivo de la consulta o revisión hospitalaria.
- Análisis: contiene la descripción del hallazgo del médico.
- Diagnóstico: describe el diagnóstico de la enfermedad del paciente.
- Plan de manejo: describe el tratamiento recetado al paciente, tanto de medicamentos como dieta, estudios necesarios, etc.

Es importante destacar que el objeto de estudio de este trabajo es el análisis de la nota médica, por lo tanto, cada sección del XML de la nota médica fue extraído para formar un solo documento por paciente. La tabla 1 muestra la cantidad de notas médicas existentes en el corpus por tipo de pacientes (COVID y No COVID).

Inicialmente el texto de las notas médicas no contenía ningún tipo de etiquetado, la única etiqueta que se tenía son las relacionadas con el paciente y el diagnóstico.

Tabla 1. Corpus de notas médicas COVID vs No COVID.

Tipo de Nota Médica	Número de Notas
COVID	98
No COVID	100

Con la colaboración de tres expertos de SEDESA, se etiquetó de manera manual cada nota médica del corpus de pacientes COVID mediante la interfaz de una plataforma web de anotación de datos Dataturks[1]. A continuación se enuncian las características etiquetadas:

1. Síntomas, se identifican las palabras que contienen referencia a síntomas presentados por el paciente.
2. Comorbilidades, se identifican las palabras que hacen referencia a enfermedades previas del paciente.
3. Medicamentos, se identifican los medicamentos recetados al paciente.
4. Medicamentos previos, se identifican los medicamentos de base que el paciente está tomando actualmente.
5. Dosis, se identifica la dosis de los medicamentos (recetados y previos).
6. Medidas (alternativas), identifica tratamientos alternativos como ozonoterapia, dieta especial, etc.
7. Signos vitales, se identifican los signos vitales como frecuencia respiratoria (FR), frecuencia cardiaca (FC), saturación de oxígeno (SATO2), tensión arterial sistólica (TS) y diastólica (TD) y temperatura.
8. Datos antropométricos, se marcan el peso y la altura del paciente.

La Figura 1 muestra el ejemplo de una nota médica etiquetada con algunas de las características descritas previamente. Es importante destacar que no todos los expediente contaban con todas las características. Con respecto al etiquetado de síntomas, solo 57 notas médicas contaban con etiquetas de síntomas para el entrenamiento y evaluación del modelo desarrollado en este trabajo.

4. Metodología

Existen 3 problemas principales para la identificación de síntomas con base en historias médicas de pacientes COVID. El primero radica en extraer únicamente los síntomas relacionados con el paciente y no otro síntoma que aparezca en cualquier parte de la nota médica, para ello es necesario entender el contexto en que se encuentra el síntoma.

El segundo problema se debe a los errores gramaticales cometidos por los médicos a la hora de redactar la nota clínica. El tercer problema son los síntomas negados que son síntomas que no presenta el paciente, es importante identificar este tipo de síntomas ya que en varias ocasiones no aportan información relevante del estado del paciente.

[1] https://docs.dataturks.com/

Fig. 1. Ejemplo de una nota médica etiquetada con características que se muestran en la figura.

El marco de extracción de entidades que se presenta consta de tres etapas. Primero, analizamos los textos clínicos y aplicamos técnicas de procesamiento de lenguaje natural que se detallan en la sección de pre-procesamiento.

En la segunda etapa se realiza la identificación de síntomas basado en diccionario y mediante una red neuronal recurrente. La tercera etapa identifica los síntomas negados mediante el algoritmo de NegEx para dar como resultado los síntomas reales presentes en cada nota médica.

4.1. Pre-procesamiento

El primer problema que se presentó dentro de la lectura de las notas médicas fue la codificación, las notas no se encontraban en formato UTF-8, esto causó que se encontrarán muchas palabras con caracteres raros, esto se resolvió mediante la implementación de expresiones regulares para reemplazar los códigos que estaban mal traducidos.

Se buscaron diversas herramientas que fueran capaces de realizar un pipeline de pre-procesamiento, se encontró que la biblioteca Natural Language Toolkit (NLTK) no realizaba de manera correcta el etiquetado gramatical de las palabras ya que el modelo de etiquetado gramatical está entrenado en textos en Inglés, razón por la cual se descartó el uso de esta biblioteca.

Seguido de esto, spaCy mostró mejores resultados, al contar con un modelo mucho más extenso con relación al idioma español de México. Después de esta estructuración de las notas médicas se buscó anonimizar las notas con el fin de no mostrar datos sensibles del paciente en nuestro proceso de análisis de dichas notas y se pasaron todas las notas a minúsculas.

4.2. Identificación de síntomas basado en diccionario

Primero se realizó la extracción de síntomas mediante un método basado en diccionarios (MBD), posteriormente se propone un marco de extracción de entidades nombradas que utiliza métodos de aprendizaje automático para el entrenamiento y prueba del reconocimiento de entidades.

Se ha construido un diccionario con 377 síntomas a partir de la traducción de síntomas de la base de datos del Unified Medical Language System (UMLS) y una corrección manual posterior. Al tener síntomas compuestos por varias palabras, como en el caso de disnea de pequeños esfuerzos, se ha recurrido a los n-gramas para realizar la separación del texto por n cantidad de palabras y poder extraer estos síntomas compuestos por varias palabras.

Tomando en cuenta los errores humanos al escribir las notas médicas se propuso utilizar la distancia de *Levenshtein* [6], que toma la distancia entre dos cadenas de caracteres como el número mínimo de ediciones (es decir, inserciones, eliminaciones o sustituciones) de un solo carácter que son necesarias para para cambiar una cadena por otra. Se uso un umbral del 90 % de similitud para considerar que dos cadenas son muy similares.

Pseudo código para la extracción de síntomas basado en diccionario:

```
def Levenshtein (symptom,n_gram, threshold):
    extracted=list()
    if symptom in n_gram:
        Add sintoma to extracted
    else:
        for gram in n_gram:
            gram= concat(gram)
            if levenshtein_distance (sintoma,gram)>threshold:
                Add sintoma to extracted
    return concat(extracted)

def symptom_extraction(record,n_gram,threshold):
    words=nota.split()
    symptoms_extracted=list()
    for symptom in dictionary:
        extraction=Levenshtein (symptom,n_gram, threshold)
        if extraction not empty:
            symptoms_extracted.append(extraction)
    return(symptoms_extracted)

def main (records, threshold=90):
    extracted=list()
    dictionary = translated UMLS symptoms
    for record in records:
        bigram= Transform record into bigrams
        trigram=Transform record into trigrams
        Add extraccion_sintomas(record,n_gram,threshold) to extracted
        Add extraccion_sintomas(record,n_gram,threshold) to extracted
    return (extracted)
```

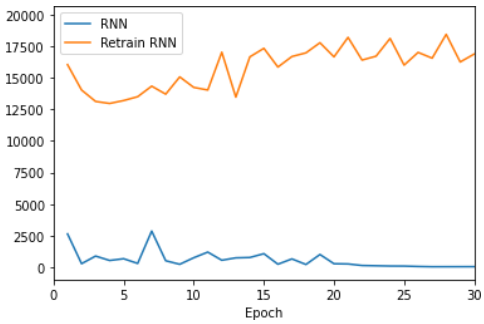


Fig. 2. Pérdida durante el entrenamiento.

4.3. Identificación de síntomas usando redes neuronales recurrentes

Una red neuronal recurrente (RNN) es un tipo de red neuronal que contiene bucles, lo que permite almacenar información dentro de la red. Las redes neuronales recurrentes utilizan su razonamiento de experiencias anteriores para informar los próximos eventos.

Las redes neuronales recurrentes se pueden considerar como una serie de redes conectadas entre sí. A menudo tienen una arquitectura en forma de cadena, lo que los hace aplicables para tareas como NER, reconocimiento de voz, traducción de idiomas, etc.

Imaginemos entrenar a un RNN con la palabra ”dolor”, dadas las letras ”d, o, l, o, r”. El RNN se entrenará en cuatro ejemplos separados, cada uno correspondiente a la probabilidad de que las letras caigan en una secuencia deseada. Por ejemplo, la red estará capacitada para comprender la probabilidad de que la letra .º”deba seguir en el contexto de ”d”. De manera similar, la letra ”l”debería aparecer después de las secuencias de ”do”.

Nuevamente, se calculará una probabilidad para la letra .º”que sigue la secuencia ”dol”. El proceso continuará hasta que se calculen las probabilidades para determinar la probabilidad de que las letras caigan en la secuencia deseada. Entonces, a medida que la red recibe cada entrada, determinará la probabilidad de la letra siguiente en función de la probabilidad de la letra o secuencia anterior. Con el tiempo, la red se puede actualizar para producir resultados con mayor precisión.

Entrenamiento del modelo. Para la implementación de la arquitectura de la red se utilizó la librería spaCy[2] los detalles de la arquitectura pueden encontrarse en su pagina oficial para la versión 2.

Se entrenaron dos RNN diferentes la primera red se entreno desde cero con las 57 notas que contenían etiquetas de síntomas. Después se re-entreno el modelo de spaCy *es_core_news_sm* para poder aplicar transferencia de conocimiento.

La Figura 2 muestra como disminuye la pérdida del modelo a medida que avanzan las épocas, llegando a la pérdida máxima aproximadamente en la época 25 con la red RNN entrenada desde cero, también podemos notar que el modelo pre-entrenado sin importar el numero de épocas las perdidas que presenta son mucho mayores.

[2] https://v2.spacy.io/api/entityrecognizer

Detección de negaciones. Una vez extraídos los síntomas debemos de diferenciar o filtrar cuales de ellos están negados, para ello se ocupó un algoritmo para la identificación de negaciones llamado NegEX[3].

Estos síntomas filtrados serán posteriormente usados junto con las características antes mencionadas para identificar los mejores tratamientos de los que se cuenta con un registro. El algoritmo NegEX busca palabras clave llamados triggers en inglés, estas son palabras que se usan para denotar negación por ejemplo la frase no presenta.

El algoritmo verifica si una palabra de interés, en este caso un síntoma, se encuentra dentro del alcance del trigger de negación tomando en cuenta las palabras que se encuentran entre el síntoma y la negación y su ubicación dentro de la oración.

El algoritmo cuenta con las siguientes clases de triggers:

- PREN: Trigger que indica negación y precede al ending en la oración (por ejemplo:no se evidencia, no se observa).
- POST: Trigger que indica negación y aparece después del ending en la oración (por ejemplo: negado, tiene que ser descartado).
- PREP: Trigger que indica posible negación y precede al ending en la oración (por ejemplo: habrá que descartar, no se corresponderá).
- POSP: Trigger que indica posible negación y aparece después del ending en la oración (por ejemplo: podrá ser descartado).
- PSEU: Trigger que indica pseudo-negación y puede preceder al ending o aparecer después del el en la oración (por ejemplo: disminuye, no se incrementa).
- CONJ: Trigger que indica conjunción o terminación (por ejemplo: pero, aunque).

Este algoritmo originalmente se diseño para el idioma inglés pero en base al trabajo realizado por Costumero et al. [1] se elaboró una nueva adaptación de NegEx para español. Esta adaptación incluye la traducción de los triggers también, se incluyó una nueva clase de *triggers* para las conjunciones negadas en el español.

Pseudo código del algoritmo NegEX:

```
for each sentence do:
    for each negation trigger (Neg1) do:
        if Neg1 is a pseudo−negation trigger then:
            Go to next negation trigger in the sentence
        else if Neg1 is a pre−negation trigger then:
            // Find scope of Neg1 forward
            if (a termination term is found or
                                another negation or pseudo−negation trigger or
                                end of sentence) then:
                Finish scope of Neg1
            end if
        else if Neg1 is a post−negation trigger then:
            Find scope of Neg1 backwards based on word distance
        end if
    end for
end for
```

[3] https://code.google.com/archive/p/negex/

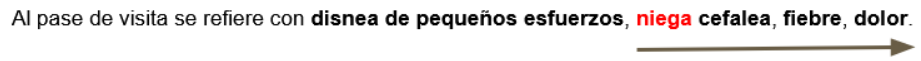


Fig. 3. Ejemplo de negación de síntomas.

En la figura 3 se han identificado los síntomas de disnea de pequeños esfuerzos, cefalea, fiebre y dolor. Podemos observar que la palabra niega afecta únicamente a los síntomas de cefalea, fiebre y dolor.

4.4. Evaluación

Para la evaluación de ambos métodos se decidió ocupar las métricas de *Precision*, *Recall* y *F1-Score*.

Para el modelo basado en redes neuronales debido a los pocos datos con los que se cuenta se utilizaron diversas técnicas para la evaluación del desempeño del modelo y métodos de re-muestreo para obtener más ejemplos durante el entrenamiento a partir de los datos proporcionados.

Validación cruzada. La validación cruzada es un procedimiento de re-muestreo que se utiliza para evaluar modelos de aprendizaje automático en una muestra de datos limitada.

1. Mezclar el conjunto de datos de forma aleatoria.
2. Dividir el conjunto de datos en k grupos.
3. Para cada grupo único:
   – Tomar un conjunto de datos de prueba.
   – Tome los grupos restantes como un conjunto de datos de entrenamiento.
   – Entrenar el modelo en el conjunto de entrenamiento y evaluarlo con en el conjunto de prueba.
   – Guardar la puntuación de la evaluación y descarte el modelo.
4. Resumir el rendimiento del modelo usando las puntuaciones de evaluación del modelo.

5. Resultados

En la Figura 4 se muestran los síntomas obtenidos de una nota médica de un paciente que presenta COVID.

El método basado en diccionarios resultó tener las calificaciones más bajas de ambos modelos. El problema principal de este método es la gran cantidad de falsos positivos obtenidos debido a que existen síntomas presentes en diferentes secciones de las notas médicas como es la sección del plan de manejo, en esta sección se le hacen recomendaciones al paciente, como la toma de ciertos medicamentos en caso, por ejemplo, de dolor o fiebre. Se sugiere que al segmentar la nota médica y sólo considerar ciertas secciones el método basado en diccionarios pueda mejorar su precisión.

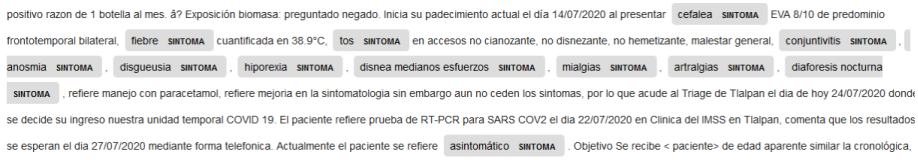


Fig. 4. Síntomas extraídos por nota médica y su localización.

Tabla 2. Resultados de la identificación de síntomas por MBD sobre las notas médicas completa y sin la sección de plan de manejo.

Métricas	**Nota médica completa**	**Nota médica sin la sección de plan de manejo**
Precision	20 %	29 %
Recall	6 %	8 %
F1-score	9 %	13 %

Tabla 3. Resultados de la identificación de síntomas por MBD aplicando y no la distancia de *Levenshtein*.

Métricas	**Con distancia de *Levenshtein***	**Sin distancia de *Levenshtein***
Precision	29 %	32 %
Recall	8 %	6 %
F1-score	13 %	11 %

Tabla 4. Resultados de la identificación de síntomas en el conjunto de prueba con los métodos basados en diccionario y redes recurrentes.

Métricas	**MBD**	**RNN**	**RNN re-entrenada**
Precision	20 %	96.37 %	75.0 %
Recall	6 %	95.68 %	75.0 %
F1-score	9 %	96.02 %	75.0 %

El cuadro 2 muestra las métricas de *Precision*, *Recall* y *F1-Score* del método basado en diccionario sobre toda la nota médica y sobre la nota médica sin la sección de plan de manejo. Se encontró además que la distancia de *Levenshtein* podría ocasionar falsos positivos, como lo es el caso de la conversión de uresis a enuresis.

El cuadro 3 muestra las métricas de *Precision*, *Recall* y *F1-Score* del método basado en diccionario sobre la nota médica sin la sección de plan de manejo, dado que ha mostrado mejores resultados, aplicando la distancia de *Levenshtein* y no aplicando la distancia de *Levenshtein*.

Como se puede observar en la Tabla 4 los resultados obtenidos con el modelo basado en RNN a pesar de ser altos pueden variar mucho, esto sucede a menudo cuando se entrena un modelo con pocos datos.

La red RNN re-entrenada tuvo peores resultados que la RNN entrenada desde cero, esto puede deberse a que el contexto en que ambas fueron entrenadas es muy diferente.

6. Conclusiones

En este artículo presentamos un método basado en redes neuronales para la identificación de síntomas los resultados obtenidos se muestran en el cuadro 4.

Comparamos este método con un sistema basado en diccionarios y encontramos que el método basado en aprendizaje automático logra mejorar los resultados en un 76.02 % en comparación con el método basado en diccionarios, esto dada la gran cantidad de falsos positivos encontrados principalmente en la sección de plan de manejo de la nota médica, se sugirió que al segmentar la nota el método basado en diccionarios mejoraría su precisión, además de revisar que tanto la distancia de *Levenshtein* mejoraba o no sus métricas.

Al final la segmentación logró mejorar los resultados y se observó que sobre ese conjunto segmentado el no ocupar la distancia de *Levenshtein* mejoraba aún más los resultados.

Las redes neuronales recurrentes aplicadas al campo del Procesamiento de Lenguaje Natural prometen un análisis contextual, por lo que es un buen identificador del análisis contextual presente en la extracción de entidades médicas, particularmente en el caso de la sintomatología.

Como trabajo futuro se espera poder recibir más datos y poder re-entrenar el modelo basado en redes neuronales y compararlo con otros modelos como *Conditional Random Fields*.

También se desea probar otros métodos para la detección de las negaciones, como el método basado en *POSTagging*, el cual es un método basado en reglas que usa la información morfológica de las palabras en cada oración para poder encontrar patrones en común. Estos patrones sirven para crear reglas que determinan si un término esta negado o no.

Referencias

1. Costumero, R., Lopez, F., Gonzalo-Martin, C., Millan, M., Menasalvas, E.: An approach to detect negation on medical documents in spanish. In: International conference on Brain Informatics and Health, pp. 366–375 (2014) doi: 10.1007/978-3-319-09891-3_34
2. Koleck, T. A., Dreisbach, C., Bourne, P. E., Bakken, S.: Natural lenguage processing of symptoms documented in free-text narratives of electronic health records: A systematic review. Journal of the American Medical Informatics Association, vol. 26, no. 4, pp. 364–379 (2019) doi: 10.1093/jamia/ocy173
3. Sampathkumar, H., Chen, X. W., Luo, B.: Mining adverse drug reactions from online healthcare forums using hidden Markov model. BMC Medical Informatics and Decision Making, vol. 14, no. 1, pp. 1–18 (2014) doi: 10.1186/1472-6947-14-91
4. Savova, G. K., Masanz, J. J., Ogren, P. V., Zheng, J., Sohn, S., Kipper-Schuler, K. C., Chute, C. G.: Mayo clinical text analysis and knowledge extraction system (cTAKES): architecture, component evaluation and applications. Journal of the American Medical Informatics Association, vol. 17, no. 5, pp. 507–513 (2010) doi: 10.1136/jamia.2009.001560
5. Sobhana, N., Mitra, P., Ghosh, S.: Conditional random field based named entity recognition in geological text. International Journal of Computer Applications, vol. 1, no. 3, pp. 143–147 (2010)

6. Yujian, L., Bo, L.: A normalized levenshtein distance metric. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1091–1095 (2007) doi: 10.1109/TPAMI.2007.1078
7. Zhou, H., Guo, W., Ke, D., Liu, N., Zhao, X., Li, C.: Annotations of chinese electronic medical record using BiLSTM-CRF based networks. In: Proceedings of the 2019 International Symposium on Signal Processing Systems, pp. 131–135 (2019) doi: 10.1145/3364908.3365290

ISSN 1870-4069

Análisis preliminar del sentimiento sobre la vacunación del COVID-19 en México

Luis Norberto Zúñiga-Morales[3], Arturo Zúñiga-López[1], Juan Villegas-Cortez[2], Carlos Avilés-Cruz[1] Felipe Morales-Torres[3]

[1] Universidad Autónoma Metropolitana, Departamento de Electrónica, Unidad Azcapotzalco, México

[2] Universidad Autónoma Metropolitana, Departamento de Sistemas, Unidad Azcapotzalco, México

[3] Universidad Iberoamericana, Instituto de Investigación Aplicada y Tecnología, México

{azl, juanvc, caviles}@azc.uam.mx, {lzun.morales, felipe.mor.torres}@gmail.com

Resumen. El análisis de información en las redes sociales hoy día es un tema de interés general para muchas disciplinas del conocimiento, esto porque se han convertido en un instrumento de comunicación de información masivo. Actualmente es común que las personas determinen su criterio por lo que ven como información mayoritaria en la Internet y específicamente en redes sociales. En este trabajo presentamos un análisis de sentimientos, basados en textos compartidos en la red social Twitter, los llamados tuits, mismos que previamente han sido clasificados en tres estados. Los tuits tienen una complejidad lexicográfica que representan un nuevo reto para el reconocimiento de patrones, y es así que nosotros presentamos la aplicación de la máquina de soporte vectorial sobre una gran cantidad de tuits referidos al tema de la pandemia del COVID-19 y la vacunación. El resultado alcanzado en clasificación en esta etapa preliminar no es alta por la poca cantidad de tuist categorizados, sin embargo, podemos considerar que la continuación del trabajo a futuro puede ser de apoyo para criterios de políticas sociales y de salud, además de entender desde una nueva perspectiva este tipo de patrones.

Palabras clave: COVID-19, reconocimiento de patrones, aprendizaje profundo, análisis de sentimientos, redes sociales

Luis Norberto Zúñiga-Morales, Arturo Zúñiga-López, Juan Villegas-Cortez, et al.

Preliminary Sentiment Analysis of COVID-19 Vaccination in Mexico

Abstract. The analysis of information in social networks today is a topic of general interest for many disciplines of knowledge, because they have become an instrument of mass information communication. Nowadays it is common for people to determine their criteria by what they see as majority information on the Internet and specifically in social networks. In this paper we present a sentiment analysis, based on texts shared on the social network Twitter, the called tweets, which have been previously classified into three states. The tweets have a lexicographic complexity that represent a new challenge for pattern recognition, and so we present the application of the support vector machine on a large number of tweets referring to the topic of the COVID-19 pandemic and vaccination.The result achieved in classification at this preliminary stage is not high due to the small amount of categorized tweets, however, we can consider that the continuation of the work in the future can be of support for social and health policy criteria, in addition to understanding from a new perspective this type of patterns.

Keywords: COVID-19, pattern recognition, deep learning, sentiment analysis, social networks.

1. Introducción

Hoy en día los servicios de redes sociales han cambiado la forma en que las personas expresan sus opiniones y puntos de vista [22] e.g., Twitter es una red social muy popular de mensajes cortos, con 140 caracteres como máximo en su primera etapa, y desde el año 2020 permite 280 caracteres como máximo, que pretende ser un reflejo de lo que está pasando en un momento dado, como el brote de coronavirus que ha supuesto un problema grave para la economía mundial y ha afectado a la mayoría de las naciones.

Lo anterior ha provocado restricciones de viaje, cierre de negocios no esenciales y procedimientos de cuarentena. A la luz de estas medidas, la mayoría de la gente ha recurrido a las redes sociales para expresar su opinión sobre todo lo que está sucediendo en el mundo. El impacto de las plataformas de redes sociales se está volviendo más notable que nunca. Los sitios de redes sociales se consideran el gran centro de datos global porque las personas usan sus aplicaciones e invierten mucho tiempo en estos medios de comunicación [1].

Los recientes desarrollos en el campo de los sistemas de información y las plataformas de intercambio de opiniones han impulsado la investigación para analizar las opiniones expresadas en estas redes sociales, que se presenta en la literatura como "análisis de sentimientos" [1]. El límite de 140 caracteres por tuit hace que los tuits sean concisos y fáciles de entender, al tiempo que brindan una idea de las opiniones y sentimientos de las personas, de ahí que hay varios estudios que pretenden usar Twitter para analizar la situación del COVID-19 a nivel mundial.

Las vacunas son sin duda uno de los mayores logros de la medicina moderna, y hay esperanzas de que puedan constituir una solución para detener la pandemia de COVID-19 en curso [10], sin embargo, en la última década la oposición a las vacunas ha encontrado un lugar en los medios digitales y sociales como medio principal de organización y difusión de información, además aunado a la creciente preocupación por los derechos humanos y el escepticismo hacia la vacuna y sus efectos pueden hacer que el proceso de vacunación se convierta en una tarea complicada [16].

Desde el comienzo de la pandemia, las noticias recientes han puesto de relieve el aumento de la desinformación en línea y la oposición a las vacunas [4]. Estudios recientes han demostrado que los mensajes relacionados con la vacunación son uno de los vectores más activos para la propagación de información errónea y desinformación sobre salud. Aunque son una pequeña fracción del público en general, los oponentes a las vacunas tienen una presencia enorme en línea y especialmente en Twitter.

A pesar de los recientes esfuerzos de Twitter para limitar la propagación de afirmaciones de salud falsas y engañosas, muchas cuentas de usuario oponentes a las vacunas permanecen activas en la plataforma. Aunque algunas cuentas tuitean casi exclusivamente sobre vacunas, muchas otras también discuten otros tipos de contenido, lo que permite identificar subgrupos en función de intereses compartidos como la política, la salud pública o la actualidad [12].

Esta investigación tiene como objetivo identificar los sentimientos sobre la vacunación del COVID-19 en México a partir de tuits. El Análisis de Sentimientos (AS) o minería de opinión, es una tarea de clasificación automática de textos que utiliza diversas herramientas de disciplinas como Procesamiento de Lenguaje Natural, Lingüística Computacional y Minería de Textos [7, 11]. En este trabajo de tipo ingenieril nos apegamos a entender por "sentimiento" la primera acepción del diccionario VOX de la Lengua Española[4], que lo define como "Estado de ánimo o disposición emocional hacia una cosa, un hecho o una persona".

Los tuits se clasifican en positivos, neutrales o negativos. Nos permitimos aclarar esto para delimitar nuestro alcance de estudio desde la inteligencia artificial, respetando a los profesionales de la salud mental, psicólogos y psiquiatras, que consideramos les podemos aportar para ellos dar un análisis apoyado en nuestras conclusiones.

En la sección 2 presentamos el estado del arte de nuestra investigación, la metodología propuesta la mostramos en la sección 3, los experimentos y el análisis de sus resultados se exponen en la sección 4 y, finalmente compartimos nuestras conclusiones en la sección 5.

2. Estado del arte

La gran mayoría de los estudios de investigación que cubren el análisis del sentimiento de los tuits se inclinan más hacia los algoritmos de aprendizaje automático [18]. Los investigadores a menudo utilizan una metodología exploratoria y descriptiva, así como los datos visuales y textuales, para obtener información valiosa basada en el método de clasificación de aprendizaje automático [22].

[4] Diccionario General de la Lengua Española Vox. Copyright © 2012, 2020 Larousse Editorial, S.L., under licence to Oxford University Press. All rights reserved.

Sethi et al. [19], hizo predicciones de los sentimientos de las personas en Twitter mediante la construcción de un modelo para explorar el sentimiento real de las personas sobre COVID-19. Hicieron una comparación entre cinco clasificadores, que son regresión logística, Naïve Bayes multinomial, árboles de decisión, bosque aleatorio, XGBoost y Maquinas de soporte vectorial (SVM).

Los resultados mostraron que SVM y los árboles de decisión superan al otro clasificador. Sin embargo, el clasificador SVM es estable y confiable en todas las pruebas. Además, la precisión máxima del modelo propuesto fue del 93 %, lo que indica numéricamente que el modelo tiene la capacidad de analizar la emoción de las personas dentro de los tuits "COVID-19".

Chakraborty et al. [6], analizaron tuits retuiteados con aprendizaje profundo, el análisis revela que si bien las personas tuiteaban principalmente de manera positiva sobre COVID-19, los usuarios de Internet estaban ocupados re-tuiteando tuits negativos y que no se podían encontrar términos útiles. La precisión alcanzó hasta el 81 % cuando se usan clasificadores de aprendizaje profundo, mientras que el 79 % cuando se usa el modelo formulado basado en una regla difusa para identificar los sentimientos de los tuits.

Abdulaziz et al. [1], analizaron en un conjunto de datos de tuits en inglés sobre COVID-19. La implementación se realizó utilizando LDA (Latent Dirichlet Allocation) para encontrar los temas más importantes relacionados con el Coronavirus. Se entrenó con el 80 % del conjunto de datos y se probó con el 20 %. Además, se utilizó el sentimientos de los tuits recopilados utilizando un enfoques basados en el léxico para clasificar los sentimientos de las personas.

Sontayasara et al. [22], analizaron sentimientos utilizando el algoritmo de máquina de soporte vectorial. Los resultados mostraron una precisión de clasificación del 75.83 % basada en tres clasificaciones de sentimiento: positivo y negativo. Por tanto, este estudio podría proporcionar una idea de las opiniones y sentimientos de los viajeros relacionados con el negocio del turismo.

Rustam et al. [17], realizaron un análisis de opinión de los tuits de COVID-19 utilizando un enfoque de aprendizaje automático supervisado. Se utilizó la arquitectura Long Short-Term Memory (LSTM) del modelo de aprendizaje profundo, para la obtención de información. Singh et al. [20], estudian las opiniones de las personas para comprender su estado mental, para lo cual realizan un análisis de sentimientos utilizando el modelo BERT en los tuits.

Utilizan dos conjuntos de datos; un conjunto de datos que se recopila mediante tuits hechos por personas de todo el mundo, y el otro conjunto de datos que contiene los tuits hechos por personas de la India. Los resultados experimentales muestran que la precisión de la validación su modelo es de aproximadamente el 94 %.

3. Metodología

Nuestro texto de trabajo comprende diversas publicaciones de Twitter cuyo tema se encuentra relacionado con el COVID-19, para las cuales se busca determinar su sentimiento para analizar su comportamiento con el tiempo.

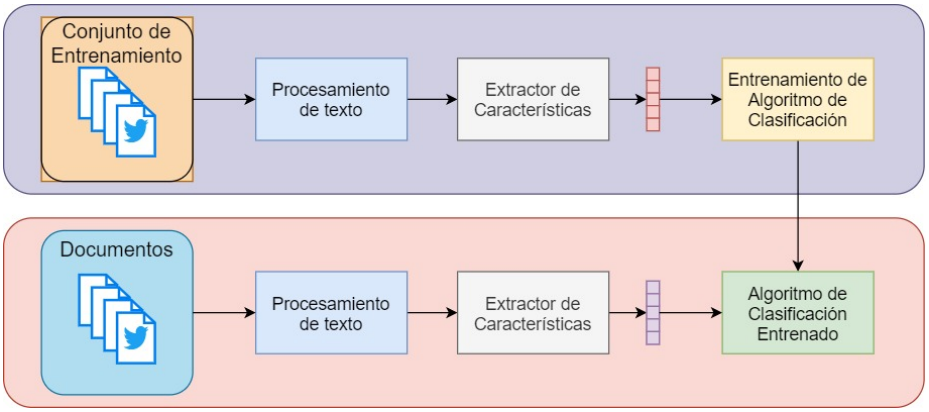


Fig. 1. Diagrama general de la metodología propuesta. El bloque superior representa el módulo de entrenamiento, mientras que el inferior representa el módulo de clasificación.

Para lograr esta clasificación, se propone un modelo compuesto de dos fases: 1) Fase de entrenamiento, y 2) Fase de clasificación. La Figura (1) muestra la idea general del modelo propuesto. En la fase de entrenamiento se busca entrenar un algoritmo de aprendizaje supervisado con información previamente anotada.

Durante la fase de clasificación, se utiliza el clasificador obtenido en el punto anterior para determinar la clase de cada documento facilitando su posterior análisis.

3.1. Construcción del conjunto de datos

La construcción del conjunto de datos consta de dos fases: la recopilación de información y el procesamiento de datos.

Recopilación de datos. Para la recopilación de datos se eligió Twitter como fuente de origen. Específicamente, se utilizó la API v2[5] de Twitter, con una cuenta con acceso a la modalidad de investigador académico.

El conjunto de datos incluye únicamente mensajes publicados entre el 1° de enero de 2020 y el 31 de marzo de 2021, relacionados con COVID-19, en idioma español y cuyo origen geográfico fuese México. Además, dentro de la consulta se especificó evitar publicaciones marcadas como retuit para evitar la repetición de publicaciones.

Para cada tuit se extrajeron los siguientes campos: texto, identificador del tuit, fecha de publicación y métricas públicas, la cual incluye conteos de retuits, respuestas, *likes* y *quotes*. La Figura 2 muestra la distribución de la información del conjunto de datos en un periodo mensual.

Anotación de datos. Para el modelo propuesto se consideran únicamente dos clases para los datos, "-1" y "1". La etiqueta -1 se utiliza para denotar aquellos mensajes que expresen un sentimiento negativo hacia el tema del COVID-19.

[5] Twitter API v2, URL: https://developer.twitter.com/en/docs/twitter-api/early-access

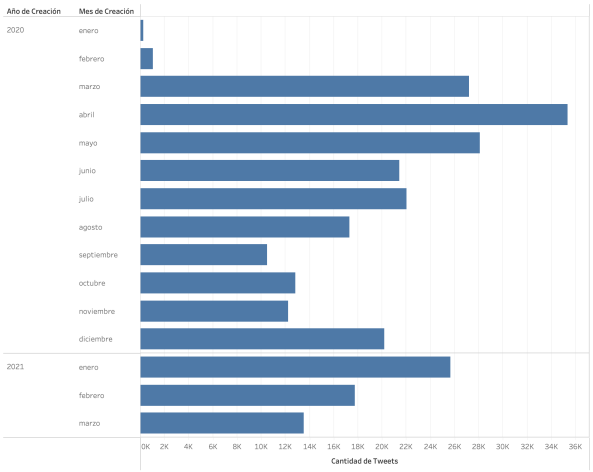


Fig. 2. Distribución temporal del año 2020 y parte del 2021, de los tuits recopiladas en el conjunto de datos.

Por otro lado, la etiqueta 1 se usa para identificar mensajes con una inclinación positiva y ajena hacia el COVID-19. Para su anotación, se recurrió a tres expertos en el campo de salud pública a los que se les facilitó una guía con las instrucciones de anotación. Al final, solamente se conservaron los tuits donde los tres expertos anotaron el mismo valor.

Procesamiento de datos. El preprocesamiento de datos se encarga de normalizar el texto para su posterior alimentación a los algoritmos de clasificación. El primer paso es cambiar el texto de cada tuit a letra minúscula. En un tuit se presentan diversos elementos que sirven para etiquetar diversos elementos de difusión, tales como el hashtag (#), cashtag ($) o menciones de usuario (@).

Estas etiquetas, aunque útiles para identificar información relevante, no suelen aportar información para el clasificador al no ser propios de una clase [13]. En nuestro caso, se eliminan cashtags debido a que no se habla de un tema financiero predominante en la investigación, y se eliminan las menciones de usuario por respeto a su privacidad.

Además, se elimina cualquier enlace a otro sitio presente en las publicaciones. Posteriormente, el texto filtrado se somete a un proceso de tokenización y lematización para su análisis previo, pero no para la extracción de características.

3.2. Análisis de sentimientos

Para realizar el AS sobre el conjunto de datos que se construyó, se utiliza la siguiente estrategia. Primero, se extraen características del texto de cada tuit por medio de BERT (Bidirectional Encoder Representations from Transformers). Después, las características extraídas se alimentan a un algoritmo de aprendizaje supervisado para su entrenamiento.

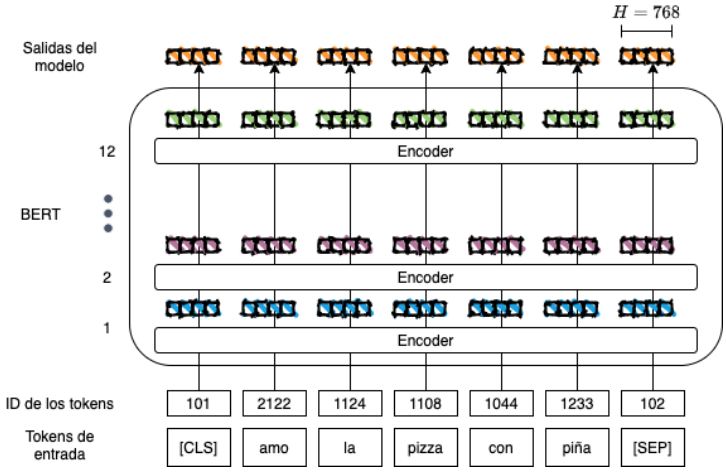


Fig. 3. Esquema general del modelo de lenguaje BERT en su modalidad base, la cual tiene 12 capas de encoders, 12 cabezales de auto atención y el tamaño del vector oculto es 768. Nótese los tokens adicionales [CLS] y [SEP] que se añaden al texto.

Extracción de características. BERT [8] es un modelo de representación de lenguaje basado en la idea de pre-entrenar modelos de lenguaje. La novedad de BERT es su entrenamiento bidireccional, ya que considera el contexto de las palabras de derecha a izquierda y viceversa, al mismo tiempo.

Para cumplir sus objetivos, BERT emplea Transformers [24], específicamente la arquitectura de sus codificadores. Inicialmente, BERT fue entrenado para trabajar con dos idiomas, chino e inglés, pero poco a poco se generaron modelos que abarcaron distintos idiomas.

Para el modelo del artículo se utiliza BETO [5], un modelo de BERT pre-entrenado con un gran corpus en español. El modelo consta de 12 capas de auto atención cada uno con 16 cabezales de atención, donde el tamaño vector oculto es 1024.

Algoritmo de clasificación. Para clasificar cada tuit según su polaridad, se utiliza un algoritmo de aprendizaje supervisado. En particular, se elige a la Máquina de Vectores de Soporte (MVS) [23], un método de clasificación que mapea datos de un conjunto de entrenamiento a diferentes espacios para construir un hiperplano que permita separar los miembros de cada clase.

La MVS ha demostrado ser un algoritmo de aprendizaje fuerte para la clasificación de texto [14] y puede considerarse como un método para establecer un marco de referencia al comparar distintos métodos de clasificación debido a su desempeño en tareas para clasificar la polaridad del sentimiento en tuits [9].

Validación del clasificador. La matriz de confusión [21] permite visualizar el desempeño de un algoritmo de clasificación. Cada columna indica la clase que que el clasificador predice y cada fila es la clase real a la que pertenece. La Tabla 1 muestra la matriz de confusión para el caso de clasificación binaria.

Tabla 1. Matriz de confusión para el caso de clasificación binaria, donde se elige una clase para que sea considerada la positiva y la otra la negativa.

		Valor Predicho	
		Clase 1	**Clase 2**
Valor	Clase 1	Positivo Verdadero(PV)	Falso Negativo(FN)
Real	Clase 2	Falso Positivo(FP)	Negativo Verdadero(NV)

Para evaluar el desempeño de los clasificadores se emplean cuatro medidas: precisión (precision), exhaustividad (recall), exactitud (accuracy) y la medida $F1$. En el caso de clasificación binaria se definen como:

– Exactitud: la medida más intuitiva, la razón de las instancias clasificadas correctamente y el total de los elementos clasificados:

$$\text{Exactitud} = \frac{PV + NV}{PV + NV + FP + FN}. \tag{1}$$

– Precisión: examina la razón de instancias clasificadas positivamente correctas:

$$\text{Precisión} = \frac{PV}{PV + FP}. \tag{2}$$

– Exhaustividad: efectividad del clasificador para identificar etiquetas positivas:

$$\text{Exhaustividad} = \frac{PV}{PV + FN}. \tag{3}$$

– Medida F1: es un promedio ponderado de la precisión y la exhaustividad:

$$\text{F1} = \frac{2 \cdot \text{Exhaustividad} \cdot \text{Precisión}}{\text{Exhaustividad} + \text{Precisión}}. \tag{4}$$

4. Experimentos y resultados

En esta sección se muestran los detalles de la implementación expuesta en la sección anterior, así como los resultados obtenidos y su análisis.

Para la implementación del modelo se utilizó un equipo de cómputo con sistema operativo MS-Windows ver 10, 64 bits, procesador Intel Core i7-7700HQ 2.80GHz, 16 GB RAM y una GPU GeForce GTX 1080. El modelo se programó en su totalidad con Python 3.8.

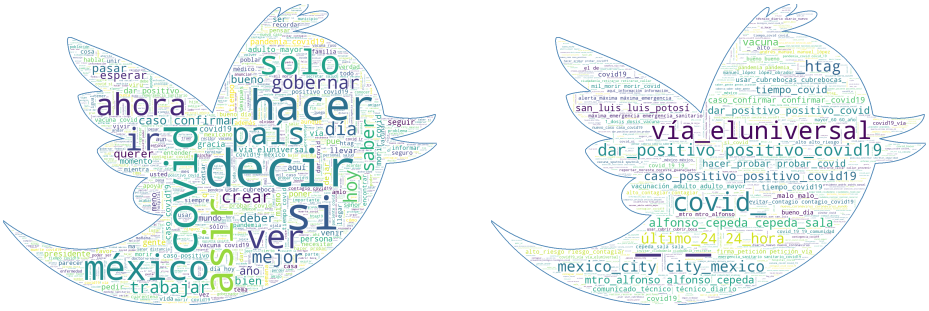
En la Tabla 2 se muestra un pequeño resumen del conjunto de datos.

A continuación, se presenta un análisis preliminar del conjunto de datos, el cual permite obtener una idea general del contenido del mismo. En particular, se realiza un análisis visual por medio de una diagrama de nubes de palabras y se explora los diferentes tópicos que conforman el conjunto de datos por medio de LDA.

En la Figura (4) se puede apreciar el diagrama de nube de palabras que se realiza al procesar cada tuit en el conjunto de datos.

Tabla 2. Resumen del conjunto de datos COVID-19 MX.

Periodo de Recolección	01/01/2020 - 31/03/2021
Cantidad de tuits recolectados	265,448
Query	(vacuna OR vacunación OR vacunar OR covid OR covid19 OR covid-19) lang:es, place_country:MX, -is:retweet
Campos solicitados	author_id, text, retweet_count, reply_count, like_count, quote_count, id, created_at



(a) (b)

Fig. 4. Diagrama de nube de palabras para (a) palabras lematizadas y (b) bigramas de palabras lematizadas que consideran todos los tuits del conjunto de datos COVID-19 MX. Entre más grande sea la palabra en la imagen, mayor es su frecuencia entre los tuits.

Cabe resaltar que se lematiza cada palabra, lo que ayuda a reducir la dimensión del conjunto conformado por las palabras únicas de los tuits. Como es de esperar, lo relacionado con COVID-19 resulta ser el tema principal de los documentos.

Sin embargo, también es posible apreciar aspectos relacionados como la política, la situación laboral y económica generada por la crisis, y la presencia de medios informativos digitales, entre otros.

El modelado de temas [2] es un tipo de modelo estadístico que permite inferir distintos temas que ocurren en una colección de documentos. La idea principal es que, entre más se relacionan ciertas palabras, se espera que aparezcan de forma conjunta en varios documentos de la colección.

El LDA [3] representa cada documento como una mezcla de temas por medio de palabras y su probabilidad. En este trabajo, se utiliza la implementación de LDA de Gensim[6], utilizando Term Frequency Inverse Document Frequency (TFIDF).

La Tabla (3) muestra los resultados obtenidos en el modelado de temas. Aunque el algoritmo permite modelar los documentos por medio de ciertas palabras, es responsabilidad de los autores interpretar los resultados.

[6] URL: https://radimrehurek.com/gensim/

Tabla 3. Resultado del modelado de temas usando TFIDF.

Tema	Composición
Vacunación y Vuelta a la Normalidad	Palabra: 0.012*covid + 0.008*si + 0.008*covid19 + 0.007*ir + 0.006*hacer + 0.005*casa + 0.005*poder + 0.004*vacuna + 0.004*ver + 0.004*dar
Casos confirmados y muertes	Palabra: 0.020*caso + 0.013*confirmar + 0.012*covid19 + 0.008*méxico + 0.007*2020 + 0.007*nuevo + 0.007*coronavirus + 0.006*defunción + 0.006*reportar + 0.006*mil
Vacunación	Palabra: 0.008*covid19 + 0.007*covid + 0.006*si + 0.006*decir + 0.006*hacer + 0.005*ir + 0.005*vacuna + 0.004*dar + 0.004*gobernar + 0.004*19
Consejos para la pandemia	Palabra: 0.008*covid19 + 0.008*mexico + 0.007*tiempo + 0.006*covid + 0.004*hacer + 0.004*día + 0.004*poder + 0.004*bueno + 0.004*probar + 0.004*si
Personal e instituciones de Salud	Palabra: 0.008*covid19 + 0.008*medida + 0.006*salud + 0.005*contingencia + 0.004*prevención + 0.004*sanitario + 0.004*hospital + 0.003*evitar + 0.003*pandemia + 0.003*médico

Tabla 4. Estadísticas del entrenamiento clasificador.

Precisión	Exhaustividad	Exactitud	F1
75.83 %	75.34 %	76.12 %	75.83 %

4.1. Extracción de características

Para la extracción de características se utiliza el modelo pre-entrenado BETO y su implementación en Huggingface[7] y Pytorch[8]. Para implementar BERT (y BETO), se utiliza un tipo especial de tokenización [8], la cual se encuentra incluida en el modelo, por lo que no se realiza el proceso de tokenización tradicional para extraer *word-embeddings*.

En cuanto al vector de características, Devlin et al. [8] ofrecen diversas opciones al momento de considerar los *word-embeddings* para las palabras. Sin embargo, para la tarea de clasificación, utilizan el último vector oculto del token especial [CLS], como se ve en la Figura (3). Este último es el vector de características que se alimenta al algoritmo de clasificación (MVS) para determinar la clase a la que pertenece.

4.2. Algoritmo de clasificación

Para la implementación de la MVS se utilizó la librería Scikit-learn [15], una librería de aprendizaje automático para Python. Utilizando los vectores de características extraídos con BETO, se utiliza una SVM con kernel de función de base radial utilizando como parámetros $\gamma = 2{,}3 \times 10^{-4}$ y $c = 1$.

[7] URL: https://huggingface.co/dccuchile/bert-base-spanish-wwm-cased

[8] URL: https://pytorch.org/

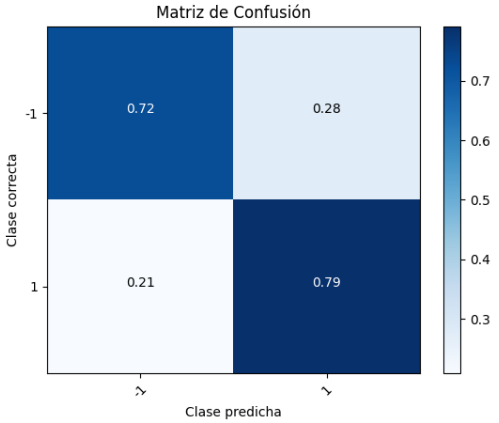


Fig. 5. Matriz de confusión para el clasificador entrenado.

Además, se realiza validación cruzada con 10 pliegues en una proporción $80 - 20$. En total, se usaron $2,113$ datos anotados previamente para el entrenamiento del clasificador.

La Figura (5) muestra la matriz de confusión del modelo entrenado, en la Figura (6) se observa la distribución de tuits positivos y negativos en un periodo mensual, y la Tabla (4) muestra las estadísticas del clasificador construido.

5. Conclusiones

El análisis en la redes sociales hoy en día es un tema de interés general para muchas disciplinas del conocimiento, esto porque se han convertido en un instrumento de comunicación de información masiva. Actualmente es común que las personas determinen su criterio por lo que ven como información mayoritaria en la Internet y específicamente en redes sociales.

Es este trabajo presentamos un análisis preliminar basados en textos compartidos en la red social Twitter, los llamados tuits, mismos que previamente han sido clasificados en dos estados.

Los tuits tienen una complejidad lexicográfica que representa un nuevo reto para el reconocimiento de patrones, y es así que nosotros presentamos la aplicación de una maquina de soporte vectorial sobre una cantidad tuits referidos al tema de la pandemia del COVID-19 y la vacunación para México, pero consideramos que se puede aplicar este estudio para otros países.

El resultado alcanzado en esta etapa preliminar no es alta si sólo contemplamos la poca cantidad de tuits categorizados (2,113), pero estamos por encima de otros trabajos que presentan metodologías numéricas más profundas, por todo esto podemos considerar que la continuación del trabajo a futuro puede ser de apoyo para criterios de políticas sociales y de salud, además de entender desde una nueva perspectiva este tipo de patrones.

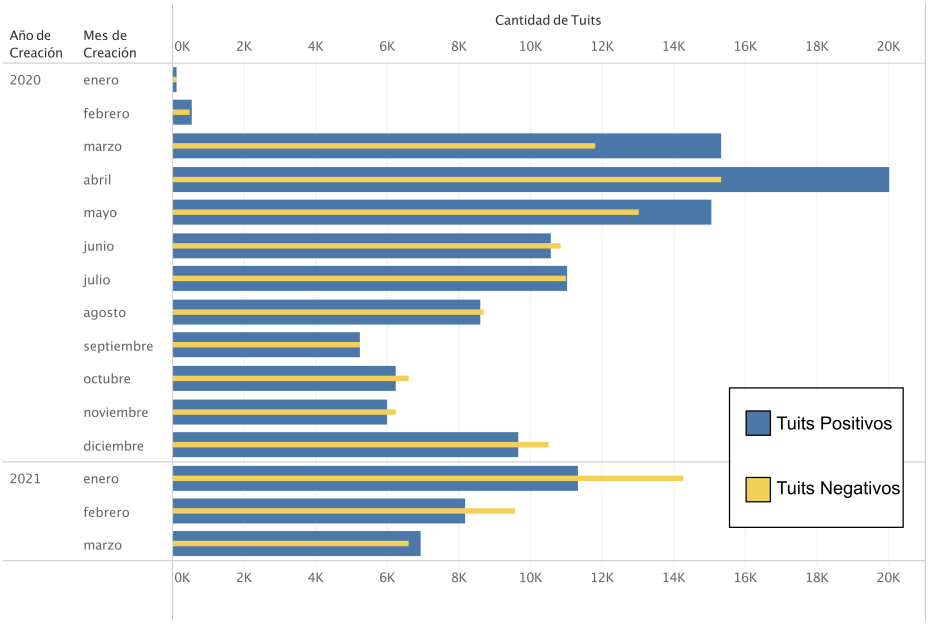


Fig. 6. Distribución temporal de tuits positivos (azul) y negativos (amarillo) del conjunto de datos completo. El etiquetado masivo se realiza mediante el clasificador entrenado (MVS) previamente.

La información en Twitter se genera de forma dinámica y creciente dependiendo los temas de tendencia en una región geográfica determinada, y esto nos ayuda a orientar nuestros esfuerzos hacia un análisis rápido con ventanas de tiempo-captura de la información, que pueda proporcionar a los especialistas de estudios sociales de los temas de interés, una herramienta en tiempo real de los hashtags de interés. Para trabajo a futuro se ha pensado en tener más datos etiquetados y probar con otras técnica de aprendizaje para compararlas y ver quien ofrece mejores resultados.

Referencias

1. Abdulaziz, M., Alsolamy, M., Alotaibi, A., Alabbas, A.: Topic based sentiment analysis for covid-19 tweets. International Journal of Advanced Computer Science and Applications, vol. 12, no. 1, pp. 626–636 (2021) doi: 10.14569/IJACSA.2021.0120172
2. Blei, D. M.: Probabilistic topic models. Communications of the ACM, vol. 55, no. 4, pp. 77–84 (2010) doi: 10.1145/2133806.2133826
3. Blei, D. M., Ng, A. Y., Jordan, M. I.: Latent dirichlet allocation. Journal of Machine Learning Research, vol. 3, pp. 993–1022 (2003)
4. Bonnevie, E., Gallegos-Jeffrey, A., Goldbarg, J., Byrd, B., Smyser, J.: Quantifying the rise of vaccine opposition on twitter during the COVID-19 pandemic. Journal of Communication in Healthcare, vol. 14, no. 1, pp. 12–19 (2021) doi: 10.1080/17538068.2020.1858222
5. Canete, J., Chaperon, G., Fuentes, R., Ho, J. H., Kang, H., Pérez, J.: Spanish pre-trained BERT model and evaluation data. In: Practical Machine Learning for Developing Countries (2020)

6. Chakraborty, K., Bhatia, S., Bhattacharyya, S., Platos, J., Bag, R., Hassanien, A. E.: Sentiment analysis of COVID-19 tweets by deep learning classifiers - a study to show how popularity is affecting accuracy in social media. Applied Soft Computing, vol. 97 (2020) doi: 10.1016/j.asoc.2020.106754
7. Daily, S. B., James, M. T., Cherry, D., Porter, J., Darnell, S. S., Isaac, J., Roy, T.: Affective computing: Historical foundations, current applications, and future trends. In: Emotions and Affect in Human Factors and Human-Computer Interaction, pp. 213–231 (2017)
8. Devlin, J., Chang, M. W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding, (2018) doi: 10.48550/arXiv.1810.04805
9. Elbagir, S., Yang, J.: Sentiment analysis of twitter data using machine learning techniques and scikit-learn. In: Proceedings of the International Conference on Algorithms, Computing and Artificial Intelligence, pp. 1–5, no. 57 (2018) doi: 10.1145/3302425.3302492
10. Germani, F., Biller-Andorno, N.: The anti-vaccination infodemic on social media: A behavioral analysis. Plos One, vol. 16, no. 3, pp. e0247642 (2021) doi: 10.1371/journal.pone.0247642
11. Guo, S., Zhang, G.: Using machine learning for analyzing sentiment orientations toward eight countries. Sage Open, vol. 10, no. 3, pp. 1–15 (2020) doi: 0.1177/2158244020951268
12. Jamison, A. M., Broniatowski, D. A., Dredze, M., Sangraula, A., Smith, M. C., Quinn, S. C.: Not just conspiracy theories: Vaccine opponents and proponents add to the covid-19 'infodemic' on twitter. Harvard Kennedy School Misinformation Review, vol. 1, no. 3 (2020) doi: 10.37016/mr-2020-38
13. Oliveira, N., Cortez, P., Areal, N.: Stock market sentiment lexicon acquisition using microblogging data and statistical measures. Decision Support Systems, vol. 85, pp. 62–73 (2016) doi: 10.1016/j.dss.2016.02.013
14. Patil, G., Galande, V., Kekan, V., Dange, K.: Sentiment analysis using support vector machine. International Journal of Innovative Research in Computerand Communication Engineering, vol. 2, no. 1, pp. 2607–2612 (2014)
15. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D.: Scikit-learn: Machine learning in python. Journal of Machine Learning Research, vol. 12, pp. 2825–2830 (2011)
16. Praveen, S., Ittamalla, R., Deepak, G.: Analyzing the attitude of indian citizens towards COVID-19 vaccine–a text analytics study. Diabetes & Metabolic Syndrome: Clinical Research & Reviews, vol. 15, no. 2, pp. 595–599 (2021) doi: 10.1016/j.dsx.2021.02.031
17. Rustam, F., Khalid, M., Aslam, W., Rupapara, V., Mehmood, A., Choi, G. S.: A performance comparison of supervisedmachine learning models for covid-19 tweets sentiment analysis. PLoS ONE, vol. 16, no. 2, pp. 1–23 (2021) doi: 10.1371/journal.pone.0245909
18. Samuel, J., Ali, G. G. M. N., Rahman, M. M., Esawi, E., Samuel, Y.: COVID-19 public sentiment insights and machine learning for tweets classification. Information, vol. 11, no. 6, pp. 314 (2020) doi: 10.3390/info11060314
19. Sethi, M., Pandey, S., Trar, P., Soni, P.: Sentiment identification in covid-19 specific tweets. In: International Conference on Electronics and Sustainable Communication Systems, pp. 509–516 (2020) doi: 10.1109/ICESC48915.2020.9155674
20. Singh, M., Jakhar, A. K., Pandey, S.: Sentiment analysis on the impact of coronavirus in social life using the BERT model. Social Network Analysis and Mining, vol. 11, no. 1 (2021) doi: 10.1007/s13278-021-00737-z
21. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Information Processing & Management, vol. 45, no. 4, pp. 427–437 (2009) doi: 10.1016/j.ipm.2009.03.002

22. Sontayasara, T., Jariyapongpaiboon, S., Promjun, A., Seelpipat, N., Saengtabtim, K., Tang, J., Leelawat, N.: Twitter sentiment analysis of bangkok tourism during covid-19 pandemic using support vector machine algorithm. Journal of Disaster Research, vol. 16, no. 1, pp. 24–30 (2021) doi: 10.20965/jdr.2021.p0024
23. Vapnik, V., Cortes, C.: Support-vector networks. Machine Learning, vol. 20, no. 3, pp. 273–297 (1995)
24. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems, vol. 30 (2017)

ISSN 1870-4069

Perfilado demográfico de celebridades de redes sociales

Juan-Carlos Alonso-Sánchez, Luis-Miguel López-Santamaría, Juan Carlos Gomez

Universidad de Guanajuato,
Departamento de Ingeniería Electrónica,
México

{jc.alonsosanchez, lm.lopezsantamaria, jc.gomez}@ugto.mx

Resumen. El perfilado de autor en redes sociales es una tarea que ha tomado auge en los últimos años para tratar de predecir de forma automática los atributos demográficos de una población objetivo de usuarios, a partir de la información que éstos generan o comparten. Estos atributos pueden ser aprovechados por distintas organizaciones y compañías para propósitos de seguridad, mercadotecnia, educación, estadísticas poblacionales, entre otros. En este artículo se presenta un estudio sobre el análisis de los mensajes de texto publicados por celebridades de redes sociales (usuarios populares), para con base en ello predecir el perfil demográfico de tales usuarios, formado por su género, ocupación y año de nacimiento. Para la tarea se utiliza un conjunto de datos de 1,920 celebridades de Twitter, formado por 5,066,608 tweets principalmente en inglés. A partir de estos datos se realizaron experimentos extrayendo una serie de características textuales de los tweets y con ellas se construyeron diversos modelos de aprendizaje de máquina. Se realizó una evaluación del uso de características y modelos siguiendo una validación cruzada estratificada de 10 partes y se midió el área bajo la curva ROC. Los resultados indican que algunos atributos como el año de nacimiento son complicados de predecir. Se observa de igual forma, que características como los vectores de palabras fastText presentan buen desempeño sobre todo en combinación con modelos de aprendizaje discriminativos.

Palabras clave: Perfilado de autor, minería de datos, aprendizaje de máquina, redes sociales.

Demographic Profiling of Celebrities in Social Networks

Abstract. Author profiling in social media is a task that has become popular in recent years to automatically predict the demographic attributes from a population of users, based on the information they generate and share. These attributes can be exploited by different organizations and companies for purposes of security, marketing, education, population statistics, among others. This article

presents a study on the analysis of text messages posted by social network celebrities (popular users), to predict the demographic profile of these users, conformed by their gender, occupation and year of birth. A dataset of 5,066,608 tweets, mainly in English, by 1,920 Twitter celebrities is used for the task. From this data, experiments were conducted by extracting a series of textual features from the tweets and with these features various machine learning models were built. An evaluation of the different features and models was performed following a stratified 10-fold cross-validation, measuring their performance with the area under the ROC. The results indicate that some attributes such as year of birth are difficult to predict. It is also observed, that features such as fastText word vectors perform well, especially in combination with discriminative learning models.

Keywords: Author profiling, data mining, machine learning, social networks.

1. Introducción

El perfilado de autor se entiende como el análisis del contenido generado o compartido por un usuario con la finalidad de predecir de forma automática atributos demográficos que caractericen a ese usuario, tales como su edad, género, ocupación [20], rasgos de personalidad [9], nivel educativo, orientación política [2], entre otros.

Esta tarea ha tomado relevancia en los últimos años gracias a la abundante información que las personas generan y comparten en distintos medios a través de internet.

Uno de los medios más populares donde las personas crean y comparten información es en las redes sociales, que cuentan con millones de usuarios que todos los días expresan sus gustos, opiniones e ideas a través de publicaciones que contienen información en varias modalidades, como texto, imágenes y video.

El perfilado de autor en redes sociales tiene distintas aplicaciones, ya que permite sectorizar a los usuarios por grupos dependiendo de sus atributos demográficos. Con esta sectorización, distintas empresas y organizaciones pueden ajustar el contenido y las herramientas que proveen a los usuarios con fines de mercadotecnia, promoción política, programas sociales, información educativa, entretenimiento, entre otros.

Por ejemplo, en la mercadotecnia puede apoyar a las empresas para realizar campañas de productos para usuarios con características específicas. Adicionalmente, con el perfilado de autor se puede lograr una identificación primaria de usuarios que tienen un comportamiento anómalo (acoso, hostigamiento, intento de robo de información, terrorismo) dentro de las redes sociales y cuya información demográfica está oculta, esto con propósitos de seguridad.

En el presente artículo se realiza un estudio sobre el perfilado demográfico de celebridades de redes sociales. Una celebridad se considera un usuario de la red que tiene un número considerables de seguidores dentro de la misma. La tarea consiste en analizar los mensajes de texto publicados o compartidos por la celebridad y con base en ello predecir los atributos demográficos de género, ocupación y año de nacimiento.

Para conducir el estudio, se utilizó el conjunto de datos de entrenamiento publicado en PAN@CLEF 2020[1] que está formado por los tweets de 1,920 celebridades.

En este conjunto, un usuario se considera celebridad si tiene al menos 10 seguidores. Para este trabajo, del conjunto de datos original se filtraron los tweets que utilizaran un alfabeto no occidental, quedando un total de 5,066,608 tweets, con un promedio de 2,639 tweets por celebridad.

Los tweets en su mayoría se encuentran en inglés, con algunos en otros idiomas como el español. En el conjunto de datos, las celebridades están clasificadas en dos géneros (hombre, mujer), cuatro ocupaciones (político, creador, artista, deportista) y en 60 años de nacimiento (entre 1940 y 1999).

Partiendo de estos datos, a los tweets de las celebridades se les extrajeron las siguientes características textuales: palabras, emoticones/emojis, etiquetas (# o hashtags), menciones (@ o ats), abreviaturas y los vectores de palabras fastText. Cada una de estas características revela diferentes aspectos del contenido que generan o comparten los usuarios.

Empleando las características extraídas se construyeron modelos de aprendizaje de máquina para realizar la predicción de los atributos demográficos. Se entrenaron y probaron los modelos de clasificadores multinomiales simples de Bayes (MNB o Multinomial Naïve Bayes), k vecinos más cercanos (KNN o K-Nearest Neighbors), bosques aleatorios (RF o Random Forest), regresión logística (LR o Logistic Regression) y máquinas de vectores de soporte lineales (LSVM o Linear Support Vector Machines).

Para el estudio, se experimentó con las combinaciones de modelos de aprendizaje y características utilizando una validación cruzada estratificada de 10 partes, con el fin de obtener resultados consistentes estadísticamente. El desempeño de cada combinación se midió utilizando la métrica del área bajo la curva ROC (AUC), que es una métrica popular en clasificación de textos, principalmente cuando se tienen clases desbalanceadas (donde algunas clases tienen mayor cantidad de ejemplos de entrenamiento que otras).

La contribución de nuestro trabajo radica en el estudio del desempeño de diferentes características textuales y modelos de aprendizaje para la tarea de perfilado demográfico de celebridades de redes sociales, intentando responde las siguientes preguntas de investigación: 1) ¿Hay un modelo de aprendizaje de máquina con mejor desempeño? 2) ¿Hay una característica textual con un mejor desempeño? 3) ¿Hay una combinación de modelo de aprendizaje y característica textual con un mejor desempeño?

El resto del presente artículo se organiza de la siguiente manera. La sección 2 presenta una revisión de los trabajos relacionados encontrados en la literatura. La sección 3 explica la metodología utilizada para el estudio, incluyendo la descripción del conjunto de datos y los detalles de la experimentación. La sección 4 muestra los resultados obtenidos con el estudio de modelos de aprendizaje y características textuales. Finalmente, la sección 5 presenta las conclusiones y algunas ideas para trabajos futuros.

[1] Disponible en: https://pan.webis.de/clef20/pan20-web/celebrity-profiling.html

2. Trabajos relacionados

El estudio de perfilado de autor en redes sociales utilizando el contenido textual que generan los usuarios, se ha abordado a lo largo de los años siguiendo diferentes enfoques.

Dentro de los atributos demográficos que se han extraído para la tarea de perfilado se incluyen la edad, el género, la ocupación, el nivel socioeconómico, entre otros; siendo la predicción de edad y género los atributos más populares para determinar [3]. Sin embargo, otras subtareas como la identificación de rasgos de personalidad [9] u ocupación [20], también han cobrado relevancia en años recientes.

Uno de los principales eventos donde se han presentado investigaciones sobre el estudio de perfilado de autor en redes sociales es en las conferencias de PAN[2]. PAN forma parte de CLEF (Conference and Labs of Evaluation Forum), en donde desde el 2013 se realiza anualmente la tarea de perfilado de autor para la predicción de edad, género, idioma nativo, ocupación y rasgos de personalidad [14, 12, 16, 17, 15, 13].

En estas conferencias se han utilizado diversos conjuntos de datos extraídos de Twitter, los cuales contienen el texto de las publicaciones generadas por los usuarios. Los conjuntos de datos se han conformado principalmente por publicaciones en inglés, aunque también se han agregado otros idiomas como el español, el portugués, el italiano, el neerlandés y el árabe.

A través de las ediciones de PAN@CLEF se han presentado una diversidad de trabajos que han hecho uso de diferentes enfoques para la tarea de perfilado de autor. Se han utilizado diferentes características textuales como palabras, emoticonos/emojis [7], bolsa de palabras (bag-of-words), n-gramas, diccionario de palabras, vectores de palabras, entre otras.

De igual manera, se han utilizado diferentes modelos de aprendizaje de máquina como máquinas de vectores de soporte, regresión logística, clasificadores bayesianos y modelos de aprendizaje profundo (Deep Learning).

Recientemente, en las conferencias de PAN@CLEF se ha presentado el estudio de perfilado de celebridades. Considerando a una celebridad como un usuario de una red social que tiene un número determinado de seguidores. El objetivo es la predicción de variables demográficas como el género, edad, ocupación y grado de fama utilizando el contenido generado en Twitter [19] por la celebridad o por sus seguidores [20].

Para el perfilado de celebridades utilizando el contenido generado por las mismas, en [11] utilizaron máquinas de vectores de soporte y regresión logística para la predicción de ocupación, edad y género. Los autores en [10] utilizaron un modelo de regresión logística para predecir la edad, género y grado de fama, mientras que para predecir la ocupación utilizaron un modelo multimodal simple de Bayes.

De igual manera, utilizaron un número promedio de palabras por tweet, emojis, longitud de palabras, hashtags, hipervínculos, menciones, entre otra. En [8], los autores emplearon vectores tf-idf (term-document frequencye inverse document frequency) formados a partir de unigramas de palabras, así como también trigramas de caracteres delimitados por palabras.

[2] https://pan.webis.de/

Los autores usaron clasificadores como máquinas de vectores de soporte con kernels lineales y RBF, regresión logística, bosques aleatorios, y clasificadores de potenciación de gradiente.

En cuanto al perfilado de celebridades utilizando el contenido generado por sus seguidores, los autores en [1] usaron una matriz de tf-idf generada a partir del contenido textual generado por los seguidores.

Esta matriz se introdujo en una red neuronal LSTM para la predicción. Los autores en [5] utilizaron características como el promedio de todos los vectores de palabras de los tweets de los seguidores, palabras vacías (stopwords), hashtags, emojis, menciones y links.

Utilizaron modelos de regresión logística, máquinas de vectores de soporte y bosques aleatorios. Por otro lado, en [6], los autores utilizaron representaciones léxicas en conjunto con clasificadores de regresión logística para la predicción de la edad y ocupación, mientras que para la predicción del género usan un modelo de máquinas de vectores de soporte.

3. Metodología

La metodología se encuentra conformada por tres fases que son la adquisición de los datos, el procesamiento de los datos y la experimentación. En la última fase se describen los procesos de la construcción de modelos y la evaluación de estos mismos. Las tres fases se encuentran descritas a continuación.

3.1. Adquisición de los datos

En este artículo se utilizó el conjunto de datos de la conferencia PAN@CLEF 2020 para la tarea de celebrity profiling[3], el cual se extrajo directamente de Twitter por los organizadores de la conferencia. Este conjunto de datos está conformado por el contenido textual de las publicaciones realizadas por 1,920 celebridades.

Del conjunto original se eliminaron aquellas publicaciones con un alfabeto diferente al occidental, quedando un total de 5,066,608 tweets, para un promedio de 2,639 tweets por celebridad. Los tweets en su mayoría se encuentran en inglés, con algunos en otros idiomas como el español. Las celebridades se encuentran etiquetadas con tres atributos demográficos: género (hombre y mujer), año de nacimiento (entre 1940 y 1999) y ocupación (político, creador, artista y deportista).

En la Tabla 1 se observa la distribución de usuarios por género y ocupación. Como se puede ver en la tabla, el número de usuarios hombres (56 %) es ligeramente mayor al número de usuarios mujeres (44 %).

Esta distribución de género refleja en cierta medida la que se encuentra en Twitter, donde el 68.5 % de los usuarios son hombres[4]. Por otro lado, se observa una distribución más homogénea para cada una de las clases del atributo ocupación.

Por motivos de ilustración, se agruparon los años de nacimiento en décadas, y su distribución con respecto al género se muestra en la Tabla 2.

[3] Disponible en: https://pan.webis.de/clef20/pan20-web/celebrity-profiling.html

[4] https://bit.ly/2QqCiRs

Tabla 1. Distribución de usuarios por género y ocupación.

Género	Político	Creador	Artista	Deportista	Total
Mujer	128	240	240	240	848
Hombre	352	240	240	240	1072
Total	480	480	480	480	1920

Tabla 2. Distribución de usuarios por género y década de nacimiento.

Género	1940s	1950s	1960s	1970s	1980s	1990s	Total
Mujer	20	64	119	217	285	143	848
Hombre	68	150	237	264	257	96	1072
Total	88	214	356	481	542	239	1920

En la tabla se observa un predominio de usuarios nacidos en los años 1980s, seguidos de los nacidos en los años 1970s. En la tarea de predicción, se considera el año exacto de nacimiento.

3.2. Procesamiento de datos

En esta paso se procesaron los tweets para extraer diferentes características textuales. Primero se concatenaron todos los tweets correspondientes a un usuario en una sola cadena de texto.

El proceso se aplicó a todos los usuarios, de tal forma que un usuario queda expresado como una cadena de larga de texto. Posteriormente, se emplearon una serie de expresiones regulares para extracción de cinco características textuales: palabras, emoticones/emojis, etiquetas (# o hashtags), menciones (@ o ats) y abreviaturas comunes. Para la palabras se realizó un proceso en donde se removieron aquellas palabras que eran muy cortas (longitud < 3), muy largas (longitud > 35) y palabras vacías (stopwords).

Para ello, se utilizó una lista de palabras vacías en inglés proporcionada por la librería NLTK. En el caso de las abreviaturas, se recopiló a través de internet una lista de las 1,374 abreviaturas más comunes en Twitter.

Al final del proceso limpieza, agrupamiento de información y extracción de características, se obtuvieron cinco archivos. Cada archivo contenía 1,920 líneas; siendo cada una de estás líneas las características de una celebridad. Utilizando una validación cruzada estratificada a 10 partes, se dividió cada archivo en 10 conjuntos de entrenamiento y 10 conjuntos de prueba.

Para cada una de las características textuales se extrajo un vocabulario (características únicas).

En la Tabla 3 se muestran los tamaños de cada vocabulario para el conjunto de datos completo. En esta tabla se observa que las características con vocabularios más extensos son las menciones y las palabras; mientras que las abreviaturas tienen el vocabulario más pequeño.

Utilizando el vocabulario correspondiente de cada una de las características, se realizó un proceso de vectorización con el método tf-idf (term frequency inverse document frequency), el cual se encuentra definido por la ecuación 1:

$$tfidf(t,d) = tf(t,d) \times idf(t), \tag{1}$$

Tabla 3. Tamaño del vocabulario por característica.

Palabras	Emoticones	Etiquetas	Menciones	Abreviaturas
662,308	1,334	407,959	1,068,617	864

donde $tf(t, d)$ es la frecuencia en la que ocurre la término t en el documento d, y término idf se encuentra definido por la ecuación 2:

$$idf(t) = \log \frac{1 + n_d}{1 + df(d, t)} + 1. \tag{2}$$

En la ecuación 2, $df(d, t)$ es el número de documentos d que contienen al término t, el término n_d es el número total de documentos. Para cada conjunto de entrenamiento de cada característica se calculó el término idf el cual sería utilizado para la vectorización del conjunto de prueba.

Adicionalmente, se construyeron matrices usando como características los vectores de palabras fastText. El modelo fastText mide estadísticas de coocurrencia entre palabras a partir de un conjunto de datos de entrenamiento.

Para este trabajo se utilizó un modelo preentrenado sobre un conjunto de datos en inglés de Wikipedia y Common Crawl[5], el cual contiene un diccionario de más de 2 millones de palabras, cada una representada con un vector de 300 características. Para las características de vectores fastText se calculó el vector promedio de todos los vectores de palabras encontradas en los tweets de un usuario.

De esta manera, cada celebridad se presenta como un vector promedio de 300 características densas. El proceso de vectorización se aplico a todos los usuarios en cada conjunto de entrenamiento y prueba.

3.3. Experimentación

Al terminar el proceso de vectorización, se realizó una experimentación con diferentes modelos de aprendizaje de máquina. Los modelos que se utilizaron siguen varios enfoques: probabilístico (clasificador simple de Bayes o MNB), basado en instancias (k vecinos más cercanos o KNN), reglas de decisión (bosques aleatorio o RF), y discriminativos (máquinas de vectores de soporte lineales o LSVM, y regresión logística o LR).

Se decidió utilizar estos modelos ya que, como ha sido mencionado por otros autores en la tarea de perfilado de autor utilizando información textual, los modelos basado en aprendizaje profundo no han logrado mejorar el desempeño de modelos más tradicionales [18, 5].

Con cada uno de los modelos se aplicó la validación cruzada estratificada de 10 partes, para que los resultados obtenidos fueran sólidos estadísticamente.

Con los modelos LSVM, LR, RF y KNN, se realizó una subvalidación cruzada de 3 partes para cada conjunto de entrenamiento, con el fin de encontrar los valores óptimos para sus hiperparámetros. En la Tabla 4 se pueden observar los diferentes valores que se consideraron en la optimización del hiperparámetro de cada modelo.

Una vez encontrado el valor óptimo, se construye el modelo final con ese valor y con todo el conjunto de entrenamiento.

[5] Disponible en: https://fasttext.cc/docs/en/supervised-models.html

Tabla 4. Valores considerados para los hiperparámetros.

Modelo	Parámetro	Descripción	Valores
KNN	k	Número de vecinos	[1, 2, 3, 5, 10]
RF	r	Número de árboles	[5, 10, 15, 20]
LR	c	Parámetro de regularización	[0.1, 1, 10, 100]
LSVM	c	Parámetro de regularización	[0.1, 1, 10, 100]

Para medir el desempeño de los modelos, se utilizó una métrica basada en la matriz de confusión, formada por las celdas de: verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos.

Esta matriz muestra la relación entre las clases reales de los usuarios contra las clases predichas por los modelos.

La métrica utilizada para evaluar a cada uno de los modelos fue el área bajo la curva ROC (Receiver Operating Characteristic). La curva ROC grafica la razón de verdaderos positivos contra la razón de falsos positivos en varios umbrales.

El área bajo la curva ROC (AUC o Area Under the Curve) evalúa el grado de separabilidad, midiendo la probabilidad de que un modelo clasifique a un usuario elegido aleatoriamente en una clase, más que a un usuario de otra clase elegido aleatoriamente. Esta métrica es particularmente útil cuando la distribución entre clases no es uniforme, como es el caso de los atributos género y año de nacimiento.

Como base de comparación se consideran dos modelos. El primero es uno totalitario, el cual asignaría todos los usuarios del conjunto de prueba a la clase mayoritaria. El segundo es uno aleatorio uniforme, el cual asignaría un usuario a una clase aleatoria con la misma probabilidad para todas. Para ambos modelos la métrica AUC sería de 0.5.

Todos los códigos para el procesamiento y experimentación se realizaron en Python utilizando las librerías NLTK, emoji, scikit-learn y fasttext. El código está disponible en el siguiente repositorio https://github.com/jcgcarranza/rcs_celebrity_profiling. Los datos procesados como fueron usados en este artículo están disponibles en https://zenodo.org/record/4767751.

4. Resultados

En las tablas 5, 6 y 7 se muestran los resultados obtenidos por las distintas características textuales extraídas y los diferentes modelos de aprendizaje utilizados para la predicción de los atributos de género, ocupación y año de nacimiento, respectivamente.

En las tablas, los renglones 3 a 7 indican los modelos de aprendizaje probados: MNB, KNN, RF, LR, LSVM. Las columnas 2 a 7 indican las características textuales extraídas de las publicaciones para construir y probar los modelos: palabras, emojis/emoticones, etiquetas, menciones, abreviaturas y los vectores de palabras fastText.

Las celdas muestran el promedio de la métrica AUC para el uso de un modelo con una característica siguiendo la validación cruzada, con la desviación estándar entre paréntesis.

Tabla 5. Resultados (AUC) para género.

	Característica						
Modelo	**Palabras**	**Emoticones**	**Etiquetas**	**Menciones**	**Abreviaturas**	**fastText**	**Promedio**
MNB	0.71(0.37)	0.76(0.38)	0.71(0.36)	0.64(0.34)	0.57(0.31)	–	0.68
KNN	0.75(0.36)	0.72(0.37)	0.69(0.35)	0.70(0.36)	0.68(0.35)	0.77(0.39)	0.72
RF	0.77(0.39)	0.77(0.39)	0.72(0.37)	0.72(0.37)	0.74(0.38)	0.77(0.39)	0.75
LR	**0.88(0.44)**	0.79(0.40)	0.71(0.37)	0.68(0.36)	0.78(0.40)	**0.88(0.44)**	**0.79**
LSVM	**0.88(0.44)**	0.78(0.40)	0.71(0.37)	0.69(0.36)	0.76(0.39)	**0.88(0.44)**	0.78
Promedio	0.80	0.76	0.71	0.69	0.71	**0.83**	–

El renglón 8 muestra el promedio de la métrica de forma transversal para todos los modelos por característica. De forma similar, la columna 8 muestra el promedio de la métrica de forma transversal para todas las características por modelo.

En lo que respecta al género, se observa que tanto la combinación de las palabras con los modelos LR y LSVM, como la combinación de los vectores fastText con los mismos modelos, producen ambos resultados similares, alcanzando un 0.88 en AUC, siendo el resultado más alto para la predicción de este atributo y 38 % más alto que la base de comparación.

Si se revisan los promedios generales por modelo, se observa que LR es el que presenta el mejor desempeño con un 0.79, considerando el uso transversal de las características; aunque LSVM tiene un comportamiento similar. En cuanto a los promedios generales de características, se observa que fastText presenta el mejor resultado a lo largo del uso de distintos modelos con un 0.83.

Considerando que los vectores fastText producen una representación más pequeña que el uso de palabras, y por lo tanto un tiempo de entrenamiento y prueba más rápido, se puede considerar a esta característica como más adecuada para predecir el género de las celebridades. En cuanto al modelo de aprendizaje, LR usa el mismo principio que LSVM pero su tiempo de entrenamiento es menor, por lo que se puede considerar más adecuado para predecir el género de las celebridades. Por otro lado, dada la escala de los valores obtenidos en la predicción de este atributo, el atributo no es tan difícil de predecir, pero hay margen para mejorar.

Se puede especular que los errores en este atributo pueden deberse a la superposición de palabras entre géneros. Es decir, que las celebridades de ambos géneros utilizan palabras similares con la misma frecuencia. Adicionalmente, también es posible que el desbalanceo entre las clases del atributo tenga una afectación negativa en el desempeño.

Analizando los resultados para la ocupación, se observa que los valores más altos de desempeño se obtienen con la combinación de palabras o vectores fastText con los modelos LR o LSVM. Todas estas combinaciones producen un valor de 0.94 para AUC, siendo 44 % más alto que la base de comparación.

Revisando los promedios generales por modelo, se determina que el valor más alto se obtiene con los modelos LR y LSVM, ambos obteniendo un desempeño de 0.89

Tabla 6. Resultados (AUC) para ocupación.

	Característica						
Modelo	**Palabras**	**Emoticones**	**Etiquetas**	**Menciones**	**Abreviaturas**	**fastText**	**Promedio**
MNB	0.91(0.38)	0.83(0.30)	0.89(0.35)	0.90(0.38)	0.80(0.31)	–	0.87
KNN	0.90(0.36)	0.76(0.26)	0.86(0.32)	0.90(0.37)	0.77(0.26)	0.91(0.37)	0.85
RF	0.89(0.35)	0.79(0.28)	0.84(0.32)	0.87(0.35)	0.83(0.30)	0.91(0.38)	0.86
LR	**0.94(0.40)**	0.84(0.31)	0.88(0.35)	0.90(0.38)	0.85(0.33)	**0.94(0.40)**	**0.89**
LSVM	**0.94(0.40)**	0.84(0.32)	0.89(0.35)	0.90(0.36)	0.85(0.33)	**0.94(0.34)**	**0.89**
Promedio	**0.92**	0.81	0.87	0.90	0.82	**0.92**	–

promediado de forma transversal a todas las características. En cuanto a los promedios generales por característica, tanto las palabras como los vectores fastText tienen el mejor desempeño con el uso de todos los modelos, con un desempeño de 0.92.

Por las mismas razones que con el género, se puede considerar a los vectores fastText y al modelo LR como las mejores opciones en cuanto a característica y modelo para predecir la ocupación de las celebridades.

En este caso, la escala de valores para la métrica es mayor que la del género, por lo que es un atributo más sencillo de predecir. Se puede especular que hay una mejor separación entre las distribuciones de palabras.

Es decir, que las celebridades de las diferentes ocupaciones utilizan palabras diferentes con diferentes frecuencias. Adicionalmente, el balanceo de los usuarios entre las clases de este atributo, afecta positivamente el desempeño.

En el caso de los resultados para el año de nacimiento, los valores de desempeño más altos se obtienen con la combinación de palabras o vectores fastText con el modelo LSVM, dando un valor de 0.69 para AUC, que es 19 % más alto que la base de comparación.

El promedio más alto para un modelo a lo largo de las características lo obtiene LSVM con 0.64; mientras que el promedio más alto para una característica a lo largo de los modelos lo obtiene tanto las palabras como los vectores fastText con 0.60.

De nueva cuenta, se puede considerar a los vectores fastText como la mejor característica para predecir el año de nacimiento de las celebridades, aunque en este caso el modelo recomendado es LSVM. Por las escalas de los valores de la métrica AUC, se observa que predecir el año de nacimiento es más complejo que los otros dos atributos.

Una razón importante es por el gran número de clases posibles (60 años/clases); considerando que para la clasificación de textos, en general, entre mayor número de clases existe, más complejo es el problema, como se ha observado en otros ámbitos [4].

Una segunda razón es que se tiene un desbalanceo entre clases más pronunciado que en los otros atributos, lo cual afecta en mayor medida el desempeño.

En la Tabla 8 se presenta el promedio del desempeño de las combinaciones de características y modelos para los tres atributos: género, ocupación y año de nacimiento. Las combinaciones que obtienen mejores resultados son el uso de las palabras o los

Tabla 7. Resultados (AUC) para año de nacimiento.

	Característica						
Modelo	**Palabras**	**Emoticones**	**Etiquetas**	**Menciones**	**Abreviaturas**	**fastText**	**Promedio**
MNB	0.59(0.02)	0.65(0.02)	0.60(0.02)	0.59(0.02)	0.59(0.02)	–	0.60
KNN	0.51(0.01)	0.52(0.01)	0.51(0.02)	0.51(0.03)	0.51(0.01)	0.51(0.02)	0.51
RF	0.54(0.01)	0.54(0.01)	0.52(0.01)	0.53(0.02)	0.56(0.02)	0.55(0.02)	0.54
LR	0.66(0.02)	0.61(0.01)	0.57(0.02)	0.60(0.02)	0.63(0.03)	0.66(0.02)	0.62
LSVM	**0.69(0.01)**	0.68(0.02)	0.59(0.02)	0.57(0.02)	0.66(0.02)	**0.69(0.02)**	**0.64**
Promedio	**0.60**	0.60	0.56	0.56	0.59	**0.60**	–

Tabla 8. Promedio de resultados para los tres atributos (AUC).

	Característica						
Modelo	Palabras	Emoticones	Etiquetas	Menciones	Abreviaturas	fastText	**Promedio**
MNB	0.74	0.75	0.74	0.71	0.66	–	0.72
KNN	0.72	0.67	0.69	0.74	0.66	0.73	0.70
RF	0.74	0.70	0.70	0.71	0.71	0.74	0.71
LR	0.82	0.75	0.72	0.73	0.76	0.83	**0.77**
LSVM	**0.84**	0.77	0.73	0.72	0.76	**0.84**	**0.78**
Promedio	0.77	0.72	0.71	0.71	0.70	**0.78**	–

vectores fastText con el modelo LSVM alcanzando un valor de 0.84, un 34 % más alto que la base de comparación.

El mejor desempeño general de LSVM con respecto a LR se debe al desempeño de LSVM con el atributo de año de nacimiento, ya que en los otros dos atributos ambos modelos se comportan igual.

5. Conclusiones

En este artículo se desarrolló un estudio del comportamiento de distintas características textuales en combinación con diversos modelos de aprendizaje de máquina para la tarea de perfilado demográfico de celebridades en redes sociales.

Para esta tarea se analizó los mensajes de texto publicados o compartidos por una celebridad y con base en ellos se predijeron los atributos demográficos de género, ocupación y año de nacimiento.

Para ello se experimentó con un conjunto de 5,066,608 tweets, mayormente en inglés, correspondientes a 1,920 celebridades de Twitter. De acuerdo con los experimentos, para el predecir el perfil demográfico de celebridades de Twitter, se concluye lo siguiente:

– Los vectores de palabras fastText, como característica para representar el contenido textual de las celebridades, tienen el mejor desempeño para predecir los atributos demográficos de éstas, tanto de forma individual como en su desempeño agregado.
– Otras características textuales como las palabras también muestran un buen desempeño en la predicción; sin embargo, su uso implica una representación más extensa que consume más memoria, y requiere de un mayor tiempo de entrenamiento y prueba de los modelos de aprendizaje.
– El resto de características textuales, emoticones, etiquetas, menciones y abreviaturas, presentan un desempeño moderado. Es de resaltar el uso de las abreviaturas, que con un vocabulario tan pequeño mantienen un comportamiento aceptable, con valores entre 3 % a 10 % abajo de los mejores resultados.
– Los modelos de aprendizaje que siguen un enfoque discriminativo, LR y LSVM, tienen el mejor desempeño para predecir los atributos de género y ocupación para las celebridades; mientras que el modelo LSVM tiene el mejor desempeño para predecir el año de nacimiento. De forma agregada, el modelo LSVM es el que tiene el mejor desempeño. No obstante, el modelo LR presenta mejores tiempos de entrenamiento y prueba, por lo que para los atributos de género y ocupación sería recomendable su uso.

Algunas ideas por explorar para trabajos futuros incluyen el uso de otros modelos de clasificación como las redes neuronales profundas, las cuales pueden funcionar adecuadamente con características densas como los vectores fastText. También se puede considerar el uso de características estilísticas, como las partes del discurso, o las frecuencias de palabras funcionales, puntuaciones o errores gramaticales.

Por último, sería interesante explorar el uso de métodos de extracción de características latentes tales como Latent Dirichlet Allocation, Latent Semantic Indexing, Principal Component Analasis, Biased Discriminant Analysis y Non Negative Matrix Factorization, los cuales se encargan de calcular asociaciones entre palabras para agruparlas en tópicos o temas.

Referencias

1. Alroobaea, R., Almulihi, A. H., Alharithi, F. S., Mechti, S., Krichen, M., Belguith, L. H.: A deep learning model to predict gender, age and occupation of the celebrities based on tweets followers. In: CLEF (Working Notes) (2020)
2. Cohen, R., Ruths, D.: Classifying political orientation on twitter: It's not easy! In: Proceedings of the International AAAI Conference on Web and Social Media (2013)
3. Garcia-Guzman, R., Andrade-Ambriz, Y. A., Ibarra-Manzano, M. A., Ledesma, S., Gomez, J. C., Almanza-Ojeda, D. L.: Trend-based categories recommendations and age-gender prediction for pinterest and twitter users. Applied Sciences, vol. 10, no. 17, pp. 5957 (2020)
4. Gomez, J. C.: Analysis of the effect of data properties in automated patent classification. Scientometrics, vol. 121, no. 3, pp. 1239–1268 (2019)
5. Hodge, A., Price, S.: Celebrity profiling using twitter follower feeds. In: Proceedings of the Working Notes of the Conference and Labs of the Evaluation Forum (2020)
6. Koloski, B., Pollak, S., Škrlj, B.: Know your neighbors: Efficient author profiling via follower tweets. In: CLEF (Working Notes) (2020)

7. López-Santamaría, L. M., Gomez, J. C., Almanza-Ojeda, D. L., Ibarra-Manzano, M. A.: Age and gender identification in unbalanced social media. In: 2019 International Conference on Electronics, Communications and Computers (CONIELECOMP), pp. 74–80 (2019)
8. Martinc, M., Skrlj, B., Pollak, S.: Who is hot and who is not? profiling celebs on twitter. In: Proceedings of the Working Notes of the Conference and Labs of the Evaluation Forum (2019)
9. Moreno, D. R. J., Gomez, J. C., Almanza Ojeda, D. L., Ibarra Manzano, M. A.: Prediction of personality traits in twitter users with latent features. In: Proceedings of the International Conference on Electronics, Communications and Computers (CONIELECOMP), pp. 176–181 (2019)
10. Moreno-Sandoval, L. G., Puertas, E., Plaza-del Arco, F. M., Pomares-Quimbaya, A., Alvarado-Valencia, J. A., Alfonso, L.: Celebrity profiling on twitter using sociolinguistic. In: CLEF (Working Notes) (2019)
11. Radivchev, V., Nikolov, A., Lambova, A., Cappellato, L., Ferro, N., Losada, D., Müller, H.: Celebrity profiling using TF-IDF, logistic regression, and SVM. In: CLEF (Working Notes) (2019)
12. Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daelemans, W.: Overview of the 2nd author profiling task at pan 2014. In: CEUR Workshop Proceedings, vol. 1180, pp. 898–927 (2014)
13. Rangel, F., Rosso, P., Montes-y Gómez, M., Potthast, M., Stein, B.: Overview of the 6th author profiling task at pan 2018: Multimodal gender identification in twitter. In: Proceedings of the Working Notes Papers of the CLEF, pp. 1–38 (2018)
14. Rangel, F., Rosso, P., Koppel, M., Stamatatos, E., Inches, G.: Overview of the author profiling task at pan 2013. In: Proceedings of the Conference on Multilingual and Multimodal Information Access Evaluation, pp. 352–365 (2013)
15. Rangel, F., Rosso, P., Potthast, M., Stein, B.: Overview of the 5th author profiling task at pan 2017: Gender and language variety identification in twitter. Working notes papers of the CLEF, pp. 1613–0073 (2017)
16. Rangel, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. In: Proceedings of the Conference and Lab of the Evaluation ForumEF, pp. 2015 (2015)
17. Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4th author profiling task at pan 2016: Cross-genre evaluations. Working Notes Papers of the CLEF, vol. 2016, pp. 750–784 (2016)
18. Takahashi, T., Tahara, T., Nagatani, K., Miura, Y., Taniguchi, T., Ohkuma, T.: Text and image synergy with feature cross technique for gender identification. In: Proceedings of the Working Notes Papers of the CLEF (2018)
19. Wiegmann, M., Stein, B., Potthast, M.: Overview of the celebrity profiling task at PAN 2019. In: Proceedings of the Working Notes of the Conference and Lab of the Evaluation Forum (Working Notes) (2019)
20. Wiegmann, M., Stein, B., Potthast, M.: Overview of the celebrity profiling task at pan 2020. In: Proceedings of the Conference and Lab of the Evaluation Forum (2020)

ISSN 1870-4069

Comparación entre métodos de alineamiento de múltiples secuencias para análisis filogenético de secuencias de ADN vaginales en R

Isaí Angulo-Jiménez, Juana Canul-Reich,
Betania Hernández-Ocaña

Universidad Juárez Autónoma de Tabasco,
División Académica de Ciencias y Tecnologías de la Información,
México

{juana.canul, betania.hernandez}@ujat.mx,
182H13001@egresados.ujat.mx

Resumen. En la presente investigación se documentan los resultados de aplicar métodos alineamiento de múltiples secuencias (MSA) a un conjunto de datos de secuencias con información perteneciente al microbioma de mujeres caucásicas. El objetivo fue realizar un flujo de trabajo intuitivo para análisis filogenético en R. Se definieron métodos MSA a través de una comparación entre los métodos: DECIPHER, ClustalW, ClustalOmega y MUSCLE, siendo ClustalW el más citado por la literatura. Estos resultados son importantes debido a que dentro de un mismo flujo de trabajo se hace uso de la implementación del algoritmo DADA2 para tareas de preprocesamiento y DECIPHER como herramienta MSA, logrando facilitar y simplificar las tareas para el usuario final, permitiendo realizar este tipo de tareas de manera práctica.

Palabras clave: Alineamiento de múltiples secuencias, microbioma, vaginosis bacteriana.

Multiple Sequence Alignment Comparison in R for Phylogenetic Analysis from Vaginal Sequence Data

Abstract. This research documents the results of applying multiple sequence alignment (MSA) methods to a sequence data set with information pertaining to the microbiome of Caucasian women. The objective was to create an intuitive workflow for phylogenetic analysis in R. MSA methods were defined through a comparison between the methods: DECIPHER, ClustalW, ClustalOmega and MUSCLE, with ClustalW being the most cited in the literature. These results are important because the implementation of the DADA2 algorithm for pre-processing tasks and DECIPHER as an MSA tool are used within the same workflow, thus facilitating and simplifying the tasks for the final user, allowing this type of analysis to be carried out in a practical way.

Keywords: Multiple sequence alignment, microbiome, bacterial vaginosis.

1. Introducción

El análisis computacional de los datos de secuencias a menudo implica el uso de distintos programas, códigos o herramientas que no necesariamente se encuentren relacionados entre sí, o pueden estar escritos en diferentes lenguajes de programación, diseñados para plataformas específicas, o en el peor de lo casos poseer una implementación poco intuitiva.

Las tareas de preprocesamiento, clasificación taxonómica, alineamiento de múltiples secuencias (MSA) y obtención de árboles filogenéticos emplean métodos matemáticos que están implementados en diferentes herramientas, lo que ocasiona que existan diversos formatos para su almacenamiento, haciendo tediosa la tarea del análisis.

R [11], a pesar de ser un entorno para estadística, permite análisis filogenético a través de paquetes e implementación de funciones, lo cual ayuda a que el manejo de secuencias y sus respectivos análisis puedan realizarse dentro de una sola plataforma. Una tarea central para el análisis filogenético es la obtención de un árbol de distancias entre las secuencias, para lo cual se requiere que las secuencias sean alineadas previamente [3].

Los MSA logran su objetivo mediante la programación dinámica, método que requiere tiempo y espacio en memoria de orden $N * M$, en donde N y M son el ancho de las secuencias a y b, respectivamente. Debido a la complejidad que implica el alineamiento de secuencias largas, heurísticas son utilizadas para acelerar el alineamiento sin impactar negativamente en la precisión.

Esta precisión varía en función del número de secuencias que son añadidas al alineamiento, pues puede ser que la identidad entre secuencias o similitud cada vez sea menor, lo cual implicaría la obtención de un alineamiento impreciso. Este problema se encuentra frecuentemente en muestras de secuencias donde hay gran diversidad bacteriana, como en el caso de la microbiota vaginal. De acuerdo con Ortiz-Rodriguez [10]:

"la vaginosis bacteriana, de origen polimicrobiano, es una alteración de la ecología vaginal donde la flora normal se ve prácticamente sustituida por gérmenes anaerobios. Muchos microorganismos han sido propuestos como causa de esta enfermedad, como la Gardnerella, Atopobium, Leptotrichia, Sneathia spp".

Han surgido trabajos que ejemplifican el uso de una sola herramienta para lograr un análisis filogenético, como por ejemplo Dadasnake, de Weißbecker et al. [15], el cual es un script en Python que hace uso del Divisive Amplicon Denoising Algorithm (DADA2) [2] para el preprocesamiento de secuencias y ClustalOmega como método de alineamiento, aunque no es su fin realizar un análisis filogenético.

Su uso está orientado a la ejecución en infreaestructuras de cómputo de alto rendimiento, las cuales cuentan con abundantes recursos de hardware, por lo que acceder a ellas en un principio podría implicar un problema al hacer estudios preliminares de este tipo.

De igual manera, Toparslan et al. [14] realizan un flujo de trabajo para secuencias de ADN mitocondriales escrito en su totalidad en R, pero no hay tareas de preprocesamiento debido a la naturaleza de las secuencias. Se hace uso del método de alineamiento ClustalW.

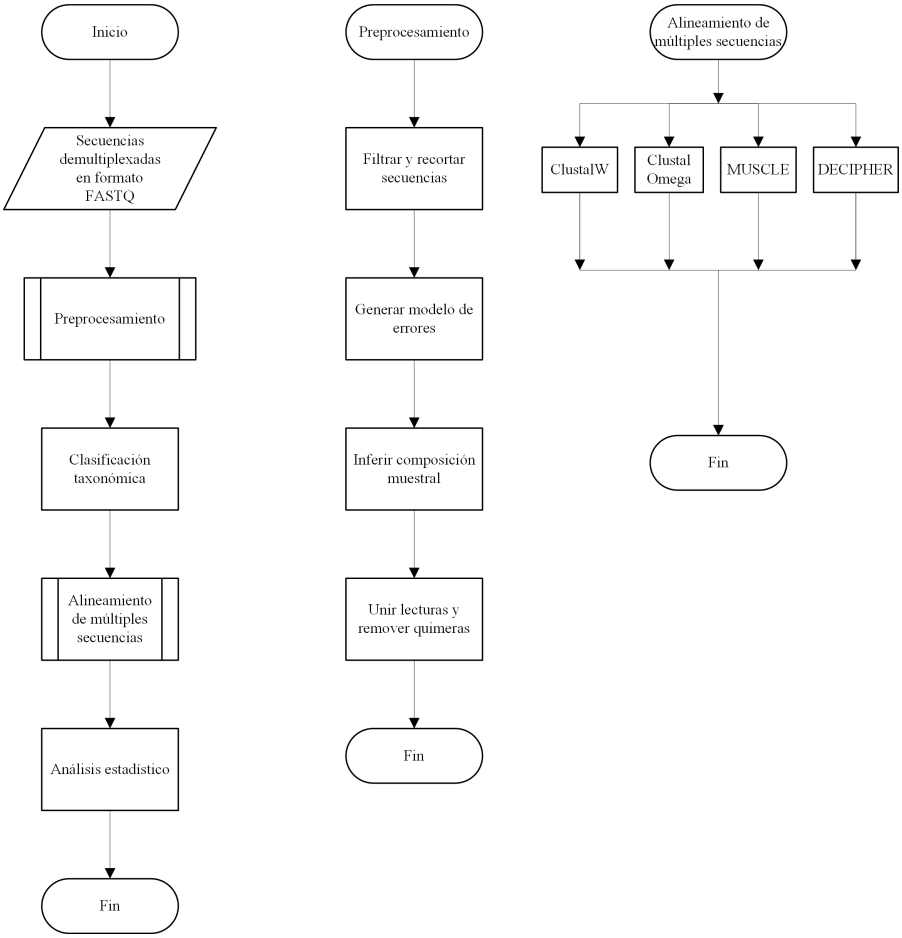


Fig. 1. Metodología.

El propósito de esta investigación es proveer un flujo de trabajo unificado en R que incluya implementaciones ya existentes para permitir tareas de análisis filogenético en una única plataforma, así como seleccionar métodos que puedan ser ejecutados en un equipo de cómputo personal. Para ello se hace una comparación entre implementaciones de los siguientes MSA: ClustalW, ClustalOmega, MUSCLE y DECIPHER.

2. Materiales y métodos

A continuación se detalla el conjunto de datos utilizado en el presente trabajo, así como conceptos relacionados con este artículo, como los distintos MSA utilizados. De igual manera se describen las medidas de rendimiento utilizadas para la evaluación estadística de los resultados obtenidos de cada alineamiento. Por último, se mencionan la paquetería necesaria para obtener los resultados de este artículo.

Tabla 1. Parámetros de entrada de los métodos MSA.

Clustal W			
Apertura de Hueco	Extensión de Hueco	Máximo de Iteraciones	Orden
15	6.66	3	Entrada
ClustalOmega			
Apertura de Hueco	Extensión de Hueco	Máximo de Iteraciones	Orden
6	1	Sin límite	Entrada
MUSCLE			
Apertura de Hueco	Extensión de Hueco	Máximo de Iteraciones	Orden
400	0	16	Entrada
DECIPHER			
Apertura de Hueco	Extensión de Hueco	Máximo de Iteraciones	Orden
16	1	2	Entrada

2.1. Conjunto de datos

Las secuencias pertenecen a un grupo de 155 mujeres caucásicas embarazadas. El microbioma de estas mujeres fue caracterizado mediante la secuenciación del gen 16S ARN ribosomal de las regiones V3-V4 en una plataforma MiSeq. La fuente de los datos es del European Nucleotide Archive (ENA), con el número de acceso PRJNA544732 y cargado por el Centro Médico Universitario de Liubliana, Eslovenia. Los datos obtenidos se encuentran comprimidos con la extensión FASTQ [5].

2.2. Métodos de alineamiento de múltiples secuencias

El problema de alinear secuencias está clasificado como un problema NP-completo [3]. Los métodos MSA se dividen en 4 tipos: exactos (programación dinámica), progresivos, basados en consistencia, e iterativos [6]. Los métodos ClustalW, ClustalOmega y MUSCLE se encuentran disponibles en R a través del paquete msa [1], mientras DECIPHER se encuentra en el paquete DECIPHER [17]. A continuación, cada método MSA utilizado:

ClustalW: Este método MSA basado en alineamiento progresivo inicia alineando pares de secuencias ya sea por el método k-tuple de Wilbur y Lipman o por el método de programación dinámica completo de Needleman-Wunsch, con los cuales obtienen medidas de distancia con las que se construye un árbol guía con el método Neighbour-Joining y por último alinea progesivamente las secuencias más cercanas acorde al árbol guía [13].

ClustalOmega: Este método MSA, también progresivo, se basa en un método de agrupamiento mBed para un alineamiento inicial, luego reagrupa por k-means. Utiliza UPGMA como método para construer el árbol guía y produce un alineamiento final alineando dos perfiles usando modelos ocultos de Markov (HMM) [12].

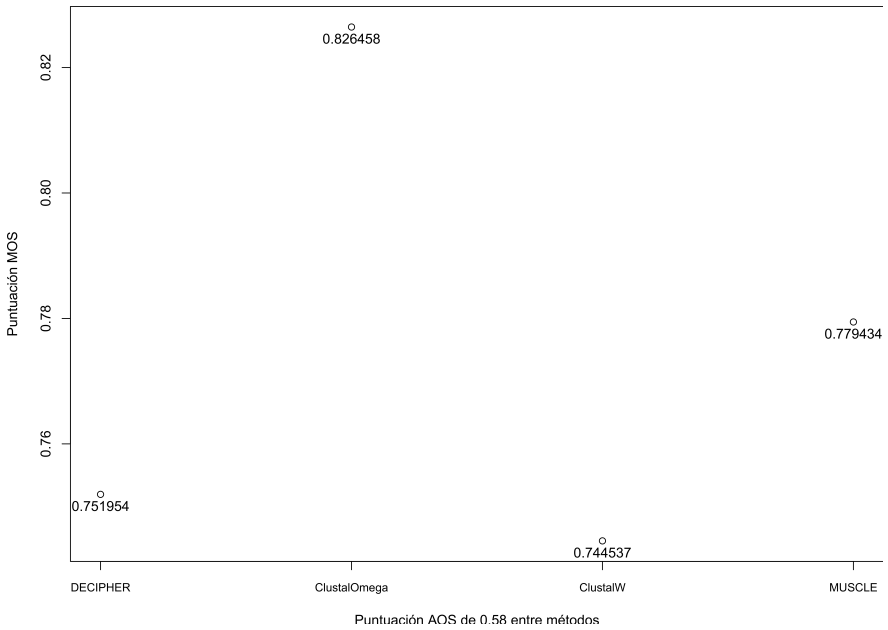


Fig. 2. Puntuaciones preliminares MOS y AOS para los cuatro métodos MSA.

MUSCLE: A diferencia de los otros tres métodos, este tiene una aproximación iterativa, en lugar de progresiva. Se reevalúan los alineamientos a través de dos distancias distintas: k-mer para pares de secuencias sin alinear y Kimura para las alineadas. Este procedimiento se repite siempre que se encuentre un alineamiento con una mejor puntuación que el anterior [4].

DECIPHER: Método MSA que toma en cuenta el contexto de las secuencias a través de la predicción de estructuras secundarias en el contexto de una secuencia local, incrementando la precisión del método. Esto permite la generación escalable de alineamientos de secuencias grandes manteniendo una precisión alta aún en conjuntos diversos de secuencias [16].

2.3. Medidas de puntuación

Las tareas tales como las búsquedas de homología entre secuencias, anotación genómica, predicción de la estructura de una proteína, así como áreas de biología evolutiva computacional, redes reguladoras de genes, y genómica funcional dependen del resultado de un método MSA.

El resultado obtenido de estas tareas bioinformáticas antes mencionadas tendrá una mayor significancia biológica a mayor precisión del resultado del MSA [7]. Sin embargo, debido a que no existe una función objetivo para medir verdaderamente la precisión o correctividad biológica de un alineamiento, existen métodos basados en distintas suposiciones.

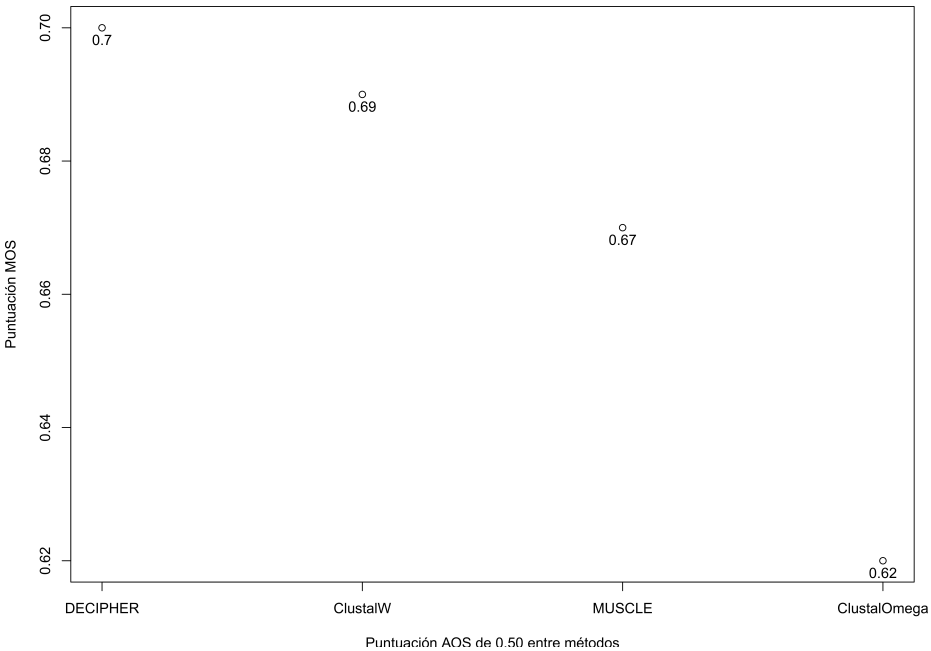


Fig. 3. Puntuaciones finales MOS y AOS para los cuatro métodos MSA.

La comparación cuantitativa de dos métodos MSA distintos ayuda a tomar decisiones sobre qué regiones están preservadas o cuáles deben ser removidas para tareas posteriores [8].

Coincidencia (OS): La función refleja la similitud entre dos alineamientos Q_a y Q_b, y está definida como la relación entre la cardinalidad de la intersección de dos conjuntos de residuos alineados y la cardinalidad promedio de cada conjunto:

$$Q_{ab} = \frac{|Q_a \cap Q_b|}{(|Q_a| + |Q_b|)/2}. \tag{1}$$

Coincidencia promedio (AOS): Cada alineamiento se representa mediante el concepto de residuos de pares alineados. Cada uno de estos pares son extraídos de todos los alineamientos m de entrada. La dificultad de un caso de alineamiento está definida por la puntuación de coincidencia promedio entre todos los alineamientos de entrada:

$$AOS = \frac{\sum_i^{m-1} \sum_{j=i-1}^{m} O_{ij}}{m(m-1)/2}. \tag{2}$$

Esta medida representa qué tan dispersos están los alineamientos en el espacio de todas las soluciones y se seleccionó como medida principal para decidir qué alineamiento utilizar.

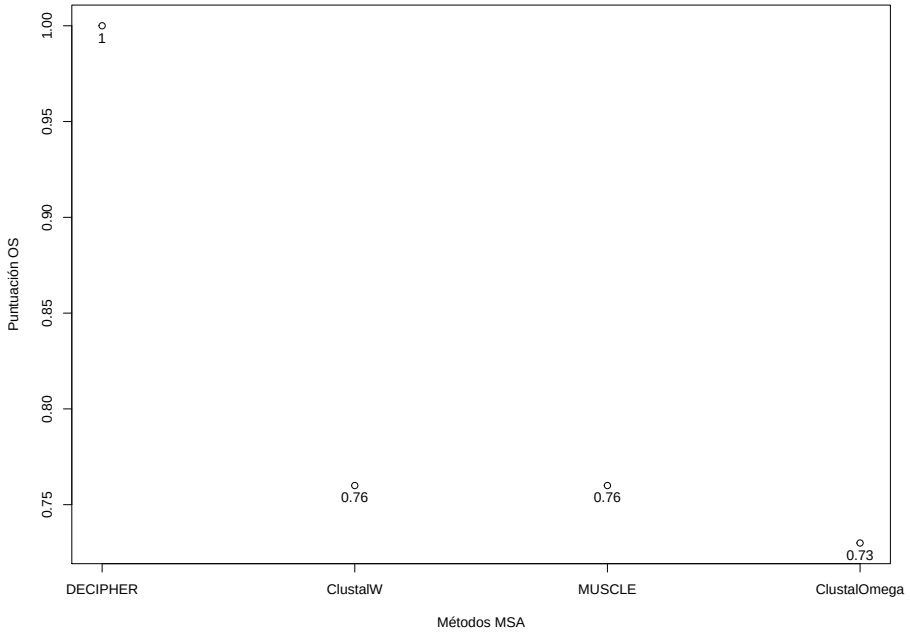


Fig. 4. Puntuación de coincidencia OS respecto al método DECIPHER.

Para casos simples, un método MSA dará como resultado alineamientos similares y el valor AOS será muy cercano a 1, mientras en casos difíciles su valor será cercano a 0.

Coincidencia múltiple (MOS): Se asignan puntuaciones a cada par de residuos alineados reflejando su proliferación en todos los alineamientos. Sea $n(\sigma)$ el número de los $m-1$ alineamientos que contienen σ.

Un par que ocurra en todos los alineamientos es, en consecuencia, asignado con la puntuación mayor $(m-1)$ mientras que un par que ocurre en un solo alineamiento es asignado con la puntuación menor de cero. Estas puntuaciones son sumadas para el alineamiento Q_a para determinar su puntuación de coincidencia múltiple:

$$MOS\left(Q_A\right) = \frac{\sum n(\sigma) : \sigma \in Q_a}{\left|Q_a\right|\left(m-1\right)}. \tag{3}$$

El numerador suma las puntuaciones de cada par de residuos alineados presentes en el alineamiento Q_a . El denominador refleja la puntuación máxima posible. Los residuos alineados que son encontrados en varios alineamientos son más confiables, y el alineamiento con el mayor número de tales pares se asume como el más significativo biológicamente.

Tabla 2. Tiempo de ejecución de funciones principales.

Preprocesamiento		
Proceso	**Tiempo estimado por tictoc en segundos (s)**	**Multihilo (verdadero o falso)**
Filtro y recorte para Cutadapt	504.795 s	Verdadero
Cutadapt	1098.482 s	Verdadero
Filtro y recorte al resultado de Cutadapt	446.305 s	Verdadero
Modelo de errores (forward)	38.051 s	Verdadero
Verdadero	195.136 s	Verdadero
Union de lecturas e inferencia muestral con DADA2	2637.742 s	Verdadero
Clasificación taxonómica		
Proceso	**Tiempo estimado por tictoc en segundos (s)**	**Multihilo (verdadero o falso)**
Asignación taxonómica	183.517 s	Verdadero
Asignación taxonómica	651.717 s	Falso
Metodos MSA		
Proceso	**Tiempo estimado por tictoc en segundos (s)**	**Multihilo (verdadero o falso)**
ClustalW	2908.898 s	Falso
ClustalOmega	100.779 s	Falso
MUSCLE	2558.235 s	Falso
DECIPHER	61.368 s	Verdadero
Validacion estadística		
Proceso	**Tiempo estimado por tictoc en segundos (s)**	**Multihilo (verdadero o falso)**
MUMSA AOS = 0.58	43.188 s	Falso
MUMSA AOS = 0.50	115.923 s	Falso
MUMSA coincidencia OS	44.139 s	Falso

2.4. Paquetería utilizada

Se usaron los programas externos a R Cutadapt en su versión 2.1 con Python 3.8.5 y MUMSA en su versión 1.0, pero se incluyeron dentro del flujo propuesto con llamadas a los programas por medio de R.

Además, los paquetes en R versión 4.0.5 con las respectivas versiones de los mismas fueron: msa 1.22, DECIPHER 2.18.1, dada2 1.18.0, gridExtra 2.3, phangorn 2.5.5, ShortRead 1.48.0, Biostrings 2.58.0, ggplot2 3.3.3, phyloseq 1.34.0, cluster 2.1.1, dendextend 1.14.0, así como tictoc 1.0, para la medición del tiempo de ejecución de tareas centrales en el flujo de trabajo. Los paquetes fueron instalados junto con las dependencias de cada uno.

3. Diseño experimental

En la Figura 1 se describe la metodología, la cual inicia con la obtención de las secuencias a utilizar, en este caso pertenecientes a muestras vaginales. Como parte inicial del preprocesamiento, con Cutadapt se removieron los primers o iniciadores con secuencias ambiguas en los primeros 17 nucleótidos de las lecturas forward y primeros 21 para las reverse. DADA2 fue aplicado a lecturas forward y reverse con una calidad mínima de 20 basada en una puntuación Phred.

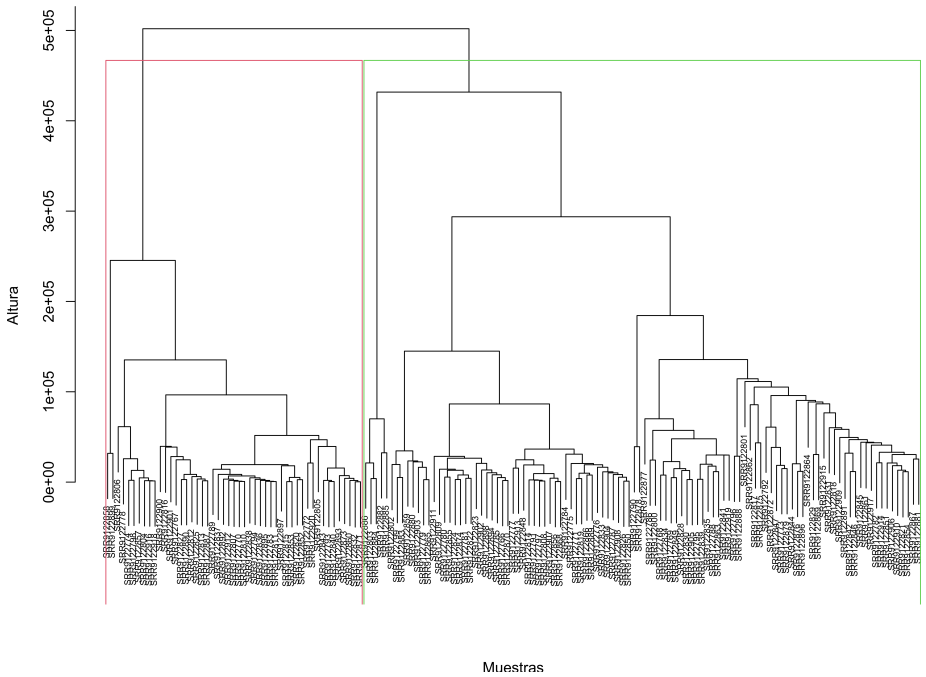


Fig. 5. Árbol jerárquico por el método Ward2 con $k = 2$.

La longitud mínima para las lecturas forward fue de 260 bases pareadas (bp) y para las lecturas reverse, 240 bp. Una remoción de secuencias quimeras fue necesaria posterior a la inferencia de variantes de secuencias (ASVs) a partir de la composición muestral, entendiendo por secuencias quimeras aquellas que se pueden construir exactamente mediante la combinación de segmentos izquierdo y derecho de dos secuencias "padre" más abundantes.

Este proceso dio como resultado 2,297 secuencias contenidas en las 155 muestras. Para la asignación taxonómica de las 2,297 secuencias, 1,974 fueron anotadas dentro del reino "Bacteria", 286 dentro de "Eukaryota", 1 dentro de "Archaea" y 36 sin anotar debido a que pertenecen al reino "Fungi", el cual no está presente en la base taxonómica utilizada. Debido al enfoque de la investigación referente a vaginosis bacteriana (VB), se trabajó solo con las 1,974 secuencias del reino "Bacteria".

Como principal parámetro de entrada se utilizaron las 1,974 secuencias obtenidas del preprocesamiento para los 4 MSA, cada método ejecutándose con sus parámetros por defecto, los cuáles se exponen en la Tabla 1.

Del resultado de cada método de alineamiento se obtuvo su respectivo árbol filogenético, el cual no requiere un análisis en esta etapa debido a que en etapas posteriores puede llegar a ser filtrado, aunque sí es requisito para la continuidad de este flujo de trabajo.

Con toda esta información obtenida, se procedió al uso de MUMSA para la validación estadística de los resultados obtenidos por los métodos MSA, con el fin de definir el método MSA a incluir en el flujo de trabajo.

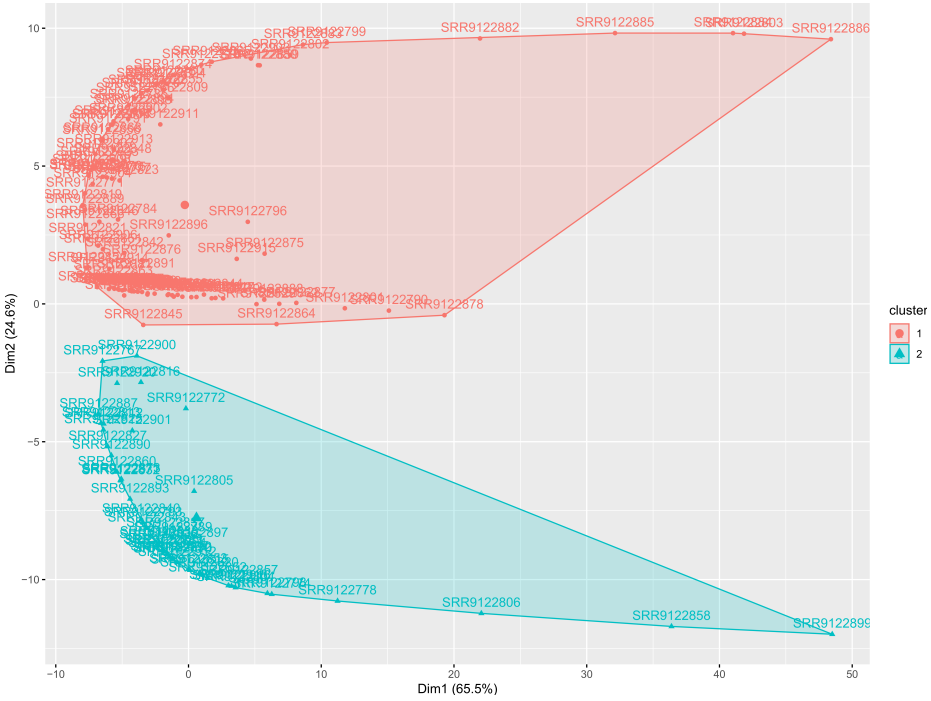


Fig. 6. Gráfico de dispersión de las 155 muestras analizadas con $k = 2$.

4. Resultados

R fue utilizado en su totalidad para la ejecución de las pruebas, aun sirviendo como interfaz para Cutadapt y MUMSA. Las características generales del equipo portátil tienen como SO Windows 10, procesador Intel Core i7-8750H de 12 núcleos a 2.208 GHz.

Para utilizar el total de núcleos en las tareas que así lo permitían se usó el Subsistema de Linux para Windows en su versión 2 con el SO Ubuntu 20.04 focal con kernel x86_64 Linux 5.4.72-microsoft-standard-WSL2 y un total de memoria RAM disponible de 25,562 GB.

Para todas las tareas realizadas se utilizó una sola semilla de valor 100, incluso para los métodos MSA. Una vez obtenida la salida de los cuatro métodos el primer uso de MUMSA fue predecir la dificultad del alineamiento partiendo de la coincidencia múltiple entre alineamientos y el resultado se observa en la Figura 2.

Esta primer medida no considera los residuos alineados a los huecos ni los pares de residuos alineados. Al tener como resultado un $AOS = 0.580479$, se está frente a un caso de alineamiento de dificultad cercana a la media, recordando el rango de dificultad $[0, 1]$, lo que supondría que el método ClustalOmega pareciese ser el indicado para cuestiones de este flujo de trabajo con un $MOS = 0.826458$, ya que supera al resto de los métodos.

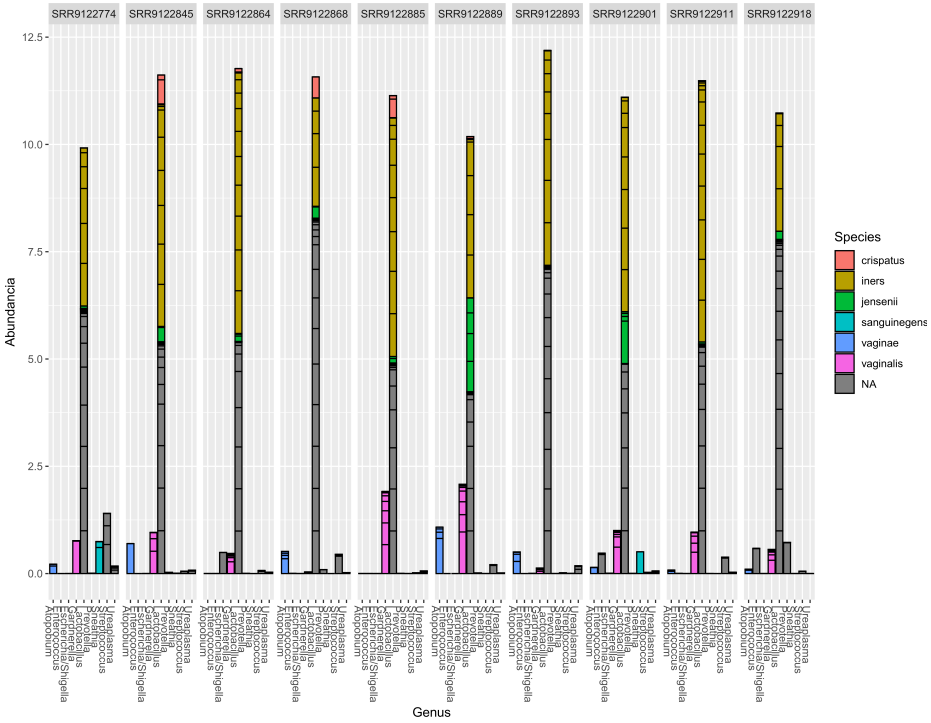


Fig. 7. Abundancia bacteriana de 10 muestras en rangos Genus y Species.

Sin embargo, al realizar la evaluación que sí toma las consideraciones de los residuos se obtiene como resultado un $AOS = 0.5$, indicando una dificultad mayor a la anteriormente obtenida, pero es el método DECIPHER con un valor $MOS = 0.7$ el que refleja una mayor confiabilidad estadística en los pares de residuos alineados a diferencia de los otros tres métodos (ver Figura 3).

Partiendo de la premisa de que el alineamiento con el mayor número de estos pares supone el biológicamente más significativo, se calculó la coincidencia OS entre métodos métodos respecto a DECIPHER, como se muestra en la Figura 4, en donde se aprecia que hay una diferencia importante en la intersección entre los demás métodos MSA con DECIPHER, que podría estudiarse a partir del número de columnas: ClustalW = 714, ClustalOmega = 908, MUSCLE = 2379, DECIPHER = 796, pero eso está fuera del enfoque de la presente investigación.

En la Tabla 2 se detalla el tiempo de ejecución de los principales procesos en el flujo de trabajo, así como la disponibilidad de multiprocesamiento de dicho proceso. Si solo se considera el tiempo aquí expuesto, el total sería de aproximadamente 11,588 s, poco más de 3 horas.

Habría que considerar que si se utiliza solo DECIPHER, método MSA más rápido y biológicamente más significativo para este análisis, el flujo de trabajo tomaría un tiempo de ejecución aproximado de 5,817 s, poco más de una hora y media.

Hasta esta etapa no se habían realizado exploraciones de la composición bacteriana a través de las muestras, ya que para esas tareas lo más conveniente es crear un objeto que pueda ser manipulable en R. El estudio de Hočevar et al. [5] posee información clínica de las muestras que son utilizadas para su investigación.

La aportación más grande del presente flujo de trabajo se centra en la posibilidad de llegar a resultados comparables en cuanto a inspección de la diversidad bacteriana de las muestras sin depender de datos clínicos, además de la posibilidad de aplicar un agrupamiento jerárquico a las 155 muestras. Todo esto con el propósito de verificar el alineamiento obtenido por el método DECIPHER.

En [5] se agrupan las 155 muestras en dos grupos: parto temprano con $n = 48$ y parto a término con $n = 107$. Debido a la ausencia de los datos clínicos propios de dicho estudio, se procedió a calcular un agrupamiento jerárquico aglomerativo (HC) mediante la función hclust usando Ward2 como método de agrupamiento, y el cálculo de valores disimilares mediante la función dist usando distancia euclideana.

Esto con el fin de saber si es posible aplicar técnicas de aprendizaje no supervisado a este tipo de secuencias en particular y que los resultados puedan ser analizados por los expertos. La semilla de valor 100 continuó siendo utilizada y al árbol jerárquico resultante se le hizo un recorte $k = 2$ para obtener dos grupos como se muestra en la Figura 5. También es posible su representación de dispersión de las muestras como en la Figura 6.

Los grupos se componen de 106 miembros el primero y 49 el segundo, lo que representa una aproximación muy cercana a lo obtenido por los datos clínicos. También se identificaron las bacterias a través de sus niveles taxonómicos. Como ejemplo, en la Figura 7 se muestran las bacterias más abundantes para 10 muestras obtenidas de un submuestreo del total de 155 con semilla de valor 100.

Los rangos comprendidos son Genus y Species en donde la mayoria de estas muestras están conformadas en gran parte por Lactobacillus, aunque también se aprecian muestras que podrían indicar VB debido a la alta concentración de Gardnerella, Atopobium o Sneathia además de una disminución de Lactobacillus.

5. Conclusiones y trabajos futuros

El fin del presente artículo fue comparar métodos MSA para así crear un flujo de trabajo intuitivo que permita un análisis filogenético en R e incluir validaciones estadísticas para tareas del análisis bioinformático.

Para ello se utilizaron herramientas ya desarrolladas para el SO Ubuntu, Cutadapt y MUMSA, y se llamaron mediante R para así complementar las tareas de preprocesamiento y validar estadísticamente los resultados de los alineamientos. El total de lecturas pareadas contenidas en las 155 muestras analizadas fue de 22,154,990.

Después de un primer filtrado con Cutadapt se obtuvieron 21,326,390, representando un 96.25 % del total. Estas fueron preprocesadas hasta llegar a 9,767,545, representando un 44.08 % del total. El número total de ASVs resultantes del preprocesamiento fue de 1,974 bacterias. Aun cuando los métodos MSA utilizados se basan en la programación dinámica, ClustalW, ClustalOmega y DECIPHER además hacen uso de alineamientos progresivos.

MUSCLE se basa en una aproximación iterativa. DECIPHER destacó del resto por la velocidad en la obtención del alineamiento, debido al uso de heurísticas al buscar k-mers de manera ordenada en la creación del árbol guía. Para este estudio se compararon los alineamientos obtenidos entre sí, una primera vez sin considerar residuos y una segunda vez tomándolos en cuenta.

ClustalOmega resultó el de mejor puntuación MOS para la primera, pero en la segunda, DECIPHER resultó el de mayor valor MOS. Debido a que se definió utilizar la puntuación AOS como parámetro, la segunda prueba tiene mayor significancia estadística, obteniendo un $AOS = 0.5$ contra el $AOS = 0.580479$ de la primera, implicando que existe una mayor dificultad en el alineamiento.

La segunda medida considerada fue la puntuación MOS, siendo DECIPHER el de mayor puntuación con un $MOS = 0.7$, lo cual indica una mayor fiabilidad biológica en el resultado del alineamiento. Así se tomó como referencia el resultado de DECIPHER y comparó contra el resto para obtener la tercer y última medida, OS, cuyos valores representan la coincidencia entre métodos respecto a DECIPHER y se visualiza en la Figura 4.

A manera de trabajo a futuro está la realización de una interfaz gráfica en R que permita el uso de este flujo de trabajo y sirva como herramienta práctica para el análisis filogenético de secuencias del gen 16S ARN ribosomal.

Referencias

1. Bodenhofer, U., Bonatesta, E., Horejš-Kainrath, C., Hochreiter, S.: MSA: An R package for multiple sequence alignment. Bioinformatics, pp. btv494 (2015) doi: 10.1093/bioinformatics/btv494
2. Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J. A., Holmes, S. P.: DADA2: High-resolution sample inference from Illumina amplicon data. Nature Methods, vol. 13, no. 7, pp. 581–583 (2016) doi: 10.1038/nmeth.3869
3. Daugelaite, J., O' Driscoll, A., Sleator, R. D.: An overview of multiple sequence alignments and cloud computing in bioinformatics. ISRN Biomathematics, vol. 2013, pp. 1–14 (2013) doi: 10.1155/2013/615630
4. Edgar, R. C.: MUSCLE: Multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Research, vol. 32, no. 5, pp. 1792–1797 (2004) doi: 10.1093/nar/gkh340
5. Hočevar, K., Maver, A., Vidmar Šimic, M., Hodžić, A., Haslberger, A., Premru Seršen, T., Peterlin, B.: Vaginal microbiome signature is associated with spontaneous preterm delivery. Frontiers in Medicine, vol. 6 (2019) doi: 10.3389/fmed.2019.00201
6. Issa, M., Hassanien, A. E.: Multiple sequence alignment optimization using meta-heuristic techniques. IHandbook of Research on Machine Learning Innovations and Trends, IGI Global, pp. 409–423 (2017) doi: 10.4018/978-1-5225-2229-4.ch018
7. Lecompte, O., Thompson, J. D., Plewniak, F., Thierry, J. C., Poch, O.: Multiple alignment of complete sequences (MACS) in the post-genomic era. Gene, vol. 270, no. 1–2, pp. 17–30 (2001) doi: 10.1016/s0378-1119(01)00461-9
8. Lassmann, T.: Automatic assessment of alignment quality. Nucleic Acids Research, vol. 33, no. 22, pp. 7120–7128 (2005) doi: 10.1093/nar/gki1020
9. Martin, M.: Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet.journal, vol. 17, no. 1, pp. 10 (2011) doi: 10.14806/ej.17.1.200

10. Ortiz-Rodríguez, C., Ley-Ng, M., Llorente-Acebo, C., Almanza-Martínez, C.: Vaginosis bacteriana en mujeres con leucorrea. Revista Cubana de Obstetricia y Ginecología, vol. 26, no. 2, pp. 74–81 (2000)
11. R Core Team: A language and environment for statistical computing. R Foundation for Statistical Computing www.R-project.org/
12. Sievers, F., Wilm, A., Dineen, D., Gibson, T. J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., Thompson, J. D., Higgins, D. G.: Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. Molecular Systems Biology, vol. 7, no. 1, pp. 539 (2011) doi: 10.1038/msb.2011.75
13. Thompson, J. D., Higgins, D. G., Gibson, T. J.: CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Research, vol. 22, no. 22, pp. 4673–4680 (1994) doi: 10.1093/nar/22.22.4673
14. Toparslan, E., Karabag, K., Bilge, U.: A workflow with R: Phylogenetic analyses and visualizations using mitochondrial cytochrome b gene sequences. PLOS ONE, vol. 15, no. 12, pp. e0243927 (2020) doi: 10.1371/journal.pone.0243927
15. Weißbecker, C., Schnabel, B., Heintz-Buschart, A.: Dadasnake, a Snakemake implementation of DADA2 to process amplicon sequencing data for microbial ecology. GigaScience, vol. 9, no. 12 (2020) doi: 10.1093/gigascience/giaa135
16. Wright, E. S.: DECIPHER: Harnessing local sequence context to improve protein multiple sequence alignment. BMC Bioinformatics, vol. 16, no. 1 (2015) doi: 10.1186/s12859-015-0749-z
17. Wright, E. S.: Using DECIPHER v2.0 to Analyze Big Biological Sequence Data in R. The R Journal, vol. 8, no. 1, pp. 352 (2016) doi: 10.32614/rj-2016-025

ISSN 1870-4069

Detección de bots en redes sociales usando técnicas procesamiento de lenguaje natural

Daniel Yacob-Espinosa[1], Helena Gómez-Adorno[2], Grigori Sidorov[1]

[1] Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico

[2] Universidad Nacional Autónoma de México, Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Mexico

espinosagonzalezdaniel@gmail.com, sidorov@cic.ipn.mx, helena.gomez@iimas.unam.mx

Resumen El uso de las redes sociales ha inundado nuestras vidas, no sólo para pasar el tiempo libre y la conexión entre las personas, sino también como una fuente de comunicación y difusión vía internet. Debido al aumento de usuarios y empresas que interactúan dentro de redes sociales, se han introducido ciertos programas para interactuar con los usuarios, mejor conocidos como: Bots. Los bots son programas los cuales pretenden tener un comportamiento humano en las redes sociales. En los últimos años, el uso de bots ha aumentado exponencialmente ya que pueden resolver ciertos problemas de manera rápida, pero también se han utilizado para con otros fines como: Alteración de información, creación de conflictos para usuarios humanos, difusión de noticias falsas, manipulación de opiniones, entre otros. Este trabajo de investigación muestra un método de clasificación entre usuarios humanos y bots en redes sociales principalmente Twitter. Los tweets de los usuarios son analizados y utilizados en diferentes estructuras de Ngramas donde se usan para entrenar un modelo de clasificación. El método logró una precisión superior al 90 % para los idiomas inglés y español.

Keywords: Detección de bots, redes sociales, procesamiento de lenguaje natural, usuarios humanos, tweets, clasificación, precisión.

Bot Detection on Social Media using Natural Language Processing Techniques

Abstract. Nowadays, social media is an essential component of our lives, not only for leisure time and connecting with people, but also as a source of communication and dissemination via the internet. Due to the increase of users and companies that interact within social networks, certain programs have been

introduced to interact with users, better known as bots. Bots are programs that pretend to have human behavior on social media. In recent years, the use of bots has increased exponentially since they can solve certain problems quickly, but they have also been used for other purposes such as: alteration of information, creation of conflicts for human users, dissemination of fake news, manipulation of opinions, among others. This research work shows a method of classification between human users and bots on social networks, mainly Twitter. Users' tweets are analyzed and used in different N-gram structures, which are used to train a classification model. The method achieved an accuracy of over 90% for the English and Spanish languages.

Keywords: Bot detection, social media, natural language processing, human users, tweets, classification, accuracy.

1. Introducción

En los últimos años hemos visto como las redes sociales han cambiado y se están convirtiendo en un medio de comunicación masivo donde cada día interactúan más usuarios entre ellas. Ahora que vivimos en una pandemia, la interacción en estos medios ha aumentado un 23 % según los reportes de Twitter [8].

En consecuencia a esto, las interacciones en redes sociales aumenta y a su vez, aumenta la difusión de información por estos medios. Estos aumentos en interacciones se deben a que las redes sociales son principalmente usadas por la inmediatez, facilidad y difusión de información.

Debido al uso cotidiano y en aumento en el uso de redes sociales; las empresas y marketing se comienzan a migrar e involucrarse más en estas tecnologías, haciendo uso estos medios para dar un mejor servicio como mostrar anuncios publicitarios, enfocar usuarios objetivos, etc.

Podemos ver que esto tiene sentido para dar un mejor servicio y controlar mejor al publico con el que se relacionan, pero también empiezan a usar diferentes implementaciones que les ayudan a mejorar su rol de negocios. Una implementación que ha tomado relevancia en los últimos tiempos son: Los bots.

Los bots en redes sociales son programas los cuales tratan de emular el comportamiento humano, haciendo parecer que los usuarios están interactuando con una persona como ellos a través de la red social, sin embargo es un programa de computadora.

Este tipo de programas actualmente suelen ser muy utilizados ya que las peticiones de los usuarios llegan a ser colosales y la mejor manera de distribuirlas y atenderlas es con el uso de estos programas.

Podría parecer que el uso de bots en redes sociales llega a ser benéfico en muchos aspectos pero la realidad muestra que estos programas se han visto muy involucrados en muchas controversias, una de ellas es el referéndum por la independencia catalana de octubre del 2017 [7], el cual se vio gravemente afectado por el uso de bots, ya que uno cada tres usuarios que daba su opinión era bot, donde principalmente eran usados para crear conflictos con los usuarios humanos.

Tabla 1. Combinaciones de Ngramas.

Español	Inglés
2 Caracteres-Ngramas	1 Caracteres-Ngramas
3 Caracteres-Ngramas	2 Caracteres-Ngramas
5 Caracteres-Ngramas	3 Caracteres-Ngramas
1 Palabras-Ngramas	2 Palabras-Ngramas
3 Palabras-Ngramas	3 Palabras-Ngramas

Este tipo de investigaciones muestra como este tipo de programas se ven relacionados en problemas tan sensibles para la sociedad.

Como podemos observar los bots pueden tener tareas sencillas como la de promocionar un producto o servicio, sin embargo también pueden tener tareas con otros fines como: Crear controversias de opiniones hacia un cierto interés [3].

En relación a este tipo de conflictos existen ciertas aplicaciones para ayudar a realizar la clasificación, uno de ellos es BorOrNot [2], el cual es una aplicación que nos indica la probabilidad de que un usuario en Twitter pueda ser un bot.

Esta aplicación es una gran herramienta donde no sólo usa el texto de los tweets para realizar la tarea, también emplea el uso de los metadatos de la cuenta para realizar la clasificación, como pueden ser: Cambios de foto de perfil, número de seguidores, si la cuenta ha sido verificada, entre otros.

Es una aplicación que relaciona todos estas características para determinar su tarea, sin embargo las tecnologías implementadas para el uso de bots también esta mejorando y puede complicarse si se deja de lado este problema.

Debido a este problema decidimos adentrarnos y realizar esta investigación proponiendo una solución utilizando técnicas de lenguaje natural sobre los tweets de los usuarios para realizar una clasificación e identificar los usuarios humanos y los bots.

Anteriormente participamos en PAN 2019 [5] para CLEF 2019 [1], con la tarea "Bots and Gender Profiling" donde mostramos una solución formada con estructuras de Ngramas, en particular usamos bigramas de carácter; de manera que para esta tarea decidimos usar la misma metodología; utilizando una estructura de Ngramas mucho más amplia [9].

2. Metodología

2.1. Conjunto de datos

Para realizar la investigación usamos el corpus de PAN 2019 [5], el cual consta de 3000 usuarios; los cuales; 1500 usuarios son bots y 1500 usuarios son humanos, esto para el idioma español. Para el idioma inglés usamos contamos con 4000 usuarios de los cuales 2000 son bots y los otros 2000 son humanos. Cada usuario es representado por un archivo XML el cual contiene 100 tweets de máximo 140 caracteres. Este corpus lo usamos únicamente para entrenar el modelo.

Tabla 2. Evaluación de clasificación entre bots y humanos en inglés en la tarea de PAN 2019 [5] con el corpus de entrenamiento.

Método de clasificación	1-grama	2-grama	3-grama
J48	63.44	65.54	71.29
NaiveBayes	66.48	69.25	69.25
RandomForest	83.21	85.55	83.24
RandomForest	86.79	83.24	89.31
SVM	90.42	**91.86**	89.41

2.2. Pasos del pre-procesamiento

Consideramos indispensable crear una capa de pre-procesamiento para cualquier investigación relacionada en el lenguaje, ya que en nuestro caso esta capa ayudó a tener mejor precisión para la clasificación. Los pasos del pre-procesamiento son los siguientes:

- Se cambió todo el texto a minúsculas para todos los conjutos de datos.
- Los signos de puntuación fueron removidos.
- Los dígitos fueron removidos.
- Los links y url fueron removidos directamente.
- Las menciones también fueron removidas sin ser etiquetadas.
- Los emoticones son removidos sin ser etiquetados.

2.3. Caracterísiticas

Ya que tenemos los datos preprocesados, pasamos a formar las características que nos ayuden a crear un modelo donde podamos clasificar dichos usuarios. Como parte de la investigación, decidimos entrar al concurso de PAN 2019, el concurso consistía en realizar 2 clasificaciones: Dado el conjunto de datos, se debía clasificar si un usuario era bot o humano, después ya identificados los humanos, la próxima tarea era clasificar por el género.

Dentro del concurso nos fueron asignados algunos recursos tecnológicos muy limitados por lo cual no podíamos sobrepasarlos ya que si lo hacíamos, los procesos daban por terminada la tarea y salimos sin resultados del concurso; por lo tanto; teníamos que crear un modelo que fuera limitado pero eficaz para dicha clasificación.

En nuestro caso usamos únicamente bigramas de caracteres. Utilizando los bigramas de caracteres para ambos lenguajes arrojaron 84.13 % de precisión para la clasificación entre bots y humanos evaluados por el comité del concurso.

Debido al desempeño que mostraron los bigramas, decidimos incrementar las características con diferentes combinación de Ngramas. La configuración dependiendo del lenguaje la podemos ver en la Tabla 1.

Estas características fueron seleccionadas de esta manera debido a que mientras realizábamos los experimentos e íbamos agregando o modificando la combinación de Ngramas, notábamos que bajaba la puntuación de la clasificación; como por ejemplo;

Tabla 3. Resultados de precisión evaluados en corpus de entrenamiento por PAN 2019 [5].

Clasificador	Español	Inglés
J48	77.98	83.90
NaiveBayes B	84.83	87.22
RandomForest	89.98	92.31
SVM	**94.13**	**96.04**

cuando agregábamos Ngramas de tamaño 4, la precisión bajaba un 12 %, de esta manera nos dimos cuenta que mientras hacíamos más grandes los Ngramas; tanto de caracteres como de palabras; era peor la precisión de la clasificación.

Después de obtener las combinaciones de Ngramas donde obtuvimos los mejores resultados, estos son contados y se obtiene la frecuencia de aparición dentro de texto entre todos los tweets de los usuarios. Con estas frecuencias agrupadas por cada usuario podemos crear vectores los cuales si juntamos a todos los usuarios del mismo idioma crearemos un modelo de espacio vectorial donde cada dimensión corresponde con un usuario en particular.

2.4. Modelo de espacio vectorial

De esta manera representamos los tweets en un módelo de espacio vectorial [6], donde cada columna es representada por una formación de Ngramas de los tweets, llenando así la matriz con todas las formaciones de Ngramas seleccionadas. Cada usuario es representado por una nueva dimensión en la matriz.

Si existe un Ngrama que no este dentro la matriz de un usuario, esta característica será incluida para todas las matrices y llenada con 0, esto es importante debido a que pueden ayudarnos a diferenciar los datos dentro del algoritmo clasificador.

3. Experimentos

En la parte experimental usamos distintos algoritmos los cuales puedan identificar similitudes entre los dos tipos de usuarios que tratamos de clasificar. Como habíamos comentado; anteriormente participamos en el concurso de PAN 2019 y seleccionamos la estructura de bigramas de caracteres debido a su buena precisión con el clasificador Support-vector machine.

Podemos ver en el Cuadro 4, los experimentos con las diferentes configuraciones, todas estas pruebas fueron evaluadas con K-fold Cross Validation con un K=10. En ese momento usamos este verificador debido a que no teníamos acceso al conjunto de datos de prueba. Los resultados para el idioma español se muestran en el Cuadro 2 y para el idioma inglés se encuentran en el Cuadro 3.

Con respecto al concurso [5], el ganado fue Juan Pizzarro [4], donde seleccionó una arquitectura de Ngramas de caracteres y de palabras. Su resultado para la clasificación de bots en español fue 93.33 % y en inglés fue 93.60 %, estos resultados fueron evaluados con por el cómite de PAN [5].

Continuando con la investigación, obtuvimos el conjunto de prueba de PAN 2019 [5]. Después seleccionamos esta metodología debido a que la combinación de Ngramas mostrada en el Cuadro 1, incrementaban la precisión de la clasificación.

Los resultado mostrados en el Cuadro 4 muestran diferentes clasificadores los cuales fueron evaluados únicamente con precisión en el concurso de PAN 2019 [5], resaltando que el mejor resultado obtenido fue utilizando Support-Vector Machine con la combinación de Ngramas del Cuadro 1. Con los procedimientos en esta investigación, logramos obtener mejores resultados del estado del arte, en este caso a el ganador [4] del concurso PAN de 2019 [5].

4. Conclusiones

Dados los resultados obtenidos en esta investigación, consideramos viable el uso de Ngramas para realizar una clasificación entre bot y usuarios humanos dentro de una red social. En este caso únicamente realizamos las pruebas con tweets debido al corpus que utilizamos. Planteamos realizar la misma investigación con otra red social ya que pensamos que se puede replicar esta misma metodología debido a que hacemos uso únicamente de texto y no de otro tipo de características particulares como los metadatos de los usuarios.

Algo sobresaliente que mostró esta investigación en el concurso de PAN 2019 [5] fue la precisión de la clasificación únicamente usando bigramas de caracteres para ambos lenguajes [9]. Podemos observar que ambos lenguajes guardan similitudes para este tipo de tareas y si se realiza investigaciones entre estos dos lenguajes se pueden usar metodologías muy similares para las clasificaciones.

Cabe resaltar que esta investigación fue usando únicamente tweets de Twitter, por lo cual nos parecería muy interesante replicar toda esta investigación para bots de otra red social, con esto podemos ver si existen similitudes y diferencias entre los bots de distintas redes sociales.

En la actualidad y debido a la era en la que vivimos consideramos importante continuar con mejoras de implementaciones y evolución de los bots, así mismo, llevar estas evoluciones de manera responsable ya que lo importante de este tipo de tecnologías es mejorar para ayudar y no obstaculizar o corromper a la sociedad misma.

Referencias

1. Cappellato, L., Ferro, N., Losada, D. E.: Working notes of conference and labs of the evaluation forum 2019 - conference and labs of the evaluation forum. In: Conference and Labs of the Evaluation Forum 2019 Working Notes. vol. 2380 (2019), ceur-ws.org/Vol-2380/
2. Davis, C. A., Varol, O., Ferrara, E., Flammini, A., Menczer, F.: BotOrNot: A system to evaluate social bots. In: Proceedings of the 25th International Conference Companion on World Wide Web. pp. 273–274 (2016) doi: 10.1145/2872518.2889302
3. País, E.: Así fabrican los partidos políticos un trending topic, (2018)
4. Pizarro, J.: Using n-grams to detect bots on Twitter notebook for PAN at the conference and labs of the evaluation forum 2019. In: Conference and Labs of the Evaluation Forum 2019 Labs and Workshops, Notebook Papers (2019), ceur-ws.org/Vol-2380/

5. Rangel, F., Rosso, P.: Overview of the 7th author profiling task at pan 2019: Bots and gender profiling. In: Conference and Labs of the Evaluation Forum 2019 Labs and Workshops, Notebook Papers (2019), ceur-ws.org/Vol-2380/
6. Sidorov, G.: Syntactic n-grams in computational linguistics. Springer (2019)
7. Stella, M., Ferrara, E., De-Domenico, M.: Bots increase exposure to negative and inflammatory content in online social systems. In: Proceedings of the National Academy of Sciences. vol. 115, pp. 12435–12440 (2018)
8. Vives, J.: El coronavirus dispara el número de usuarios de twitter (2020)
9. Yacob-Espinosa, D., Gómez-Adorno, H., Sidorov, G.: Bots and gender profiling using character bigrams. In: Conference and Labs of the Evaluation Forum 2019 Labs and Workshops, Notebook Papers (2019), ceur-ws.org/Vol-2380/

ISSN 1870-4069

Análisis y seguimiento de tópicos en las conferencias matutinas del presidente de México

Luis Armando Arias-Romero[1], Gabriela Ramírez-de-la-Rosa[1], Esaú Villatoro-Tello[1,2]

[1] Universidad Autónoma Metropolitana, Unidad Cuajimalpa, México

[2] Idiap Research Institute, Martigny, Switzerland

ariasluis.ar@gmail.com, {gramirez, evillatoro}@cua.uam.mx

Resumen. El lenguaje es un recurso importante para los políticos. El análisis del lenguaje de políticos como el presidente de un país es una tarea importante para diferentes disciplinas como la lingüística, sociología, y comunicación. En este artículo presentamos un método que incorpora la detección automática de tópicos en 534 conferencias matutinas del presidente de México, Andrés Manuel López Obrador, utilizando LDA. Posteriormente, a través de un sistema web, al que denominamos DICTA, una persona puede visualizar de forma rápida los temas tratados en dichas conferencias. En la evaluación experimental, se obtuvo el valor más alto de coherencia al utilizar 18 tópicos.

Palabras clave: Detección y seguimiento de tópicos, LDA, discurso político, procesamiento del lenguaje natural.

Topic Analysis and Tracking from Mexico's President Daily Press Briefing

Abstract. Language is a very useful tool to politicians. Several areas such as linguistics, sociology and communication consider important the study of political discourse analysis. In this paper we present a method for topic detection using LDA in 534 daily press briefing of the Mexico's president: Andrés Manuel López Obrador. Subsequently, through a web system, which we call DICTA, a person can quickly view the topics discussed in these briefing. Through experimental evaluation we found the highest coherence value when using 18 topics.

Keywords: Topic detection and tracking, LDA, political discourse, natural language processing.

1. Introducción

El lenguaje humano es complejo y diverso. A través del lenguaje podemos expresar pensamientos, sentimientos, emociones, etc. [15]. En la política, el lenguaje puede usarse para convencer de una idea, cambiar la forma de pensar de una comunidad de personas, inspirar, pero también dividir. El análisis del lenguaje en el contexto político se ha estudiado durante muchos años por diversos campos como la lingüística, sociología, comunicación entre otros [16].

Dependiendo de la disciplina, el estudio del lenguaje político es diverso. Por ejemplo, se estudia el vocabulario utilizado (aspectos léxicos), la relación del léxico dentro de una oración (relación lexico-gramatical) [13], el estilo del discurso, entre otras características. El análisis del lenguaje en estos campos ayuda a entender fenómenos sociales, económicos y/o políticos [3, 13].

Usualmente, el análisis de este tipo de fenómenos no depende del análisis de un sólo discurso. Así, cuando se requiere estudiar textos políticos que abarcan más de un documento (o discurso), los especialistas se enfrentan con el volumen de textos a analizar [7].

El procedimiento manual involucra la lectura de todos los textos para realizar un estudio a fondo sobre temas que aborda y su influencia o correlación en otras áreas como la afectación (positiva o negativamente) en la economía, polarización [9], o consecuencias sociales [8, 6].

Adicionalmente, la importancia o influencia del emisor del lenguaje está relacionado con el impacto de dicho discurso; particularmente en la política. En México, se podría argumentar que el actor de más influencia política es el presidente de la República, actualmente Andrés Manuel López Obrador.

Desde el inicio de su gestión, en diciembre de 2018, López Obrador ha establecido una rutina de conferencias diarias donde presenta una variedad de asuntos que quiere comunicar al país. En estas conferencias, que se transmiten en vivo por YouTube y que se reportan en la mayoría de los noticieros de cobertura nacional, el presidente anuncia los programas sociales de su gobierno, da instrucciones a sus colaboradoras y envía mensajes políticos. Con frecuencia, en las conferencias participan funcionarios de su gobierno, según el tema que quiera abordar o para atender algún problema específico [10].

La regularidad de estas conferencias genera una gran cantidad de información. Información que deberá ser organizada y analizada tan pronto como se tiene disponible. En este contexto, con ayuda del procesamiento automático de textos, se pretende generar un sistema web que permita a los analistas políticos, pero también a la población general, explorar los tópicos o temas que son discutidos a lo largo de las conferencias de prensa matutinas del actual presidente de México.

Para llevar a cabo la detección y el seguimiento de los tópicos se utilizan las transcripciones de las conferencias[3]. Este sistema no intenta sustituir la labor crítica y especializada de las personas expertas. El objetivo del sistema es apoyar en la organización temática de lo que el presidente de México comunica en las conferencias matutinas diarias.

[3] La colección de transcripciones se pueden consultar en: https://lopezobrador.org.mx/transcripciones/

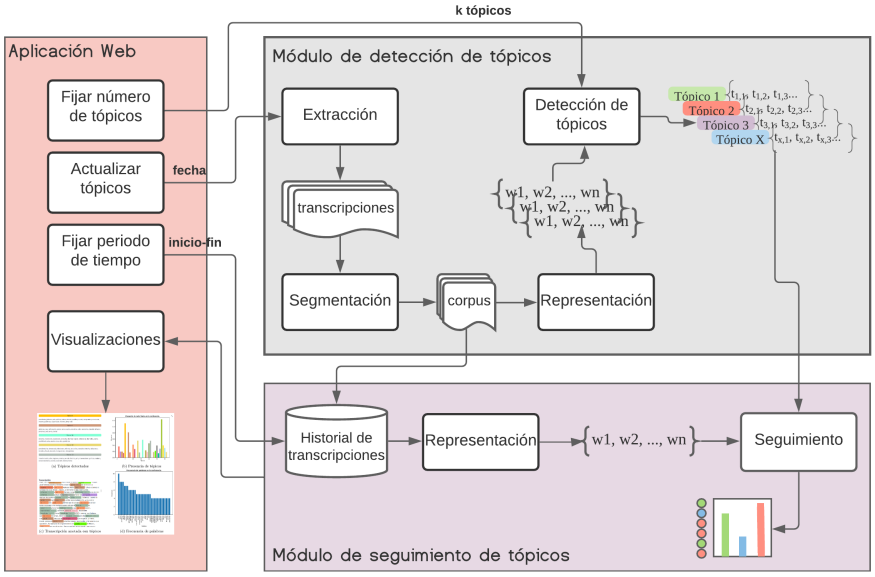


Fig. 1. Esquema general del sistema propuesto que contiene tres módulos principales: detección de tópicos, seguimiento de tópicos y la aplicación web.

En este artículo se propone realizar un sistema web que i) incorpore un método de detección automático de tópicos que demuestre generar tópicos de alta cohesión y variados, ii) incorpore un método de seguimiento de los tópicos encontrados dentro de un conjunto nuevo de transcripciones, y iii) permita explorar, mediante visualizaciones propuestas, los tópicos en una o más conferencias seleccionadas.

Por lo tanto, las aportaciones de este proyecto son dos. Primero, la evaluación de un modelo basado en LDA para la detección de tópicos en transcripciones en español. Segundo, el desarrollo de un sistema web que incorpore dicho modelo y que permite explorar los tópicos o temas que aborda el presidente de México en una o más conferencias.

2. Trabajo relacionado

La Detección y Seguimiento de Tópicos (TDT, por sus siglas en inglés) es un área de estudio dentro del procesamiento del lenguaje natural (PLN). El TDT está conformado por tres tareas [1]: i) Segmentación de una fuente de información en historias. ii) Detección de tópicos no conocidos por el sistema. Y iii) Seguimiento de tópicos conocidos por el sistema.

Existen herramientas que analizan textos en inglés para la detección de tópicos para su posterior visualización. A continuación se describen algunas que tienen características similares a las propuestas:

– **Terminology Extraction**[4]. Esta herramienta identifica términos clave dentro de un documento de texto. En resumen, se compara la frecuencia de palabras en un documento dado, con la frecuencia de uso de las palabras dentro de un lenguaje determinado. Para encontrar las palabras relevante, se utiliza la distribución de Poisson, el método de estimación por máxima verosimilitud y la frecuencia inversa de documentos. Además, utiliza un etiquetador probabilístico para identificar los términos que serán extraídos.
– **Term Extraction**[5]. Similar a la herramienta anterior, extrae la terminología de un texto de entrada. Pero Term Extraction permite la configuración de ciertos parámetros, como el número de términos a buscar dentro del texto, y el número de palabras que pueden conformar un término (un término se construye con un conjunto de palabras).
– **jsLDA**[6]. Esta herramienta implementa, en el lenguaje de programación JavaScript, el modelo de detección de tópicos LDA. Permite realizar la búsqueda de temas dentro de un conjunto documentos. Entre las configuraciones que un usuarios puede realizar están: el número de tópicos a ser encontrados y el número de iteraciones del modelo sobre los documentos. Dentro de las visualizaciones disponibles, la herramienta permite ver las correlaciones que hay entre los tópicos encontrados. Además, por cada tópico es posible visualizar una serie de tiempo de su presencia en los documentos. Adicionalmente, jsLDA muestra algunas estadísticas del vocabulario encontrado, como la frecuencia del término y la especificidad de cada término con respecto a los tópicos.

De manera general, todas las herramientas descritas son capaces de analizar textos en inglés. Aunque la mayoría de ellas no tiene visualizaciones intuitivas para el público general. La herramienta capaz de aceptar diferentes parámetros de configuración, jsLDA, produce visualizaciones orientada a describir el comportamiento del algoritmo LDA, por lo que las gráficas que genera están dirigidas a personas que conocen el funcionamiento de LDA.

3. Sistema propuesto

La Figura 1 muestra el diagrama general del sistema propuesto que contiene tres módulos: i) Módulo de detección de tópicos; ii) Módulo de seguimiento de tópicos; y iii) Aplicación Web Dicta. La descripción de cada módulo se detalla a continuación.

3.1. Módulo de detección de tópicos

La fuente primaria del sistema propuesto, como ya se ha mencionado, son las transcripciones que se publican diariamente en el sitio web del presidente de México actual[7]. Por lo tanto, el primer paso del sistema es la obtención de las transcripciones.

[4] http://labs.translated.net/terminology-extraction/
[5] http://termextract.fivefilters.org/
[6] https://mimno.infosci.cornell.edu/jsLDA/
[7] https://lopezobrador.org.mx/transcripciones/

PRESIDENTE ANDRÉS MANUEL LÓPEZ OBRADOR: Buenos días.

Bueno, vamos a informar y se trata de una muy buena noticia. Como se dijo desde el principio del gobierno, se heredó un déficit de especialistas médicos o médicos especialistas. Fue un saldo -otro más- negativo de la política neoliberal…

ALEJANDRO SVARCH PÉREZ, TITULAR DE LA COORDINACIÓN NACIONAL MÉDICA DEL INSTITUTO DE SALUD PARA EL BIENESTAR (INSABI): Con su permiso, señor presidente, señor secretario. Muy buenos días.

Como ustedes saben, lo hemos platicado en este espacio, nuestro país lamentablemente tiene un déficit estructural de médicos especialistas. Esto ha sido particularmente sensible en momentos como la pandemia que vivimos…

JORGE ALCOCER VARELA, SECRETARIO DE SALUD: Muchas gracias, señor presidente.

Sí, esta es una estrategia, pero no la única. Tenemos desde luego ya de años atrás varias escuelas de medicina y que fueron impulsadas por la doctora Sosa, que conduce este plan, también de becas de carreras de medicina y de otras especialidades a lo largo de todo el país, que superan las 100, 'Benito Juárez' es su nombre y desde luego ahí hay 12 escuelas de medicina…

INTERLOCUTOR: Disculpe, secretario, ¿tendrá alguna fecha de cuándo va a iniciar en funciones esta Universidad de la Salud y con cuántos estudiantes iniciará?...

PRESIDENTE ANDRÉS MANUEL LÓPEZ OBRADOR: Nos acaban de informar que ya está operando, funcionando…

(a) No segmentada

Buenos días. Bueno, vamos a informar y se trata de una muy buena noticia. Como se dijo desde el principio del gobierno, se heredó un déficit de especialistas médicos o médicos especialistas(...) A ver, si puedes explicar sobre las nuevas universidades. Sí, nada más que es importante que se dé a conocer que se han iniciado ya, incluso están en operación, muchas escuelas de medicina en todo el país, tanto del sistema de universidades 'Benito Juárez' como iniciativas que tomaron gobiernos locales(...) Sigue habiendo clases, pero virtuales Vamos a presentarles toda la cobertura, porque son 140 universidades públicas nuevas que están en proceso, es algo extraordinario. Acabo de inaugurar dos, una en Agua Prieta y otra en Tlatizapán, Morelos; en Agua Prieta, Sonora, y en Tlaltizapán, Morelos. Son 140 del sistema de universidades 'Benito Juárez' y están en los municipios, en las regiones más apartadas(...)

(b)Segmentada

Fig. 2. Fragmento de la transcripción del 5 de noviembre de 2019, antes y después de la segmentación usando como actor político de interés a Presidente Andrés Manuel López Obrador.

La salida general de este módulo es un conjunto de vectores, uno por cada tópico detectado. Cada vector-tópico contiene un conjunto de términos asociados a ese tópico. Este módulo se compone por cuatro procesos: Extracción, Segmentación, Representación y Detección.

Extracción. Este proceso es el encargado de realizar la recolección periódica de las transcripciones. Para la implementación de este proceso se usó la biblioteca de Python Scrapy[8]. Al final de la extracción, las transcripciones obtenidas son almacenadas en formato CSV (Comma-Separated Values).

Segmentación. Durante las conferencias de prensa diarias de López Obrador a menudo participan funcionarios de su gobierno. Estas participaciones también son transcritas. Por lo tanto, el objetivo de este proceso es identificar el contenido textual relacionado con el actor político objetivo y el resto de las participaciones es eliminada. En el caso de este proyecto, el actor político objetivo es el presidente de México.

[8] https://scrapy.org/

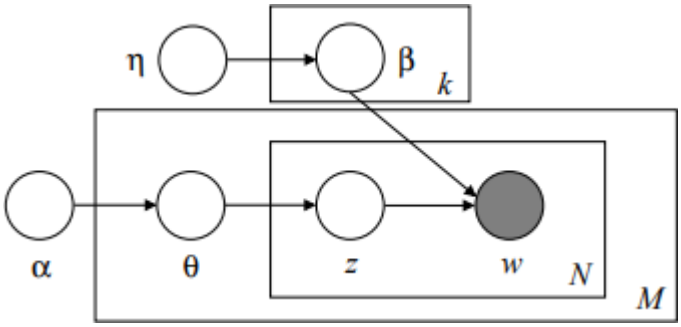


Fig. 3. Representación gráfica del modelo LDA.

Las transcripciones que contienen sólo la información del político de interés formarán parte del corpus de documentos que se usarán para la detección de los tópicos a través LDA. Como puede verse en la Figura 1 el corpus se almacena en una base de datos que contiene el historial de las transcripciones que la aplicación web usará posteriormente.

En la Figura 2a, se puede observar un fragmento de una transcripción sin segmentar (entrada de este proceso) y posteriormente, en la Figura 2b aparece sólo el texto correspondiente al actor de interés (i.e., López Obrador). Este fragmento corresponde a la transcripción de la conferencia de prensa del 5 de noviembre de 2019.

Representación. Una vez que se extraen y segmentan las transcripciones, se preparan los textos para la representación. Dado que el interés del sistema propuesto es la detección de temas o tópicos tratados dentro de las conferencias matutinas, se eliminan todas las palabras que no sean sustantivos, con el fin de conservar únicamente palabras de contenido. El conjunto completo de pasos en el pre-procesamiento se lista a continuación (los tres primeros procesos fueron realizados con expresiones regulares):

- Se transforma todo el texto a minúsculas.
- Se eliminan los números contenidos dentro del texto.
- Se elimina cualquier signo de puntuación.
- De cada una de las transcripciones se extraen los sustantivos[9].

Cabe destacar que las transcripciones se realizan con ayuda de una técnica estenográfica, por lo que en múltiples ocasiones existen errores ortográficos contenidos dentro de la transcripción y por ende dentro del conjunto de documento ya preprocesado.

Detección. Una vez pre-procesado y representado el conjunto de documentos a analizar, se utiliza el algoritmo de detección de tópicos para aprender los tópicos relevantes de ese conjunto. Específicamente, el modelo empleado para la detección de tópicos es Latent Dirichlet Allocation (LDA) [5].

[9] Para obtener los sustantivos se utilizó: https://spacy.io/models/es

Evolución de los tópicos.

Distribución de tópicos en el tiempo.

Participantes en la conferencia del dia 2020-07-21 :

	Nombre
0	SECRETARIO MARCELO EBRARD CASAUBÓN
1	HUGO LÓPEZ GATELL RAMÍREZ
2	PRESIDENTE ANDRÉS MANUEL LÓPEZ OBRADOR
3	JORGE ALCOCER VARELA SECRETARIO DE SALUD

Participantes en las conferencias.

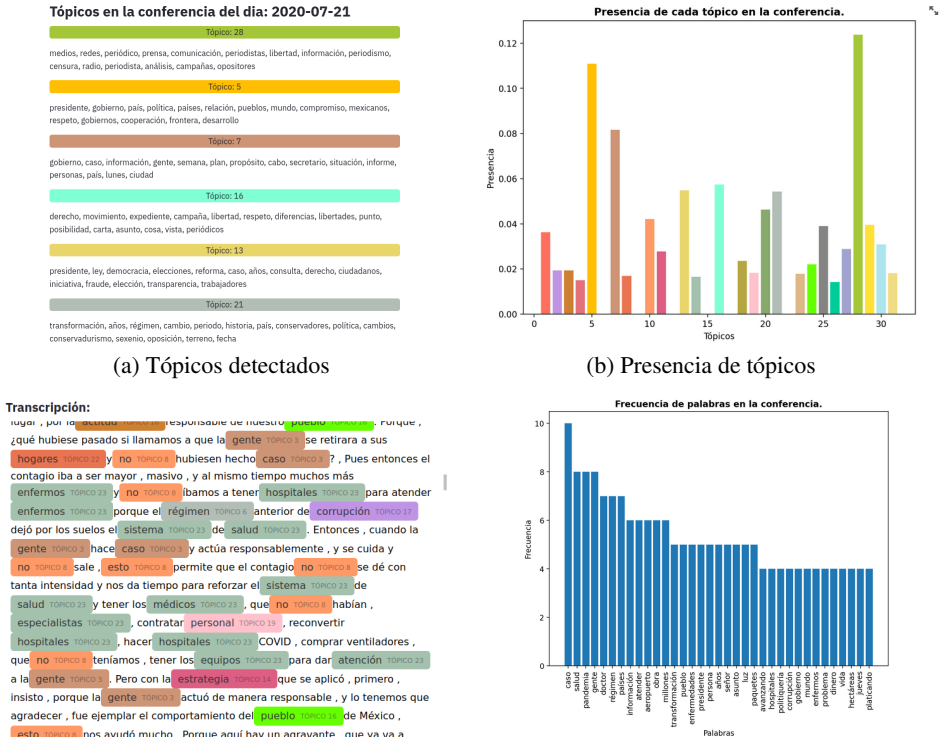
Fig. 4. Visualizaciones generales en DICTA.

En la sección 4 se presentan los experimentos para determinar el número adecuado de tópicos para este conjunto de documentos. La salida de este proceso son los tópicos detectados por el modelo.

Cada tópico esta asociado con un conjunto de términos que describen o que pertenecen al tópico en cuestión. Junto a cada uno de los términos se encuentra un valor que indica la pertenencia que cada término tiene con respecto al tópico.

La idea general detrás del modelo LDA, consiste en que los documentos están representados como una distribución aleatoria sobre tópicos latentes, donde cada tópico se caracteriza por una distribución de términos.

Esto es, se asume que los tópicos existen antes que los documentos y que estos documentos se construyen a partir de tales tópicos [5, 4]. En la Figura 3 se puede ver la representaciósn gráfica del modelo LDA.



(a) Tópicos detectados

(b) Presencia de tópicos

(c) Transcripción anotada con tópicos

(d) Frecuencia de palabras

Fig. 5. Visualizaciones en DICTA que contiene información de una conferencia específica.

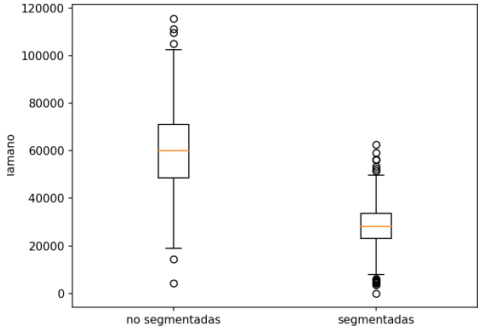
Cada nodo es una variable aleatoria; α y η son distribuciones Dirichlet; β es una distribución de palabras, una para cada tópico; θ es una distribución de tópicos, una por cada documento; N, M y k denotan replicación; N denota la colección de palabras dentro de cada documento; M es el conjunto de documentos en la colección; y k el número de tópicos. Finalmente, w denota una palabra en un documento y z un tópico dentro de un conjunto de tópicos [5].

3.2. Módulo de seguimiento de tópicos

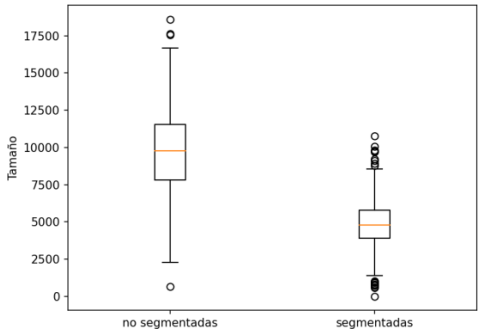
Este módulo hace el seguimiento de los tópicos (previamente detectados) en un conjunto dado de transcripciones de las conferencias matutinas. Para su funcionamiento, este módulo necesita dos parámetros.

El primer parámetro es el periodo de tiempo de las transcripciones a analizar del historial de transcripciones (ver Figura 1). Estas transcripciones deberán ser pre-procesadas y representadas usando los mismos procesos descritos en la etapa de Detección de tópicos (ver Sección 3.1).

El segundo parámetro, es el conjunto de vectores correspondientes a los tópicos detectados que corresponde a la salida del Módulo de detección de tópico explicado en la sección anterior.



(a) Tamaño (en caracteres) de las transcripciones no segmentadas y segmentadas.



(b) Tamaño del vocabulario de las transcripciones no segmentadas y segmentadas.

Fig. 6. Estadísticas del conjunto de datos usados en la evaluación experimental.

La salida de este módulo es un nuevo vector que contiene los tópicos encontrados dentro de la transcripción y el valor correspondiente a la pertenencia de cada tópico dentro del documento analizado.

Luego, este vector es enviado al sistema web para generar la visualización correspondiente. En la Figura 1 se representa esta salida como una distribución de los tópicos en los documentos analizados y las palabras (círculos) de cada tópico encontradas.

3.3. Dicta: Aplicación web de visualización de tópicos

El sistema web denominado DICTA[10], integra los módulos descritos anteriormente y permite al usuario visualizar los tópicos de un conjunto de conferencias matutinas dado un rango de fechas. Como puede verse en la Figura 1, la comunicación con el módulo de detección de tópicos se realiza a través de la selección del número de tópicos simultáneamente con la opción de actualizar los tópicos.

La comunicación con el módulo de seguimiento de tópicos se lleva a cabo a través de fijar el periodo de tiempo de análisis y posteriormente, con la visualización generada. Las funcionalidades más relevantes de DICTA son:

[10] DICTA y la base de datos utilizada están disponibles en https://github.com/lyr-uam/dicta

Tabla 1. Primeras 20 palabras asociadas a los cuatro tópicos generados con Top2Vec. Entre paréntesis, el tema asignado por los autores a cada tópico.

Tópico ID	Términos asociados
0 (política exterior)	migratorio Centroamérica ebrad marcelo donald humanos aranceles fenómenos confrontación migración cooperación relaciones exteriores soberanía guerra trump naciones paz derechos violencia
1 (educación / clases)	educativo gratuita maestros educativa educación medicina apartadas calidad normales servicios universidades medicinas mejorar medicamentos superior vendían salud escuelas niveles unam
2 (salud)	epidemia endemia coronavirus camas enfermos intensiva terapia recomendaciones hugo ventiladores enfermeras salvar proyecciones hospitalización especialistas medico crisis cuidarnos normalidad científicos
3 (energía)	estancias pemex infantiles cancelar energética contratos transparencia lopez simulación obrador comisión signifique expediente electricidad debate gasoducto organizaciones llamada organismo barriles

– Fijar el número de tópicos. Este proceso permite al usuario indicar cuántos tópicos desea explorar. Entre mayor el número de tópicos, más fina es la organización de temas a visualizar. Si no se fija un número, el sistema usará el obtenido en la etapa de evaluación (que se describe en la Sección 4).

– Actualizar tópicos. Esta opción, permite lanzar el módulo de detección de tópicos para que se puedan acceder a las transcripciones más recientes. Adicionalmente, se considera como entrada el número de tópicos fijado por la opción anterior.

– Fijar periodo de tiempo. Este proceso valida que las fechas ingresadas contengan una transcripción dentro de la base de datos (o historial de transcripciones). Usualmente en los días festivos en México o fines de semana no existen conferencias matutinas. Después de esta validación se cargan a memoria las conferencias que pertenezcan al rango de tiempo especificado.

Una parte importante del sistema propuesto es la generación de visualizaciones dirigidas al público general. Por lo tanto, el sistema genera seis diferentes gráficos divididos en información general de las conferencias (Figura 4) e información específica de una conferencia a analizar (Figura 5). A continuación se describen brevemente cada visualización.

– Distribución de tópicos en el tiempo. Esta gráfica (Figura 3.1) muestra la presencia, en porcentajes de cada tópico en la conferencias de prensa de la base de datos en un momento dado. En el eje de las x se grafican las conferencias de la más antigua a la más nueva; y en el eje de las y se grafica el porcentaje relativo al 100 % de la transcripción dada, de la presencia de cada tópico. Por ejemplo, el tópico 5 aparece en todas las conferencias entre el 10 y el 20 % del total de cada transcripción.
– Participantes en las conferencias de prensa. Este gráfico (Figura 3.1) muestra los participantes de una conferencia dada. En este proyecto nos enfocamos en el análisis de un único actor político: el presidente de México, sin embargo, existe información que en un futuro puede ser relevante.

Tabla 2. Primeras 30 palabras asociadas a los cuatro tópicos generados con LDA (Gensim). Entre paréntesis, el tema asignado por los autores a cada tópico.

Tópico ID	Términos asociados
0 (intro. a la conferencia)	vamos entonces va si mexico mil ahora pueblo gobierno gente ver aquí van país bien así como voy caso bueno hacer decir tiempo luego mismo importante haciendo mañanera ser puede
1 (sin asignación)	entonces vamos si va gobierno corrupción ahora ver pues caso ser bueno como mismo gente méxico ahí aquí van presidente así voy hacer poder puede bien tiempo pueblo luego país
2 (salud / economía)	mil va salud millones vamos si entonces empresas médicos hospitales pesos ahora presupuesto medicamentos van como año avión créditos dos petroleo caso gente ahí trabajadores dinero seguro bueno hacer deuda
3 (errores de transcripción)	delpresupuesto alas demanera vamos estesemana periodneoliberal quese muybien poderlegislativo atransparenar lacorrupcion enel ainformar delpoder camade elpresupuesto antidemocratica peregrinos ingrese laseguridad enlos experimentados siestamos tenemospetroleo sanchezcordero supuestmenteesta combustibleque tenerproteccion superdelegado tenganprincipios

– Tópicos detectados. Gráfica (Figura 3.1) que lista los términos de los tópicos con mayor presencia en la conferencia analizada. Cada tópico se lista con un color particular y el tópico con mayor porcentaje de presencia en la conferencia se muestra primero (note que el número del tópico es sólo un identificador, no corresponde a la importancia o mayor presencia).
– Presencia de tópicos. En esta gráfica de barras (Figura 3.1) se muestra la presencia de cada tópico en la conferencia analizada. El tópico se representa por un identificador numérico y un color asignado (cabe hacer notar que el color del tópico es consistente en todas las gráficas mostradas en esta sección).
– Transcripción anotada. Esta visualización (Figura 3.1) muestra el documento transcrito de la conferencia analizada indicando con un mismo color todos los términos pertenecientes al mismo tópico. Se muestra el tópico al que pertenece dicho término de modo que en conjunción con la gráfica de la Figura 3.1 se pueda hacer un análisis más detallado de cada conferencia.
– Frecuencia de palabras. Finalmente, en esta gráfica de barras (Figura 3.1) se muestra la frecuencia de los términos más mencionados en cada transcripción analizada. Esta gráfica no está relacionada con los tópicos directamente, sino que es un conteo independiente del número de palabras en una conferencia.

4. Evaluación del modelo de detección de tópicos

Como complemento a la aportación principal presentada en este artículo, realizamos dos tipos de validación del modelo de detección de tópicos. Por un lado se compararon de forma cualitativa dos algoritmos de detección de tópicos para elegir el modelo a

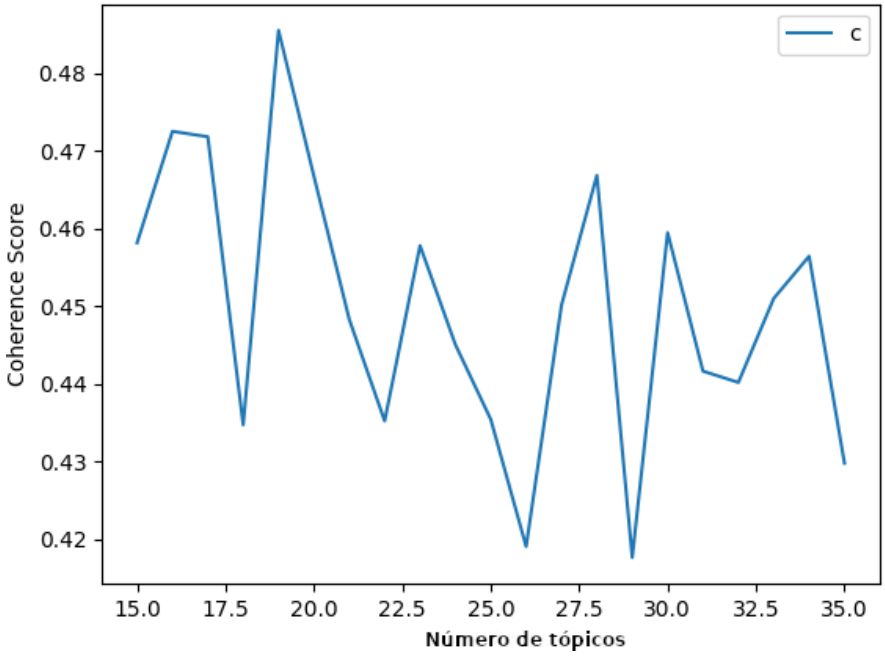


Fig. 7. Valor de coherencia en los tópicos obtenidos usando LDA.

incorporar en DICTA. Por otro lado, una vez elegido LDA, se usó una métrica de cohesión para determinar el número de tópicos recomendado para la tarea.

4.1. Conjunto de datos

El conjunto de datos está compuesto por 534 transcripciones segmentadas (como se describe en la Sección 3.1). Estas conferencias se dieron entre el 3 de diciembre del 2018 y el 8 de febrero del 2021. Por un lado, en la Figura 3.2 se puede observar que el tamaño promedio de caracteres de las transcripciones antes y después de la segmentación varía considerablemente, esto nos indica que la participación del actor de interés (el presidente de México) ocupa aproximadamente la mitad de las conferencias diarias, a pesar de que el número de participantes es más de 1.

Particularmente, el tamaño promedio de las transcripciones no segmentadas es de 60364 caracteres ($\sigma = 17106$) y el tamaño promedio de las transcripciones segmentadas es de 28199 caracteres ($\sigma = 915$). Por otro lado, en la Figura 3.2 se pueden observar el tamaño promedio del vocabulario usado en las transcripciones, antes y después de la segmentación.

El vocabulario es definido como palabras únicas. Las transcripciones no segmentadas tienen un tamaño de vocabulario promedio de 9728 ($\sigma = 2746$) palabras. Las transcripciones segmentadas tiene en promedio un tamaño de vocabulario de 4834 ($\sigma = 1572$).

4.2. Selección del modelo de detección de tópicos

En esta sección se evaluaron dos modelos de detección de tópicos: Top2Vec [2] y LDA [5] (bajo la implementación de Gemsim[11]). Top2Vec encuentra el número de tópicos de forma automática dentro del conjunto de documentos, mientras que a LDA se debe especificar el parámetro k (Figura 3). Por lo tanto, primero realizamos la prueba con Top2Vec para después utilizar el número de tópicos encontrado como parámetro de k para la prueba con LDA.

En la Tabla 1 se pueden ver los 4 tópicos generados por el modelo Top2Vec. De aquí se observa que los tópicos son identificables en áreas generales del gobierno de un país: política exterior, educación, salud, y energía. Como segundo experimento se utilizó LDA para generar 4 tópicos (este número de tópicos fue guiado por el número identificado de forma automática por Top2Vec). En la Tabla 2 se pueden observar los tópicos generados por LDA.

Es notorio que el conjunto de tópicos generados por LDA no son de la misma calidad que los generados por Top2Vec. Sin embargo, debido a la gran variedad de temas que se abordan en las conferencias matutinas del presidente de México actual, es deseable tener la posibilidad de generar tópicos más finos. A partir de la observación del tópico 2 de la Tabla 2 es viable inferir que a un mayor número de tópicos LDA podría generar temas de mejor calidad, dividiendo el tópico 2 en dos o más temas particulares.

4.3. Validación del número de tópicos

En este apartado se describe la validación del número de tópicos para las transcripciones de las conferencias del presidente de México, López Obrador. La calidad de los tópicos detectados por modelos como LDA, es medida por su grado de coherencia. Se puede decir que un tópico es coherente si la gran parte de los términos que describen a ese tópico en particular están relacionados [14].

Dado que LDA requiere que se especifique el valor de k (ver Figura 3), es importante determinar un valor que de k que responda a las necesidades de la tarea. Un valor pequeño de tópicos resultará en tópicos demasiado generales; mientras que un valor muy grande de k podría resultar en tópicos que no se pueden interpretar o que podrían combinarse.

Así, para medir la coherencia de los tópicos se puede utilizar un conjunto de métricas llamadas Coherence Measures. Estas métricas evalúan los tópicos a través de un promedio de la similitud entre pares de términos, que son tomados de los términos principales de un tópico [12]. En [11] se desarrolló una nueva métrica denominada C_v y que tiene una alta correlación con lo que los humanos consideran buenos tópicos.

Esta métrica esta basada en la combinación de las siguientes tres Indirect Cosine Measure, NPMI(Nomalized Pointwise Mutual Information), y Boolean Sliding Window (la definición formal de la métrica se puede encontrar en [11]). El valor de C_v está normalizado y va de 0 a 1; entre mayor el valor, mayor calidad de los tópicos.

[11] https://radimrehurek.com/gensim/

Para este experimento se generaron modelos incrementando el número de tópico (de 15 a 35). En la Figura 7 se puede observar el valor de cohesión C_v de los 20 modelos construidos. Los parámetros de LDA utilizados para este experimento fueron: $\alpha = 0.1$ y $\eta = 0.9$. De aquí, el mejor valor de coherencia obtenido es con un número 18 de tópicos, seguido de 28.

5. Conclusiones

En este artículo se presentó un sistema web que apoya en el análisis del discurso político del presidente de México, Andrés Manuel López Obrador. El sistema comprende tres módulos principales: el módulo de detección de tópicos, el módulo de seguimiento de tópicos y la aplicación web.

La aplicación propuesta permite a un usuario visualizar los temas o tópicos que están presentes en el dichas conferencias. Entre las funcionalidades del sistema se encuentran: fijar un periodo de tiempo para el análisis, actualizar el modelo de detección de tópicos (fijando el número de tópicos a detectar).

El módulo de detección de tópicos utiliza LDA con la implementación de Gensim para la generación automática de tópicos. Se evaluó la cohesión de los tópicos generados por LDA, obteniendo un valor máximo de Coherence score de 0.49 con 18 tópicos. Actualmente el sistema analiza a un actor político de interés. Como trabajo futuro se buscará que el usuario pueda determinar el conjunto de actores políticos (participantes en las conferencias) que desee analizar.

Agradecimientos. Los autores y autora agradecen a la Universidad Autónoma Metropolitana Unidad Cuajimalpa por el apoyo otorgado durante la realización de este proyecto. El tercer autor además fue apoyado parcialmente por Idiap Research Institue y el SNI-CONACyT México.

Referencias

1. Allan, J., Carbonell, J. G., Doddington, G., Yamron, J., Yang, Y.: Topic detection and tracking pilot study final report (1998)
2. Angelov, D.: Top2vec: Distributed representations of topics (2020) doi: 10.48550/arXiv.2008.09470
3. Bhatia, A.: Critical discourse analysis of political press conferences. Discourse & Society, vol. 17, no. 2, pp. 173–203 (2006) doi: 10.1177/0957926506058057
4. Blei, D. M.: Probabilistic topic models. Communications of the Association for Computing Machinery, vol. 55, pp. 77–84 (2012) doi: 10.1145/2133806.2133826
5. Blei, D. M., Ng, A. Y., Jordan, M. I.: Latent dirichlet allocation. Journal of Machine Learning Research, vol. 3, pp. 993–1022 (2003)
6. Francisco-Ortega, D.: Coronavirus outbreak in Mexico: A critical discourse analysis of AMLO's speech. Open Journal for Studies in Linguistics, vol. 3, no. 2, pp. 93–100 (2020)
7. Grimmer, J., Stewart, B. M.: Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political Analysis, vol. 21, no. 3, pp. 267–297 (2013) doi: 10.1093/pan/mps028

8. Marini, A. M.: El mesías tropical: Aproximación a fenómenos populistas actuales a través del discurso de López Obrador, no. 139, pp. 153–170 (2018)
9. Navarro, F., Tromben, C.: Estamos en guerra contra un enemigo poderoso, implacable: Los discursos de Sebastián Piñera y la revuelta popular en chile. Literatura y lingüística, pp. 295–324 (2019) doi: 10.29344/0717621x.40.2083
10. Nájar, A.: Así son las mañaneras, la novedosa estrategia para gobernar de AMLO en México. BBC News Mundo, (2019)
11. Röder, M., Both, A., Hinneburg, A.: Exploring the space of topic coherence measures. pp. 399–408 (2015) doi: 10.1145/2684822.2685324
12. Rosner, F., Hinneburg, A., Röder, M., Nettling, M., Both, A.: Evaluating topic coherence measures (2014) doi: 10.48550/arXiv.1403.6397
13. Sarfo, E., Krampa, E. A.: Language at war: A critical discourse analysis of speeches of Bush and Obama on terrorism. International Journal of Social Sciences and Education, vol. 3, no. 2 (2012)
14. Syed, S., Spruit, M.: Full-text or abstract? examining topic coherence scores using latent dirichlet allocation. In: 2017 IEEE International Conference on Data Science and Advanced Analytics. pp. 165–174 (2017) doi: 10.1109/DSAA.2017.61
15. Tausczik, Y. R., Pennebaker, J. W.: The psychological meaning of words: Liwc and computerized text analysis methods. Journal of language and social psychology, vol. 29, no. 1, pp. 24–54 (2010) doi: 10.1177/0261927X09351676
16. Torío, M. Á. R.: Características del lenguaje político: La designación. Philologia Hispalensis, no. 10, pp. 7–22 (1995) doi: 10.12795/ph.1995.v10.i01.01

ISSN 1870-4069

Seguimiento de patrones geométricos en tiempo real para la mejora de habilidades psicomotoras en cirugía laparoscópica

Víctor Manuel García Negrete, Antonio Alarcón Paredes,
Gustavo Adolfo Alonso Silverio

Universidad Autónoma de Guerrero,
Facultad de Ingeniería, Maestría en Ingeniería para la Innovación y Desarrollo Tecnológico (MIIDT), Chilpancingo de los Bravo, Guerrero,
México

{victorgarcia, aalarcon, gsilverio}@uagro.mx

Resumen. En la búsqueda constante de mejorar los estándares de calidad en la cirugía laparoscópica o cirugía mínimamente invasiva (MIS: Minimally Invasive Surgery), se han desarrollado programas de entrenamiento objetivos que permiten adquirir habilidades psicomotoras quirúrgicas del cirujano en formación, antes de los niveles de especialidad. Si bien existen dispositivos para tareas o ejercicios educativos utilizando tecnologías como realidad aumentada o realidad virtual, la mayoría no cuenta con una evaluación en tiempo real y comúnmente usan software de licencia comercial. En el presente trabajo se presenta un sistema de software desarrollado en Python, con OpenCV y Tkinter, que implementa tareas de seguimiento de patrones geométricos en una Raspberry Pi 4B. Se obtuvieron datos de 10 estudiantes de medicina, quienes llevan a cabo sesiones de entrenamiento por cada patrón geométrico seleccionado. Los resultados indican una mejora significativa en el desempeño de la tarea conforme pasan los días. Lo anterior da pie a obtener datos de especialistas e incorporar algoritmos de deep learning para establecer una clasificación objetiva de cirujanos expertos y sin experiencia.

Palabras clave: Cirugía mínimamente invasiva, entrenador laparoscópico, visión computacional, habilidades quirúrgicas, Python.

Tracking Geometric Patterns in Real Time for Improving Psychomotor Skills in Laparoscopic Surgery

Abstract. In the constant search to improve quality standards in laparoscopic surgery or minimally invasive surgery (MIS: Minimally Invasive Surgery), objective training programs have been developed that allow the acquisition of psychomotor skills in the surgical experience of the surgeon in training, before of specialty levels. Although there are devices for the completion of educational tasks or exercises that use technologies such as augmented reality or virtual reality, most do not have a real-time evaluation and usually use commercially licensed software. This article presents a software system developed in Python, with OpenCV and Tkinter, that implements a task of tracking geometric patterns

generated by the system on a Raspberry Pi 4B. Data were obtained from 10 medical students, who carry out training sessions for each selected geometric pattern. The results indicate a significant improvement in task performance as the days pass by. As future work, we will obtain data from specialists and incorporate deep learning algorithms to perform an objective classification of expert and inexperienced surgeons.

Keywords: Minimally invasive surgery, laparoscopic trainer, computer vision, surgical skills, python.

1. Introducción

Desde hace algunos años, se han realizado esfuerzos por mejorar los estándares de calidad en la práctica de la cirugía laparoscópica o mínimamente invasiva (MIS: Minimally Invasive Surgery). En este contexto, han surgido programas que permiten que el cirujano en entrenamiento vaya desarrollando sus habilidades psicomotoras en torno a la práctica de la MIS desde antes de llegar a los niveles de especialidad [1].

Para lograr lo anterior, se han desarrollado dispositivos educativos llamados comúnmente *box trainers*, o entrenadores de caja, que permiten llevar a cabo ejercicios que fomenten dichas habilidades en el cirujano, operando en conjunto con algún sistema de software diseñado exprofeso [2].

Dentro de los entrenadores de caja es posible encontrar ejemplos en los que sus tareas son totalmente físicas [3, 4], algunos que utilizan realidad virtual [5-7] y otros tantos, realidad aumentada [8, 9]. Aún con sus diferencias, todos ellos poseen tareas estándares descritas en el Sistema Inanimado MISTELS [10].

El objetivo de los entrenadores es mejorar las habilidades laparoscópicas, incorporando tareas como transferencia de objetos, corte, ligadura, sutura con nudo intracorpóreo y sutura extracorpórea.

Para este propósito, existen ya algunos entrenadores consolidados en la literatura especializada, tal como EndoVis (Endoscopic Orthogonal Video System): un sistema de formación y evaluación objetiva de las habilidades psicomotrices y destrezas quirúrgicas de los cirujanos [11, 12]; EVA (Endoscopic Video Analysis): un sistema para obtener los movimientos de los instrumentos laparoscópicos basado en el seguimiento de video [13]; TrEndo: un dispositivo de bajo costo con cuatro grados de libertad para el rastreo de instrumentos quirúrgicos mínimamente invasivos [14]; o el de Allen [15], un nuevo método que combina múltiples métricas discretas de análisis de movimientos que se enfocan en la evaluación y clasificación automática de las habilidades laparoscópicas de los cirujanos en formación. En todos ellos, es común que el resultado del desempeño se dé de forma posterior.

Sin embargo, todos ellos incorporan directamente una tarea de las realizadas en una intervención quirúrgica laparoscópica. La idea principal de este trabajo es la de diseñar estrategias para desarrollar en el usuario habilidades como la percepción espacial y de movimiento, así como la coordinación mano-ojo, ya que es común que al comenzar a usar los entrenadores el usuario presente dificultades debido a que la retroalimentación visual se da en una pantalla 2D y en una escala mayor al original.

Fig. 1. Entrenador de caja.

En el presente artículo, se presenta un sistema de realidad aumentada desarrollado en Python, con OpenCV y Tkinter, que implementa una tarea de seguimiento de patrones geométricos generados por sistema, en una Raspberry Pi 4B y que se integran en un entrenador de caja.

Para brindar el resultado en tiempo real, se mide el error relativo porcentual entre la guía geométrica dada y el movimiento hecho por el usuario durante una sesión de entrenamiento por repeticiones. Si bien el entrenamiento de este tipo de habilidades se había planteado en [16], utilizan una sola figura geométrica.

Este sistema pretende ser una opción para el entrenamiento de cirujanos aprendices en formación y su mejora de habilidades quirúrgicas en cirugía laparoscópica, proponiendo generar una mejor curva de aprendizaje.

2. Sistema de entrenamiento propuesto

2.1. Materiales

Para la implementación del sistema, se construyó un entrenador de caja con componentes de fácil acceso y no exclusivos del área médica (véase Fig. 1). El material utilizado para su construcción es MDF (Medium Density Fiberboard), un producto derivado de la madera seleccionado por ser resistente, duradero y de menor costo en comparación a la madera convencional. Incorpora dos cámaras colocadas de manera ortogonal.

Para mantener una adecuada iluminación en el espacio de trabajo se ha incorporado internamente una tira de LED de luz blanca. En aras de brindar mayor realismo al usuario, se utiliza una pinza grasper de la marca *Tyco Healthcare Autosuture*, mostrada en la Fig. 2, con un grosor de 5 mm y un largo de 33.5 cm aproximadamente. Esta pinza se ingresa al entrenador en uno de sus tres orificios utilizando un trocar como medio de sujeción. En la punta de la pinza, se coloca un marcador de color verde para su posterior detección con el software.

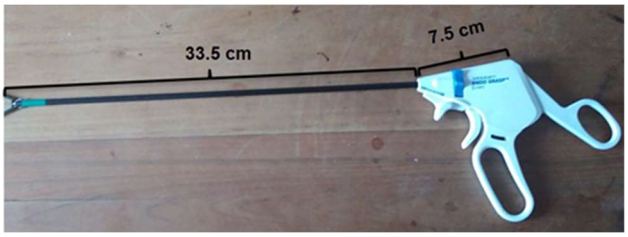


Fig. 2. Pinza grasper.

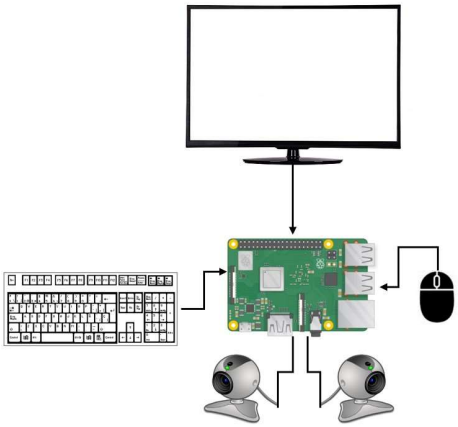


Fig. 3. Esquema de hardware.

2.2. Hardware

Con el objetivo de integrar un dispositivo capaz de realizar los ejercicios de seguimiento de patrones geométricos con ayuda de la visión computacional, se utilizó una plataforma embebida y componentes descritos a continuación (véase Fig. 3):

- Plataforma embebida: Permite ejecutar el sistema de software desarrollado. Se ha elegido la Raspberry Pi 4B ya que aporta una mayor capacidad de memoria RAM y mejor velocidad de procesamiento a comparación de otras.
- Pantalla: Para visualizar el software y los movimientos del usuario.
- Teclado: Para ingresar el nombre de los participantes.
- Mouse: Para seleccionar la figura sobre la cual se entrenará.
- Cámaras: Divididas en cámara maestra y esclava, las cuales permiten ocupar la visión artificial para llevar a cabo el entrenamiento.

2.3. Métodos

Los ejercicios de seguimiento de patrones geométricos fueron desarrollados en el lenguaje de programación Python utilizando para ello OpenCV, así como un entorno gráfico de usuario con Tkinter.

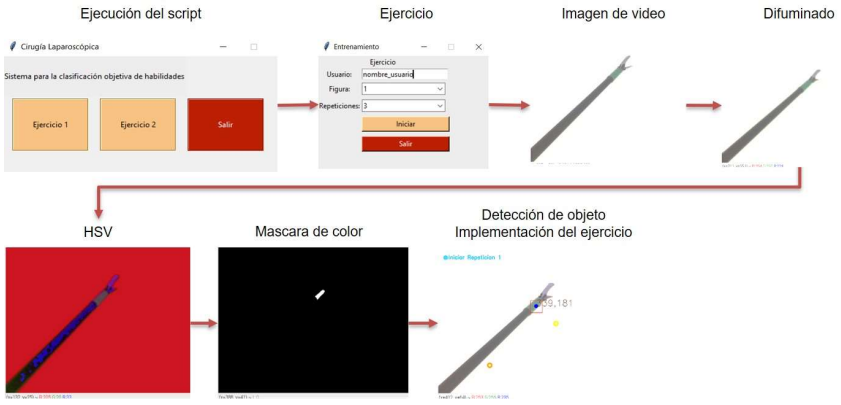


Fig. 4. Procesamiento de imagen de video.

Antes de llevar a cabo los ejercicios, se realiza la detección del instrumento laparoscópico por medio del marcador verde, a manera de calibración inicial. Para ello, lleva a cabo el procedimiento siguiente (véase Fig. 4):

- Se obtiene la imagen de video en RGB, con una configuración en dimensiones de 640 x 480 pixeles (ancho por alto).
- En seguida, se realiza un filtro de difuminado de mediana a la imagen de video, lo cual resulta útil para eliminar el ruido.
- Posteriormente, se realiza una conversión del espacio de color RGB a HSV, ya que HSV separa la saturación, del matiz y valor del color y se puede extraer el color de un objeto con mayor facilidad.
- Se obtienen umbrales HSV para identificar una gama de color verde.
- Finalmente, por medio de operaciones morfológicas y un elemento estructurante cuadrado, se elimina el ruido que pudiera existir en los umbrales de la imagen HSV.

Para iniciar el entrenamiento, se selecciona el ejercicio de seguimiento de patrones geométricos con el número de repeticiones deseadas; en este trabajo se tomaron en cuenta 5 repeticiones por participante.

Cada una de las figuras o patrones geométricos, tiene como finalidad la de entrenar al usuario en movimientos típicos de las siguientes tareas laparoscópicas:

- Figura 1. Recta: Tarea de transferencia de objetos
- Figura 2. Infinito: Sutura
- Figura 3. Elipse: Corte

A continuación, se detallan las acciones del usuario cuando ejecuta una sesión de entrenamiento, que también es descrito en la Fig. 5:

- Inicia la interfaz gráfica del sistema, donde el usuario deberá ingresar su nombre en el campo correspondiente. Seleccionara la figura o patrón geométrico, así como la cantidad de repeticiones por entrenamiento; en este trabajo se tomaron en cuenta 5 repeticiones por participante.

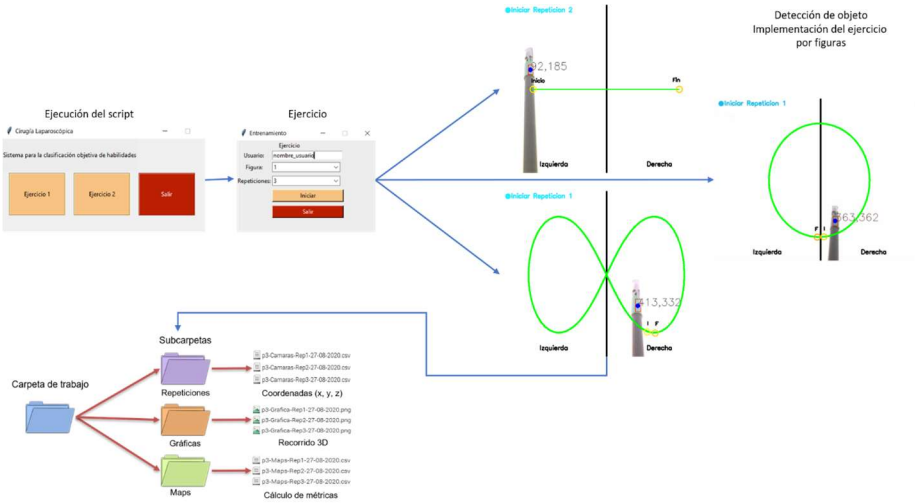


Fig. 5. Funcionamiento del ejercicio de seguimiento de patrones geométricos.

- El software dibuja en pantalla, la figura seleccionada.
- El usuario realiza el recorrido con el instrumento desde el punto de inicio establecido por el software identificado con un círculo amarillo y el punto final identificado con un círculo de anaranjado. La cámara superior obtiene las coordenadas en la misma visualización del usuario, mientras la cámara esclava tiene una vista lateral del instrumento, y obtiene la profundidad en la que se realizan los movimientos durante el recorrido.
- Cuando el marcador es colocado en el inicio, el tiempo es activado e inicia el recorrido por medio de la guía dibujada punto a punto. A la vez que se ejecutan tales acciones, las coordenadas del marcador son guardadas en un archivo CSV. Finaliza la repetición cuando el usuario llega al punto final, el sistema guarda las coordenadas y el tiempo en que se realizó el recorrido.
- Se muestra en pantalla la gráfica del patrón geométrico, así como la gráfica del recorrido hecho por el usuario para una retroalimentación visual.
- Se calcula el error absoluto porcentual para brindar el resultado de la evaluación, y también se provee el tiempo que tomó al usuario la tarea.

3. Resultados

La implementación del software con visión computacional que incorpora un ejercicio de seguimiento de patrones geométricos permite brindar un resultado en tiempo real. En este trabajo durante 5 días consecutivos se obtuvieron datos de 10 estudiantes de medicina, previamente capacitados para llevar a cabo 5 repeticiones de cada uno de los patrones geométricos del sistema propuesto con el uso del entrenador de caja.

Para ello, se mide el error relativo porcentual (1) entre la guía geométrica dada y las coordenadas del movimiento hecho por el usuario durante una sesión de entrenamiento de habilidades en cirugía laparoscópica por repeticiones.

$$ERP = \sum_{i=1}^{N} \left| \frac{V_r(i) - V_e(i)}{V_r(i)} \right| x\ 100, \quad (1)$$

en donde V_r es el valor ideal del recorrido por medio de las coordenadas de la guía geométrica, V_e es el valor estimado dado por las coordenadas del movimiento del usuario y N es el número total de coordenadas.

Estableciendo dicha fórmula de medición, se ha codificado dentro del software hecho en Python para que, al finalizar cada repetición en una sesión de entrenamiento, se genere el porcentaje del error relativo en una comparativa entre la guía geométrica y el movimiento realizado por el usuario. Se considera como sesión de entrenamiento, a la elección de la figura a utilizar para el ejercicio durante un numero establecido de repeticiones.

En la Fig. 6, se puede observar el patrón geométrico generado por el sistema de color verde y el recorrido de un usuario de color rojo, a su vez el cálculo del error relativo porcentual.

En la parte de arriba se muestra la primera repetición realizada y en la parte de abajo la última repetición, esto ha demostrado una mejoría en las habilidades psicomotoras del usuario como la precisión de movimiento y la coordinación mano-ojo, a través de una curva de aprendizaje generada por la ejecución de una sesión de entrenamiento de 5 repeticiones por cada una de las figuras presentadas.

En la Fig. 7 se muestra una gráfica del comportamiento promedio (líneas delgadas) por días en torno al error obtenido por los 10 participantes y el promedio global (línea gruesa), así como el tiempo que les tomó llevar a cabo la tarea de seguimiento.

Al observar las gráficas de error por cada figura se observa una disminución de éste conforme los días de entrenamiento transcurren; sin embargo, no sucede así para el tiempo requerido.

Es decir, si bien el tiempo permanece aproximadamente constante, la disminución del error es indicativo que el usuario comienza a coordinar mejor sus movimientos, adaptándose al entorno de trabajo y al espacio del entrenador. Esto sucede de forma similar para los tres patrones geométricos.

Además, con la finalidad de obtener una medida para identificar si existe una variabilidad significativa entre las repeticiones de los días de entrenamiento de los participantes, se llevó a cabo la prueba de Kruskal-Wallis tomando en cuenta como significativo un valor $p < 0.01$. Ésta se aplicó tanto a los valores de error promedio obtenidos por día, así como al tiempo promedio por día que necesitaron los usuarios para llevar a cabo la tarea de seguimiento. El resultado de esta prueba en cuanto al error, se obtiene un valor p = 1.5E-5, p = 5.4E-6, y p = 6.2E-10, para el patrón de recta, infinito y elipse respectivamente.

Como una medida global se calculó la prueba estadística del error tomando en cuenta los resultados de todas las figuras para cada día; el resultado fue un valor de p = 4.4E-10, lo cual habla de una diferencia significativa en el desempeño de los participantes al irse adaptando al entorno; esto puede verse en la Fig. 8. El valor p para el tiempo requerido por las tareas fue $p > 0.01$ en todos los casos, es decir no es significativo.

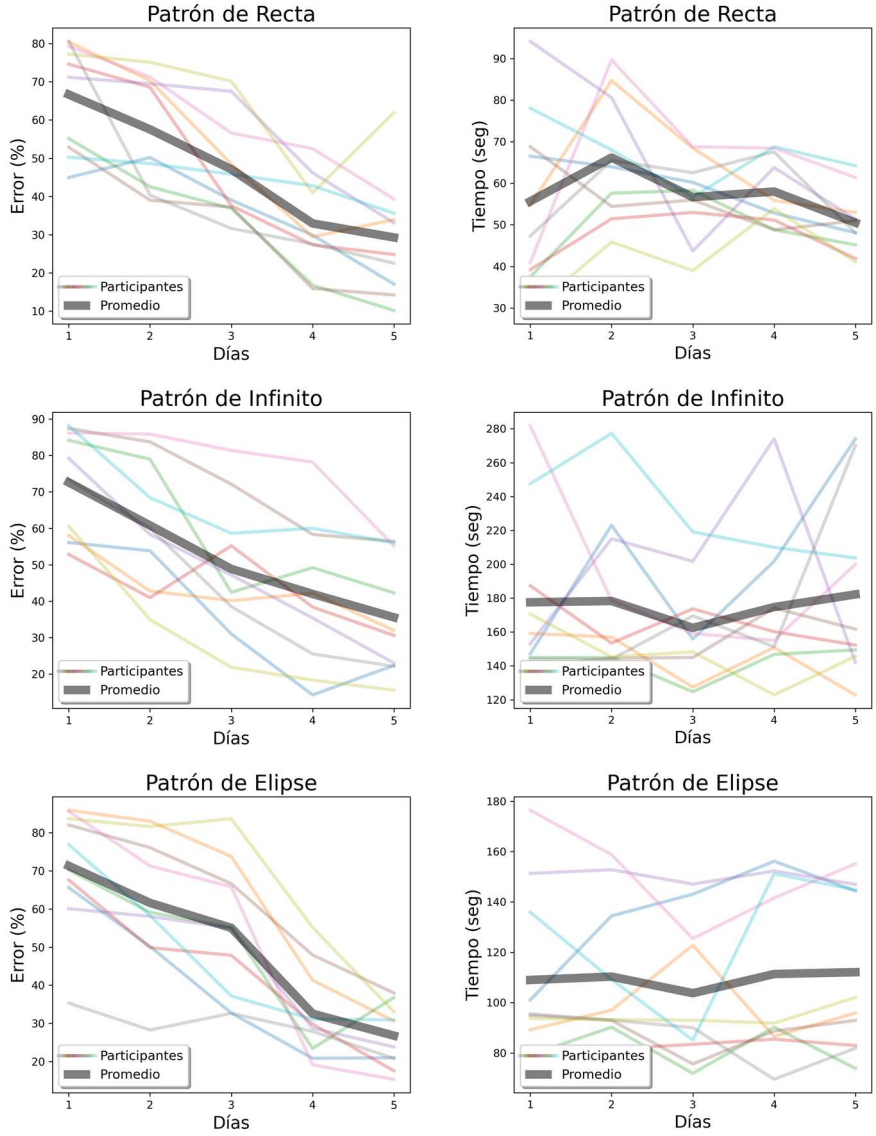


Fig. 7. Gráfica del error (izquierda) y tiempo (derecha) del seguimiento de patrones geométricos realizado por los participantes.

4. Conclusiones y trabajo a futuro

El desarrollo de sistemas de software con visión computacional se ha establecido como una herramienta útil para el ámbito de la salud, específicamente en la enseñanza médica de la MIS. Esto se debe a su gran aporte, a través del uso conjunto con un

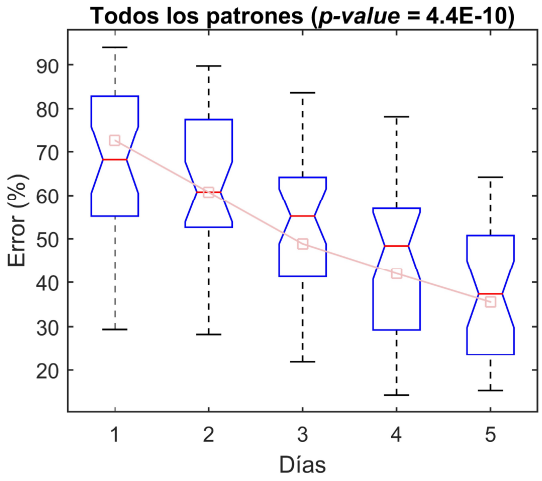


Fig. 8. Resultado de la prueba de Kruskal-Wallis durante los días de entrenamiento para los patrones de seguimiento.

dispositivo conocido como el box trainer o entrenador laparoscópico que permite a los estudiantes entrenar en un entorno virtual simulando los procedimientos reales.

Este sistema de software que incorpora ejercicios de seguimiento de patrones geométricos, pretende ser una opción en el entrenamiento de cirujanos aprendices en formación para la mejora de las habilidades quirúrgicas en cirugías laparoscópicas, proponiendo generar una mayor curva de aprendizaje y retroalimentación del nivel de competencia en tiempo real.

En este trabajo se ha mostrado cómo el hecho de llevar a cabo ejercicios que no son propiamente del tipo laparoscópico como se estipulan en el FLS, puede también tener beneficios en las habilidades de psicomotrices del cirujano en entrenamiento. La prueba estadística de Kruskal-Wallis indica que existen diferencias significativas en el desempeño de los participantes respecto de los tres patrones de seguimiento al pasar los días de entrenamiento.

Lo anterior da la pauta para seguir desarrollando el entrenador propuesto. El siguiente paso será obtener más datos de estudiantes, así como datos de especialistas del área, lo cual se ha complicado por el momento crítico de pandemia y confinamiento que estamos viviendo actualmente.

Con los datos de ambos grupos, experimentados y sin experiencia, será posible realizar pruebas utilizando algoritmos de deep learning, tales como las memorias largas a corto plazo (LSTM: Long short-term memories) o redes neuronales convolucionales (CNN: Convolutional neural networks).

Referencias

1. Harrysson, I., Hull, L., Sevdalis, N., Darzi, A., Aggarwal, R.: Development of a knowledge, skills, and attitudes framework for training in laparoscopic cholecystectomy. Am J Surgery, vol. 207, no. 5, pp. 790–796 (2014) doi: 10.1016/j.amjsurg.2013.08.049

2. García, G. A., Jiménez, G., Barrios, A. J., Guevara, R. E., Ruiz, J. P., Mendivelso, F. O.: El cambio del paradigma educativo en la enseñanza de la cirugía laparoscópica. Revista Colombiana de Cirugía, vol. 32, no. 1, pp. 40–44 (2017) doi: 10.30944/20117582.6
3. Laski, D., Stefaniak, T. J., Makarewicz, W., Proczko, M., Gruca, Z., Sledzinski, Z.: Structuralized box-trainer laparoscopic training significantly improves performance in complex virtual reality laparoscopic tasks. Wideochir Inne Tech Malo Inwazyjne, vol 7, no. 1, pp. 27–32 (2011) doi: 10.5114/wiitm.2011.25666
4. Hinata, N., Iwamoto, H., Morizane, S., Hikita, K., Yao, A., Muraoka, K, Honda, M, Isoyama, T., Sejima, T., Takenaka, A.: Dry box training with three-dimensional vision for the assistant surgeon in robot-assisted urological surgery. International Journal of Urology, vol. 20, no. 10, pp. 1037–1041 (2013) doi: 10.1111/iju.12101
5. Seymour, N. E.: VR to OR: A review of the evidence that virtual reality simulation improves operating room performance. World Journal of Surgery, vol. 32, pp. 182–188, (2008) doi: 10.1007/s00268-007-9307-9
6. Iwata, N., Fujiwara, M., Kodera, Y., Tanaka, C., Ohashi, N., Nakayama, G., Koike, M., Nakao, A.: Construct validity of the LapVR virtual-reality surgical simulator. Surg Endosc, vol. 25, no. 2, pp. 423–428 (2011) doi: 10.1007/s00464-010-1184-x
7. Willis, R. E., Gomez, P. P., Ivatury, S. J., Mitra, H. S., Van Sickle, K. R.: Virtual reality simulators: Valuable surgical skills trainers or video games? Journal of Surgical Education, vol. 71, pp. 426–433 (2014) doi: 10.1016/j.jsurg.2013.11.003
8. Viglialoro, R. M., Condino, S., Gesi, M., Ferrari, M., Ferrari, V.: Augmented reality simulator for laparoscopic cholecystectomy training. In: De Paolis, L., Mongelli, A. (eds) Augmented and Virtual Reality. AVR 2014, Lecture Notes in Computer Science, vol. 8853, Springer, Cham (2014) doi: 10.1007/978-3-319-13969-2_33
9. Botden, S. M., Buzink, S. N., Schijven, M. P., Jakimowicz, J. J.: ProMIS augmented reality training of laparoscopic procedures face validity, simulation in healthcare. The Journal of the Society for Simulation in Healthcare, vol. 3, no. 2, pp. 97–102 (2008) doi: 10.1097/SIH.0b013e3181659e91
10. Vassiliou, M. C., Ghitulescu, G. A., Feldman, L. S., Stanbridge, D., Leffondré, K., Sigman, H. H., Fried, G. M.: The MISTELS program to measure technical skill in laparoscopic surgery. Surgical Endoscopy And Other Interventional Techniques, vol. 20, pp. 744–747 (2006) doi: 10.1007/s00464-005-3008-y
11. Escamirosa, F. P., Flores, R. M. O., García, I. O., Vidal, C. R., Martínez, A. M. Face, content, and construct validity of the EndoViS training system for objective assessment of psychomotor skills of laparoscopic surgeons. Surgical Endoscopy, vol. 29, pp. 3392–3403 (2015) 10.1007/s0046 4-014-4032-6
12. Pérez-Escamirosa, F., Alarcón-Paredes, A., Alonso-Silverio, G. A., Oropesa, I., Camacho-Nieto, O., Lorias-Espinoza, D., Minor-Martínez, A.: Objective classification of psychomotor laparoscopic skills of surgeons based on three different approaches. In: Proceedings of International Journal of Computer Assisted Radiology and Surgery, vol. 15, pp. 27–40 (2020) doi: 10.1007/s11548-019-02073-2
13. Oropesa, I., Sánchez, P., Chmarra, M. K., Lamata, P., Fernández, A., SáncheZ, J. A., Jansen, F. W., Dankelman, J., Sánchez, F. M., Gómez, E. J.: EVA: Laparoscopic instrument tracking based on endoscopic video analysis for psychomotor skills assessment. Surgical Endoscopy, vol. 27, pp. 1029–1039 (2013) 10.1007/s00464-012-2513-z (2013)
14. Chmarra, M. K., Bakker, N. H., Grimbergen, C. A., Dankelman, J.: TrEndo, a device for tracking minimally invasive surgical instruments in training setups. Sens Actuators A Physical, vol. 126, no. 2, pp. 328–334 (2006) https://doi.org/10.1016/j.sna.2005.10.040
15. Allen, B., Nistor, V., Dutson, E., Carman, G., Lewis, C., Faloutsos, P.: Support vector machines improve the accuracy of evaluation for the performance of laparoscopic training tasks. Surgery Endoscopy , vol. 24, pp. 170–178 (2010) doi: 10.1007/s00464-009-0556-6

16. Alonso-Silverio, G. A., Perez-Escamirosa, F., Bruno-Sanchez, R., Ortiz-Simon, J. L., Muñoz-Guerrero, R., Minor-Martinez, A., Alarcón-Paredes, A.: Development of a laparoscopic box trainer based on open source hardware and artificial intelligence for objective assessment of surgical psychomotor skills. Surgical Innovation, vol. 25, pp. 380–388 (2018) 10.1177/1553350618777045

ISSN 1870-4069

Rendimiento del algoritmo basado en el forrajeo de bacterias con distribución uniforme, gaussiana y exponencial

Margarita Hernández-Hernández, Betania Hernández-Ocaña, José Adán Hernández-Nolasco, José Hernández-Torruco

Universidad Juárez Autónoma de Tabasco, México

192H13004@alumno.ujat.mx, {betania.hernandez, jose.nolasco, jose.hernandezt}@ujat.mx

Resumen. En este trabajo se analiza el comportamiento y rendimiento del algoritmo basado en el forrajeo de bacterias TS-MBFOA usando poblaciones de bacterias generadas con distribución gaussiana y exponencial. TS-MBFOA es una metaheurística de inteligencia colectiva que emula el comportamiento de las bacterias E.Coli, probado con éxito en problemas de optimización numérica con restricciones. Sin embargo, este algoritmo requiere de un costo computacional elevado debido a sus múltiples parámetros, por ende, en este trabajo se busca mejorar el rendimiento del algoritmo reinicializando la población de bacterias con distintas distribuciones. En este estudio preliminar el algoritmo fue probado en los problemas conocidos como la esfera y resorte de tensión/compresión, donde el algoritmo con distribución exponencial genera soluciones con mejor consistencia.

Palabras clave: Optimización, forrajeo de bacterias, metaheurística, distribución uniforme, distribución gaussiana, distribución exponencial.

Performance of the Algorithm based on the Foraging of Bacteria with Uniform, Gaussian and Exponential Distribution

Abstract. In this paper, the behavior and performance of the algorithm based on the bacterial foraging TS-MBFOA is analyzed using swarm of bacteria generated with gaussian and exponential distribution. TS-MBFOA is a swarm intelligence metaheuristic that emulates the behavior of E.Coli bacteria, successfully tested on constrained numerical optimization problems. However, this algorithm requires a high computational cost due to its multiple parameters, therefore, this work seeks to improve the performance of the algorithm by reinitializing the population of bacteria with different distributions. In this preliminary study the algorithm was tested in the problems known as the sphere and tension/compression spring, where the algorithm with exponential distribution generates solutions with better consistency.

Keywords: Optimization, bacterial foraging, metaheuristic, uniform distribution, gaussian distribution, exponential distribution.

1. Introducción

Los algoritmos bio-inspirados en la naturaleza son ampliamente usados para resolver problemas de optimización numérica con restricciones. La comunidad científica ha hecho uso de ellos para resolver problemas complejos de diversas áreas como medicina [5].

Estos algoritmos son conocidos como metaheurísticas, las cuales permiten resolver problemas de optimización con o sin restricciones, combinatorios o numéricos de manera aproximada, es decir generan una o más soluciones factibles cercanas al óptimo.

Un problema de optimización es también conocido como un problema general de programación no-lineal y se puede definir como: Minimizar ó Maximizar $f(\vec{x})$ sujeta a: $g_i(\vec{x}) \leq 0, i = 1, ..., m$ y/o $h_j(\vec{x}) = 0, j = 1, ..., p$.
donde $\vec{x} \in R^n$ tal que $n \geq 1$, es el vector de soluciones $\vec{x} = [x_1, x_2, ..., x_n]^T$, donde cada $x_i, i = 1, ..., n$ está delimitada por el límite inferior y superior $L_i \leq x_i \leq U_i$; m es el número de restricciones de desigualdad y p es el número de restricciones de igualdad (en ambos casos, las restricciones podrían ser lineales o no lineales). Si denotamos con F a la región factible (donde se encuentran todas las soluciones que satisfacen al problema) y con S a todo el espacio de búsqueda, entonces $F \subseteq S$.

Los algoritmos bio-inspirados se dividen en dos grandes grupos, los Algoritmos Evolutivos (AEs), cuyo funcionamiento se basa en emular el proceso de evolución natural y la supervivencia del más apto y los algoritmos de Inteligencia Colectiva (IC) se basan en el comportamiento social y cooperativo de organismos simples e inteligentes como insectos y aves [6].

En el año 2002 K. Passino propone al algoritmo del forrajeo de bacterias (BFOA, por sus siglas en inglés), en el cual cada bacteria trata de maximizar su energía obtenida por unidad de tiempo empleada en el proceso de forrajeo, donde también evade sustancias nocivas. Más aun, las bacterias se pueden comunicar entre sí mediante la segregación de sustancias [10].

Su funcionamiento consiste en cuatro pasos: quimiotáxis (movimientos de nados-giros), agrupamiento, reproducción y eliminación-dispersión. Una mejora de BFOA es el algoritmo BFOA modificado (MBFOA), el cual hace una disminución de los parámetros y ciclos del algoritmo [9]. Este algoritmo aplica un mecanismo para el manejo de las restricciones usando las reglas de factibilidad de Deb [3].

Otra propuesta del algoritmo es el Improved MBFOA donde la modificación implementa un mecanismo de sesgo para crear la población inicial, dos operadores de nado, tamaños de paso dinámico y el buscador local [6].

Recientemente, el algoritmo TS-MBFOA, una propuesta basada en MBFOA, intercala dos nados en el proceso quimiotáxico, el primero es el nado original con tamaño de paso aleatorio y el segundo nado incluye el operador de mutación usado en los algoritmos evolutivos para mejorar la capacidad de exploración y explotación del algoritmo [7].

Uno de los inconvenientes de esta propuesta es la convergencia prematura donde algunas veces el algoritmo puede caer en óptimos locales, esto debido a la poca diversidad en el cúmulo de bacterias provocado por los procesos de agrupamiento y reproducción del algoritmo.

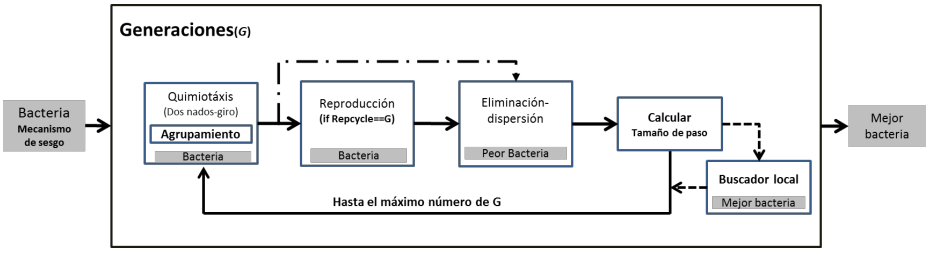


Fig. 1. Procesos generales de TS-MBFOA.

Las propuestas MBFOA, IMBFOA, TS-MBFOA hacen uso de la distribución uniforme para generar cada una de las bacterias de la población generacional del algoritmo.

Revisando el estado del arte, algunas propuestas de algoritmos bio-inspirados hacen uso de distintas distribuciones como gaussiana, gamma y exponencial para mejorar su rendimiento con la prevalencia de diversidad en la población lo que conlleva una mejor distribución de la población en el espacio de búsqueda.

En la propuesta realizada por Chen et al. [2], se hace uso de una mutación gaussiana para mejorar el rendimiento de la propuesta basada en BFOA con el objetivo de aumentar la diversidad de la población y evitar una convergencia prematura.

En esta propuesta los autores generan las nuevas posiciones de las bacterias usando una longitud de paso y la mutación gaussiana que consiste en generar un número aleatorio entre [0,1] con distribución gaussiana que escala la posición actual de la mejor bacteria de la población.

donde los autores obtuvieron competitivos resultados, al resolver problemas de pruebas clásicos, con mayor precisión y una mejor convergencia comparando contra la versión original de BFOA y otras metaheurísticas.

En las propuestas [14, 1, 13, 11], se hace uso de la distribución gaussiana y gamma para mejorar la diversidad de la población de algoritmos evolutivos, específicamente hacen uso de estas distribuciones para el operador de mutación.

En este artículo se propone hacer un reinicio a la mitad de la población de bacterias, cuando la mitad de la población de bacterias sean factibles, es decir, cuando no se violen restricciones del problema a resolver. Este reinicio se hará en una única ocasión durante las generaciones del algoritmo.

La generación de la nueva población de bacterias hace uso de la información de todas las bacterias factibles de la población, para la distribución gaussiana y exponencial el valor de la media es obtenido del conjunto de las bacterias factibles de la población.

Esto con el objetivo de que las nuevas bacterias estén dentro o cerca de la región factible del problema a resolver y evitar generar bacterias fuera del espacio de búsqueda del problema.

2. Two-Swim Modified Bacterial Foraging Optimization Algorithm (TS-MBFOA)

El TS-MBFOA es un algoritmo derivado de MBFOA propuesto para resolver PONR [4], en el cual una bacteria i representa una solución potencial y se denota como $\theta^i(j, G)$, donde j es el ciclo quimiotáxico y G es el ciclo generacional.

Una generación consta de un proceso quimiotáxico, agrupamiento, reproducción y eliminación-dispersión. En el proceso de quimiotáxis dos nados se intercalan, en cada ciclo solo un nado de explotación o exploración es realizado. El proceso comienza con el nado de explotación (nado clásico).

Sin embargo, una bacteria no necesariamente intercalará exploración y explotación en los nados, ya que si la nueva posición de un nado dado, $\theta^i(j+1, G)$ tiene una mejor aptitud (basado en las reglas de factibilidad) que la posición original $\theta^i(j, G)$, otro nado en la misma dirección se llevará a cabo en el siguiente ciclo.

De lo contrario, un nuevo giro será calculado.n El proceso se detiene después de N_c intentos. El nado de exploración usa la mutación entre bacterias y es calculado con la Ecuación 1:

$$\theta^i(j+1, G) = \theta^i(j, G) + (\beta)(\theta_1^r(j, G) - \theta_2^r(j, G)), \tag{1}$$

donde $\theta_1^r(j, G)$ y $\theta_2^r(j, G)$ son dos bacterias diferentes seleccionadas aleatoriamente de la población. β es un parámetro definido por el usuario utilizado en el operador de agrupamiento el cual define la cercanía de la nueva posición de una bacteria con respecto a la posición de la mejor bacteria de la población, en este operador, β es un parámetro de control positivo para escalar los diferentes vectores en (0,1], es decir, escalas de la zona donde una bacteria puede moverse. El nado de explotación es calculado con el Ecuación 2:

$$\theta^i(j+1, G) = \theta^i(j, G) + C(i, G)\phi(i), \tag{2}$$

donde la dirección del paso $\phi(i)$ se calcula con el operador de giro original de BFOA definido en la Ecuación 3:

$$\phi(i) = \frac{\Delta(i)}{\sqrt{\Delta(i)^T \Delta(i)}}, \tag{3}$$

donde $\Delta(i)$ es un vector aleatorio generado con elementos dentro de un intervalo$[-1, 1]$. $C(i, G)$ es el tamaño de paso aleatorio de cada bacteria actualizado con la Ecuación 4:

$$C(i, G) = R * \Theta(i), \tag{4}$$

donde $\Theta(i)$ es un vector generado de forma aleatoria de tamaño n con elementos dentro del rango de cada variable de decisión: $[U_k, L_k]$, $k = 1, ...n$, y R es un parámetro definido por el usuario para escalar el tamaño de paso, este valor debe ser cercana a cero (por ejemplo 5.00e-04). La inicial $C(i, 0)$ se genera utilizando $\theta(i)$. Este tamaño de paso aleatorio permite que las bacterias se puedan mover en diferentes direcciones dentro del espacio de búsqueda y evita la convergencia prematura como se sugiere en [8].

Algoritmo 1: Pseudocódigo de TS-MBFOA.

```
Crear una población inicial de bacterias aleatorias θ^i(j, 0)∀i, i = 1, ..., S_b
Evaluar f(θ^i(j, 0))∀i, i = 1, ..., S_b
for G = 1 to GMAX do
    for i = 1 to S_b do
        for j = 1 to N_c do
            En el proceso quimiotáxico intercalar los nados propuestos con las
            Ecuaciones 1 y 2. Aplicar el operador de agrupamiento con la Ecuación 5
            usando β para la bacteria θ^i(j, G)
        end
    end
    if G mod RepCycle == 0 then
        Realizar el proceso de reproducción ordenando la población de acuerdo a las
        reglas de factibilidad y eliminar a S, peores bacterias y duplicar el resto de
        bacterias S − S_r
    end
    Realizar el proceso de eliminación-dispersión eliminando a la peor bacteria
    θ^ω"(j, G) de la población actual considerando la técnica de
    manejo de restricciones.
    Actualizar el vector de tamaño de paso usando la Ecuación 4.
end
```

En el ciclo medio del proceso quimiotáxico es aplicado el operador de agrupamiento con la Ecuación 5, donde β es un parámetro positivo definido por el usuario entre (0,1):

$$\theta^i(j+1,G) = \theta^i(j,G) + \beta(\theta^B(G) - \theta^i(j,G)), \tag{5}$$

donde $\theta^i(j+1,G)$ es la nueva posición de la bacteria i, $\theta^B(G)$ es la actual posición de la mejor bacteria generacional y β es un parámetro llamado factor de escalamiento, el cual regula qué tan cerca estará la bacteria i de la mejor bacteria θ^B.

Sin embargo, si una solución viola el límite de las variables de decisión, una nueva solución de x_i es generada aleatoriamente entre los límites inferior y superior $L_i \leq x_i \leq U_i$ de las variables de decisión.

En la reproducción se ordenan las bacterias con base en la técnica de manejo de restricciones, eliminando a las peores bacterias $S_b - S_r$ y duplicando a las mejores cada cierto número de ciclos, definido por el usuario con el parámetro $\mathrm{RepCycle}$.

En la eliminación-dispersión se elimina a la peor bacteria de la población $\theta^w(j,G)$ (basado en las reglas de factibilidad) y se genera una nueva aleatoriamente. Aunque en su propuesta original de TS-MBFOA se utiliza un mecanismo de sesgo para generar la población inicial aleatoria y un buscador local, en este artículo no se hace uso de dicho mecanismo para consumir menos costo computacional.

Cabe recalcar que el manejo de restricciones de este algoritmo se hace mediante las reglas de factibilidad de Deb [3], las cuales fueron incluidas desde la versión MBFOA. La estructura del TS-MBFOA se presenta en la Figura 1. El pseudocódigo de TS-MBFOA es presentado en el algoritmo 1.

3. Distribuciones

3.1. Distribución uniforme

La distribución uniforme es una distribución continua que modela un rango de valores con igual probabilidad y se especifica mediante cotas inferior y superior [a, b]. Si $x \sim u(a, b)$, su función de distribución viene dada por:

$$F(x) = \begin{cases} 0 & \text{para} \quad x < a, \\ \dfrac{x-a}{b-a} & \text{para} \quad a \leq x < b, \\ 1 & \text{para} \quad x \geq b. \end{cases} \tag{6}$$

Aplicando el método de inversión de la función de distribución se obtiene el siguiente esquema:

1. Generar un número aleatorio u.
2. Tomar $x = a + u(b - a)$.

3.2. Distribución gaussiana

La distribución Gaussiana o normal es, sin duda, la distribución de probabilidad más importante del Cálculo de probabilidades y de la Estadística.

Fue descubierta, como aproximación de la distribución binomial, por Abraham De Moivre (1667-1754) y publicada en 1733 en su libro The Doctrine of Chances; estos resultados fueron ampliados por Pierre-Simon Laplace (1749-1827), quién también realizó aportaciones importantes.

En 1809, Carl Friedrich Gauss (1777-1855) publicó un libro sobre el movimiento de los cuerpos celestes donde asumía errores normales, por este motivo esta distribución también es conocida como distribución Gaussiana.

La importancia de la distribución normal queda totalmente consolidada por ser la distribución límite de numerosas variables aleatorias, discretas y continuas, como se demuestra a través de los teoremas centrales del límite [12].

Teorema Central del límite especifica que: Sean $x_1, ..., x_n$ variables aleatorias independientes e idénticamente distribuidas con media μ y desviación típica σ. Entonces, la distribución de:

$$\frac{\sum_{i=1}^{n} x_i - n\mu}{\sigma\sqrt{n}}, \tag{7}$$

Es aproximadamente $N(0, 1)$ cuando el tamaño muestral n es suficientemente grande. La distribución normal queda totalmente definida mediante dos parámetros: la media (μ) y la desviación estándar o desviación típica (σ). Su función de densidad es simétrica respecto a la media y la desviación estándar nos indica el mayor o menor grado de apertura de la curva que, por su aspecto, se suele llamar campana de Gauss. Esta distribución se denota por N (μ, σ).

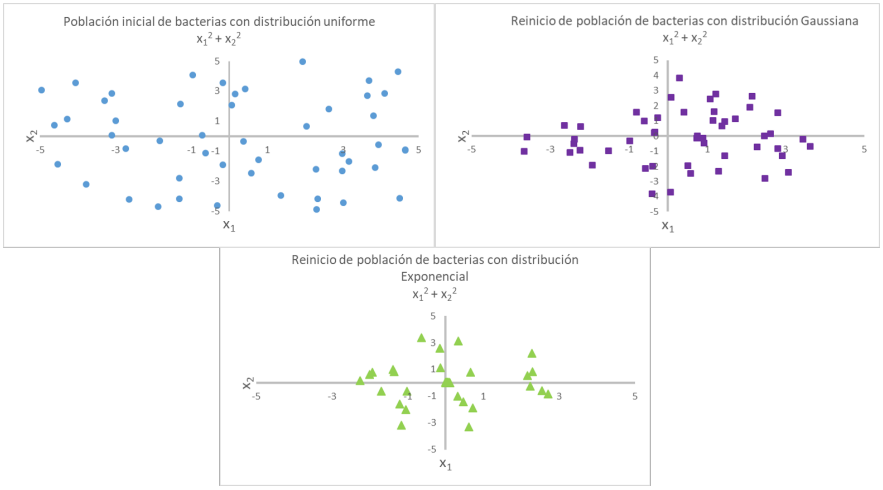


Fig. 2. Población de bacterias con distribución uniforme, gaussiana y exponencial.

La distribución $N(\mu, \sigma)$ se puede relacionar con la distribución $N(0, 1)$, mediante el siguiente proceso al que se denomina tipificación o estandarización:

$$X\ N(\mu, \sigma) \Rightarrow Z = \frac{X - \mu}{\sigma}. \tag{8}$$

A la distribución $N(0, 1)$ se le denomina Normal Estándar. En diversos lenguajes de programación como java, existen funciones programadas que permiten generar un número aleatorio con distribución normal con media y desviación estándar de cero. El siguiente esquema permite generar un número aleatorio con distribución normal con media y desviación estándar definida por el usuario final:

1. Generar un número aleatorio u con distribución $N(0, 1)$.
2. Definir los valores de σ y μ de un conjunto de valores.
3. Tomar $x = u \times \sigma + \mu$.

3.3. Distribución exponencial

La distribución exponencial es una distribución continua que se utiliza para modelar tiempos de espera para la ocurrencia de un cierto evento. Esta ley de distribución describe procesos en los que interesa saber el tiempo hasta que ocurre determinado evento; en particular, se utiliza para modelar tiempos de supervivencia [12].

Esta distribución se puede caracterizar como la distribución del tiempo entre sucesos consecutivos generados por un proceso de Poisson; por ejemplo, el tiempo que transcurre entre dos heridas graves sufridas por una persona. La media de la distribución de Poisson, λ, que representa la tasa de ocurrencia del evento por unidad de tiempo, es el parámetro de la distribución exponencial, y su inversa es el valor medio de la distribución.

Tabla 1. Mejor bacteria del algoritmo TS-MBFOA con diferentes distribuciones.

TS-MBFOA con distribución	x_1	x_2	$f(\vec{x}) = x_1^2 + x_2^2$
Uniforme	1.81824E-12	-3.86054E-12	1.82097E-23
Gaussiano	**4.02235E-15**	**-1.85888E-15**	**1.96347E-29**
Exponencial	3.80961E-14	4.19444E-14	3.21064E-27

Para generar un valor de una variable aleatoria X con distribución exponencial y parámetro λ a partir de un valor de una variable aleatoria $u = u(0,1)$ se utiliza el método de la transformada inversa para obtener la siguiente ecuación:

$$X = -\frac{1}{\lambda} \ln (1 - u) . \tag{9}$$

Utilizando el hecho de que si $u = u(0,1)$ entonces $1 - u = u(0,1)$. El esquema para generar un número aleatorio con distribución exponencial es:

1. Generar un número aleatorio u .
2. Tomar $x = -\frac{1}{\lambda} \ln(u)$.

3.4. TS-MBFOA con reinicio de población

En esta propuesta, el algoritmo TS-MBFOA es adaptado para reiniciar la mitad de la población, en este caso las peores bacterias, con una población aleatoria con distribución gaussiana o exponencial. El reinicio es único durante las generaciones del algoritmo y se lleva acabo cuando la mitad de las bacterias de la población es factible, es decir, no se violan restricciones.

Esto para garantizar que la nueva población de bacterias este cerca o dentro de la región factible del problema y evitar que se generen bacterias fuera del espacio de búsqueda determinado por el rango de las variables del problema a resolver. Inicialmente TS-MBFOA la población de bacterias es totalmente generada con distribución uniforme.

De igual manera, se incluye un validador y corrector de números aleatorios con el objetivo de generar números dentro del espacio de búsqueda del problema. Después de generar cada número aleatorio con distribuciones gaussiana o exponencial.

Para ello, cada número es validado verificando que se encuentre dentro del rango mínimo y máximo de cada variable de diseño del problema a resolver. Cuando un número aleatorio viola los rangos permitidos, un nuevo número aleatorio es generado con el mismo tipo de distribución usada. Para el caso de la distribución uniforme, una bacteria es generada con la Ecuación 10:

$$\theta^i(j, G) = L_i x_i + u * (U_i x_i - L_i x_i), \tag{10}$$

donde $L_i x_i$ es el límite inferior de la variable x_i, $U_i x_i$ es el límite superior de la variable x_i y u es un número aleatorio entre (0,1). Una bacteria con distribución gaussiana o normal es generada con la Ecuación 11:

$$\theta^i(j, G) = u * \sigma(x_i) + \mu(x_i), \tag{11}$$

Tabla 2. Estadísticas básicas de las 30 ejecuciones independientes de TS-MBFOA con diferentes distribuciones.

Criterios	TS-MBFOA uniforme	TS-MBFOA gaussiana	TS-MBFOA exponencial
Mejor	**0.012665672**	0.012665852	0.012665788
Media	0.012670502	0.012670288	**0.012669101**
Peor	0.012682834	0.012692327	**0.012680927**
Desv. Est.	4.25232E-06	5.1684E-06	**3.8474E-06**

donde u es un número aleatorio con distribución $N(0,1)$, en lenguaje java corresponde a la función nextGaussian(). μ es la media de la mitad de la población de bacterias factibles para cada variable x_i y σ es la desviación estándar de la mitad de la población de bacterias factibles para cada variable x_i. Una bacteria con distribución exponencial es generada con la Ecuación 12:

$$\theta^i(j,G) = \left(-\frac{1}{\lambda}\right)\ln(u), \tag{12}$$

donde u es un número aleatorio entre [0,1] y λ es la media de la mitad de la población de bacterias factibles para cada variable x_i. La media y desviación estándar utilizada en estas distribuciones hacen uso de la información obtenida por las bacterias hasta la generación donde la mitad de la población de bacterias es factible.

4. Resultados

El algoritmo TS-MBFOA con reinicio de población fue probado en dos problemas de pruebas y ejecutado una computadora con las características: Laptop de 4GB RAM, procesador de 2.3Ghz y un sistema operativo Windows de 64 bit. El lenguaje de programación fue java con el entorno de desarrollo integrado Netbeans IDE 8.0.2.

Dos experimentos fueron realizados para analizar el rendimiento del algoritmo con diferentes distribuciones. En el experimento 1, TS-MBFOA es probado con el problema conocido como la esfera con el objetivo de observar de manera gráfica la distribución de la población de bacterias con las diferentes distribuciones, cabe mencionar que el algoritmo TS-MBFOA inicia con una población con distribución uniforme.

El problema de la esfera se representa como $f(\vec{x}) = x_1^2 + x_2^2$, en este problema el rango de ambas variables es [-5, 5]. Los parámetros del algoritmo TS-MBFOA fueron calibrados con un conjunto de 50 ejecuciones independientes con cinco combinaciones de valores diferentes a los parámetros.

La mejor combinación de valores resultante fue: bacterias (S_b) = 50, tamaño de paso (R) = 0.0005, factor de escalamiento (β) = 1.8, ciclo quimiotáxico (N_c) = 12, bacterias a reproducir (S_r) = 1, frecuencia de reproducción ($\mathrm{RepCycle}$) = 100 y número de evaluaciones = 15,000.

En la Figura 3.2 se presenta la población inicial de bacterias del algoritmo TS-MBFOA en tres ejecuciones independientes. En el problema de la esfera es posible graficar sus variables debido a que solo cuenta con dos dimensiones.

Tabla 3. P-value de la prueba Wilcoxon signed rank test aplicada a los resultados de TS-MBFOA con diferentes distribuciones.

TS-MBFOA uniforme vs exponencial	**TS-MBFOA uniforme vs gaussiana**	**TS-MBFOA exponencial vs gaussiana**
0.05744	0.68916	0.28914

Este problema no tiene restricciones de igualdad o desigualdad, por lo tanto, el reinicio de población con distribución gaussiana y exponencial es llevada a cabo en la generación 10 del algoritmo para efectos de visualizar como quedan dispersas las bacterias en el espacio de búsqueda.

Como se puede observar en las gráficas, la población de bacterias con distribución exponencial se encuentra más agrupada hacia el origen, cabe mencionar, que el valor óptimo de la función de la esfera se encuentra en dicho lugar, cuando x_1 y x_2 toman el valor de cero.

La población de bacterias con distribución uniforme es la que abarca más lugares en el espacio de búsqueda, lo cual permite una mayor diversidad en la población, sin embargo, esto puede alentar al algoritmo a encontrar las zonas prometedoras donde se encuentre el óptimo global del problema.

Finalmente, la población de bacterias con distribución gaussiana hace el efecto de campana, generando pocas bacterias en los límites del espacio de búsqueda y concentra a muchas en la parte central de los límites de ambas variables de diseño.

En la Tabla 1 se presenta a la mejor bacteria de la población al finalizar la ejecución del algoritmo TS-MBFOA con diferentes distribuciones, donde el algoritmo TS-MBFOA con distribución gaussiana obtuvo el mejor resultado.

En el experimento 2, el algoritmo TS-MBFOA con diferentes distribuciones fue probado en el problema del resorte de tesión/compresión, donde se busca minimizar el peso de un resorte sujeto a restricciones de desviación mínima, tensión de corte, frecuencia de oleada, límites sobre el diámetro exterior, esto sobre variables de diseño.

El diseño de la función para ser procesadas en el algoritmo TS-MBFOA cuenta con las siguientes variables de diseño: Diámetro del cable d(x_1), Diámetro del rollo D(x_2), y el número de rollos involucrados N(x_3). Formalmente, el problema puede expresarse de la siguiente manera:

$$(x_3 + 2)x_2\ x_1^2. \tag{13}$$

Sujeto a:

$$g_1(X) = 1 - ((x_2^3\ x_3)/(71785\ x_1^4)) \leq 0, \tag{14}$$

$$g_2(X) = ((4\ x_2^2 - x_1\ x_2)/12566(x_2\ x_1^3 - x_1^4))) + (1/5108\ x_1^2) - 1 \leq 0, \tag{15}$$

$$g_3(X) = 1 - (140.45\ x_1/x_2^2\ x_3) \leq 0, \tag{16}$$

$$g_4(X) = (x_2 + x_1/1.5) - 1 \leq 0, \tag{17}$$

donde:

$$\begin{aligned} 0.05 &\leq x_1 \leq 2, \\ 0.25 &\leq x_2 \leq 1.3, \\ 2 &\leq x_3 \leq 15. \end{aligned} \tag{18}$$

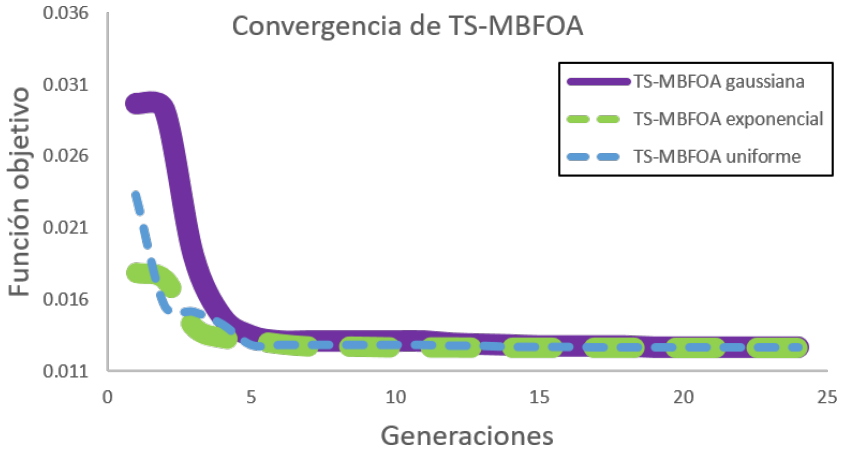


Fig. 3. Convergencia de TS-MBFOA con distribución uniforme, gaussiana y exponencial.

El algoritmo TS-MBFOA fue ejecutado en 30 corridas independientes con los mismos valores a los parámetros usados en el experimento 1. Los resultados estadísticos de las ejecuciones se presentan en la Tabla 2 donde la versión del TS-MBFOA con distribución uniforme obtuvo la mejor solución al problema, sin embargo, el algoritmo TS-MBFOA con reinicio de población exponencial obtuvo los mejores resultados en media, peor y desviación estándar.

La prueba no paramétrica de Wilcoxon signed rank test fue aplicada con un 95 % de certeza sobre el conjunto de las 30 mejores soluciones encontradas por cada versión del algoritmo TS-MBFOA. Los resultados indican que no hay una diferencia significativa al comparar las tres versiones del algoritmo TS-MBFOA como se muestra en la Tabla 3.

La gráfica de convergencia del algoritmo TS-MBFOA es presentada en la Figura 4. Los datos usados para generar las gráficas corresponden a la ejecución número 15, la cual es la mediana de las 30 ejecuciones independientes realizadas. En esta gráfica se puede observar que el algoritmo TS-MBFOA con distribución exponencial converge de manera más rápida antes de la generación número 5.

Los algoritmos con distribución uniforme y gaussiana convergen después de la generación número 5, aunque en la versión con distribución uniforme se logra apreciar como el algoritmo converge en algún óptimo local antes de la generación número 5, sin embargo, logra salir y converger en el óptimo global.

En general, el algoritmo TS-MBFOA obtiene resultados competitivos para el problema de prueba con las diferentes distribuciones. La versión con distribución exponencial obtiene resultados con menos desviación estándar, lo cual permite inferir que es más estable que las otras versiones.

5. Conclusión

El algoritmo basado en el forrajeo de bacterias E.Coli TS-MBFOA fue probado con distintas distribuciones. La versión original del algoritmo probado crea su población de bacterias con distribución uniforme.

En este artículo se propuso reiniciar la mitad de la población de bacterias (peores) con una población de bacterias aleatorias con distribuciones gaussiana y exponencial con el objetivo de observar el rendimiento del algoritmo en problemas de optimización numérica con restricciones.

El reinicio de la población se lleva acaba en una sola ocasión cuando la mitad de población de bacterias es factible debido a que se hace uso de la media y desviación estándar de esta mitad de bacterias para poder generar las nuevas bacterias con distribución gaussiana o exponencial.

Dos experimentos fueron realizados, en el primero el problema de la esfera fue implementado para observar de manera gráfica como se distribuyen las bacterias en el espacio de búsqueda. En el segundo experimento, TS-MBFOA fue probado en un problema de diseño ingenieril conocido como resorte de tensión/compresión.

30 ejecuciones independientes fueron realizadas a las tres versiones del algoritmo con distribución uniforme, gaussiana y exponencial. Donde los resultados presentan una mejor consistencia con la versión del TS-MBFOA con distribución exponencial.

Sin embargo, la prueba no paramétrica de Wilcoxon signed rank test con el 95 % de confianza determinó que no hay diferencia significativa entre los resultados de las tres versiones del algoritmo.

Como trabajo futuro, se espera probar cada una de las versiones del algoritmo TS-MBFOA con diferentes distribuciones en otros problemas de optimización numérica con restricciones y comprobar si el uso de otra distribución diferente de la uniforme permite una convergencia más rápida del algoritmo sin ser una convergencia prematura.

Referencias

1. Arabas, J., Opara, K.: Population diversity of nonelitist evolutionary algorithms in the exploration phase. IEEE Transactions on Evolutionary Computation, vol. 24, no. 6, pp. 1050–1062 (2020) doi: 10.1109/TEVC.2019.2917275
2. Chen, H., Zhang, Q., Luo, J., Xu, Y., Zhang, X.: An enhanced bacterial foraging optimization and its application for training kernel extreme learning machine. Applied Soft Computing, vol. 86, pp. 105884 (2020) doi: 10.1016/j.asoc.2019.105884
3. Deb, K.: An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering, vol. 186, no. 2-4, pp. 311–338 (2000) doi: 10.1016/s0045-7825(99)00389-8
4. Hernández-Ocaña, B., Pozos-Parra, M. D. P., Mezura-Montes, E., Portilla-Flores, E. A., Vega-Alvarado, E., Calva-Yáñez, M. B.: Two-swim operators in the modified bacterial foraging algorithm for the optimal synthesis of four-bar mechanisms. Computational Intelligence and Neuroscience, vol. 2016, pp. 1–18 (2016) doi: 10.1155/2016/4525294
5. Hernandez-Ocaña, B., Chavez-Bosquez, O., Hernandez-Torruco, J., Canul-Reich, J., Pozos-Parra, P.: Bacterial foraging optimization algorithm for menu planning. IEEE Access, vol. 6, pp. 8619–8629 (2018) doi: 10.1109/access.2018.2794198
6. Hernandez-Ocaña, B., Pozos-Parra, M. D. P., Mezura-Montes, E.: Improved modified bacterial foraging optimization algorithm to solve constrained numerical optimization problems. Applied Mathematics and Information Sciences, vol. 10, no. 2, pp. 607–622 (2016) doi: 10.18576/amis/100220
7. Hernández-Ocaña, B., Hernández-Torruco, J., Chávez-Bosquez, O., Calva-Yáñez, M., Portilla-Flores, E.: Bacterial foraging-based algorithm for optimizing the power generation

of an isolated microgrid. Applied Sciences, vol. 9, no. 6, pp. 1261 (2019) doi: 10.3390/app9061261

8. Kasaiezadeh, A., Khajepour, A., Waslander, S. L.: Spiral bacterial foraging optimization method: Algorithm, evaluation and convergence analysis. Engineering Optimization, vol. 46, no. 4, pp. 439–464 (2013) doi: 10.1080/0305215x.2013.776550
9. Mezura-Montes, E., Hernández-Ocaña, B.: Bacterial foraging for engineering design problems: Preliminary results. In: Memorias del 4to Congreso Nacional de Computación Evolutiva (2008)
10. Passino, K. M.: Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control Systems, vol. 22, no. 3, pp. 52–67 (2002) doi: 10.1109/mcs.2002.1004010
11. dos Santos Coelho, L., Ayala, H. V. H., Mariani, V. C.: A self-adaptive chaotic differential evolution algorithm using gamma distribution for unconstrained global optimization. Applied Mathematics and Computation, vol. 234, pp. 452–459 (2014) doi: 10.1016/j.amc.2014.01.159
12. de Galicia. bvón mantida e publicada en internet pola Consellería de Sanidade e o Servizo Galego de Saúde, X.: Ayuda de distribuciones de probabilidad. In: Distribuciones de Probabilidad. pp. 1–72 (2014)
13. Singh, P., Dwivedi, P., Kant, V.: A hybrid method based on neural network and improved environmental adaptation method using controlled Gaussian mutation with real parameter for short-term load forecasting. Energy, vol. 174, pp. 460–477 (2019) doi: 10.1016/j.energy.2019.02.141
14. Tirumala, S. S.: A quantum-inspired evolutionary algorithm using Gaussian distribution-based quantization. Arabian Journal for Science and Engineering, vol. 43, no. 2, pp. 471–482 (2017) doi: 10.1007/s13369-017-2641-9

ISSN 1870-4069

Diseño de una plantilla con materiales compuestos para prótesis de pie mediante algoritmos metaheurísticos

Derlis Hernández-Lara[1,2], Ricardo Gustavo Rodríguez-Cañizo[1], Emmanuel Merchán-Cruz[1], Emmanuel Tonatihu Juárez-Velázquez[1,2], Carlos Trejo-Villanueva[1,2]

[1] Instituto Politécnico Nacional,
Escuela Superior de Ingeniería Mecánica y Eléctrica
México

[2] Tecnológico Nacional de México,
Tecnológico de Estudios Superiores de Ecatepec
México

{dderlis-lara, emmanuel.juarez, carlostrejo}@tese.edu.mx,
{eamerchan, rgrodriguez}@ipn.mx

Resumen. Este trabajo presenta la utilización de un algoritmo metaheurístico, en el diseño de una plantilla para prótesis de pie hecha con materiales compuestos, con el fin de obtener los parámetros adecuados de diseño para soportar los esfuerzos específicos a los que estará sujeta. Se empleó el algoritmo de la colonia artificial de abejas. El problema en este tipo de diseños es encontrar el espesor adecuado de la pieza, lo que implica el número de láminas y las respectivas orientaciones de las fibras que la conformarán, para lo cual existen demasiadas posibles combinaciones al trabajar con laminados de *composites*. La metodología de diseño propuesta considera la teoría clásica de laminados, el proceso de fabricación y las cargas externas a las que estará sometida la plantilla para prótesis de miembro inferior. La principal contribución de este trabajo, es el uso de metaheurísticas para resolver problemas de diseño con *composites* planteados como problemas de optimización. Los experimentos realizados resultan en un diseño factible de este tipo de estructuras utilizando fibra de carbono/epoxi.

Palabras clave: Optimización evolutiva, algoritmos metaheurísticos, materiales compuestos, plantilla para prótesis.

Design of an Insole with Composite Materials for Foot Prosthesis Using Metaheuristic Algorithms

Abstract. This paper presents the use of a metaheuristic algorithm in the design of an insole for a prosthetic foot made of composite materials, in order to obtain the appropriate design parameters to withstand the specific stresses to which it will be subjected. The Artificial Bee Colony algorithm was used. The problem in this type of design is to find the appropriate thickness of the part, which implies the number of laminates and the respective orientations of the fibers that will

form it, for which there are too many possible combinations when working with composite laminates. The proposed design methodology considers the classical laminate theory, the manufacturing process and the external loads to which the lower limb prosthesis insole will be subjected. The main contribution of this work is the use of metaheuristics to solve design problems with composites posed as optimization problems. The experiments carried out result in a feasible design of this type of structure using carbon fiber/epoxy.

Keywords: Evolutionary optimization, metaheuristic algorithms, composite materials, insole for prostheses.

1. Introducción

Los materiales compuestos o *composites* son conformados por dos o más elementos, fibra y refuerzo. De manera macroscópica, son distintos de las aleaciones. Mientras que en los *composites* las deformaciones sufridas son diferentes en cada uno de sus lados, en los «materiales tradicionales» son iguales en los tres ejes (*x, y, z*). Por ende, el diseño topológico del material compuesto representa un desafío de ingeniería para obtener las especificaciones mecánicas requeridas [1].

En investigaciones previas relacionadas al diseño óptimo de estructuras, describen que en las últimas décadas las industrias como la aeronáutica y la automotriz prefieren utilizar *composites* en lugar de materiales tradicionales, debido a su excelente relación resistencia/peso y alta rigidez específica, lo que es muy atractivo y adecuado para el diseño de prótesis de miembro inferior. Otra ventaja de usar estos materiales, es que la pieza se puede diseñar seleccionando la fibra y las orientaciones adecuadas para cumplir con los requerimientos solicitados.

La flexibilidad en seleccionar estas variables para obtener los requisitos solicitados introduce complejidad en problemas de diseño, de los cuales, en la mayoría se conocen ciertas especificaciones a priori, como el espesor de la lámina, opciones para orientaciones de las mismas y el tipo de material.

Por lo tanto, el diseño de una estructura mediante *composites* se reduce a buscar orientaciones discretas de capas apropiadas y parámetros geométricos en un rango dado, para lograr la resistencia y rigidez solicitadas [2].

Este trabajo propone una metodología a partir de plantear los problemas de diseño con materiales compuestos como problemas de optimización y resolverlos mediante técnicas metaheurísticas, lo que implica el desarrollo y uso de diversos métodos, los cuales han sido una herramienta bastante útil para obtener mejores soluciones y de manera más rápida. La optimización de estructuras hechas con *composites* tiene como tarea encontrar las mejores soluciones de configuración topológica del material para resolver un problema específico.

Uno de los principales objetivos al implementar metaheurísticas en problemas de optimización, es el de resolver situaciones complejas y buscar soluciones factibles [3]. Para este trabajo, se realiza la búsqueda del número de laminados y mejor secuencia de apilamiento en el diseño de una plantilla para prótesis de tobillo-pie hecha de fibra de carbono/epoxi. Primero, se selecciona el material a utilizar; después, se realizan los cálculos teóricos necesarios; posteriormente, se propone un diseño aleatorio que cumpla con un criterio de falla; finalmente, se utiliza el algoritmo de la colonia artificial

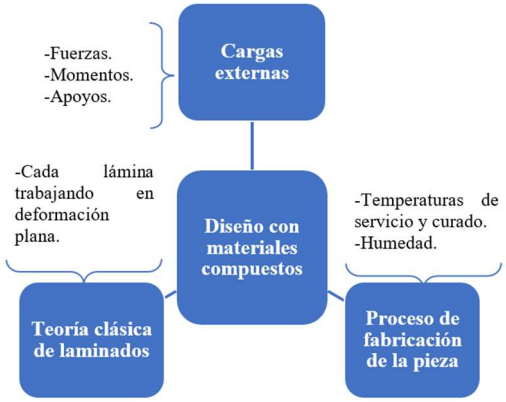


Fig. 1. Elementos a considerar en el diseño con materiales compuestos.

de abejas (ABC, del inglés, *Artificial Bee Colony*) para obtener los parámetros de diseño adecuados, que soporten las cargas y esfuerzos requeridos en su trabajo específico.

2. Antecedentes y trabajos relacionados

En [4] se diseñó un árbol de transmisión para automóviles hecho de *composites*, la metodología de diseño implicó encontrar la combinación adecuada del material para que la pieza no falle en su funcionamiento utilizando un algoritmo de búsqueda propuesto por el autor, los resultados obtenidos fueron satisfactorios.

Mientras que en [5] se utilizaron algoritmos genéticos para el mismo propósito y en [6] además se implementó un algoritmo de inteligencia de enjambre, obteniendo el espesor y la secuencia de apilamiento óptima de *composites* para la pieza, para este caso se diseñó con fibra de vidrio y fibra de carbono, realizándose la comparación respecto al diseño con acero, concluyendo que el diseño con *composites* es viable.

Con respecto a prótesis para tobillo-pie, se ha propuesto en trabajos previos la optimización de la geometría y elasticidad en un pie protésico, específicamente en la plantilla de este, en donde la función de costo a minimizar, está dada por la diferencia entre los pares de la rodilla y el pie al caminar con una prótesis transfemoral.

El diseño óptimo en una prótesis de este tipo depende de la actividad para la cual la requiera el usuario. Otro aspecto a considerar para optimizar la elasticidad del material usado en la prótesis, es analizando la rigidez del mismo, porque estos conceptos están relacionados matemáticamente con sus matrices inversas [7].

En una prótesis para pie desarrollada en Venezuela, se utilizó la técnica de modelado paramétrico de sólidos, con la finalidad de crear un prototipo virtual en 3D para aplicar el método de elementos finitos (MEF) y determinar la capacidad mecánica y funcional de la prótesis, después se optimizó el diseño combinando el método de elementos finitos

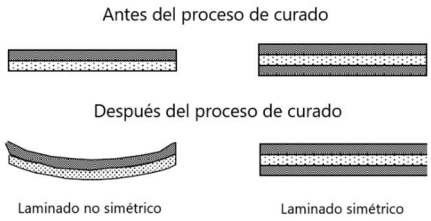


Fig. 2. Laminado simétrico [12].

y la técnica de diseño de experimentos, con el objetivo de verificar la calidad del modelo [8].

En otros trabajos se ha abordado la interacción del pie con el suelo mediante un modelo de contacto esfera-plano, a través de una formulación dinámica multicuerpo [9], y planteando el diseño como un problema de optimización, obteniendo buenos resultados en el estudio del contacto pie-suelo, que pueden servir como referencia en el diseño de plantillas para prótesis y ortesis de miembro inferior [10].

En Biomecánica se emplean metaheurísticas como algoritmos genéticos, redes neuronales, el algoritmo de abejas y la evolución diferencial para la optimización y síntesis de mecanismos en el diseño de prótesis robóticas. Por ejemplo, en [11] se optimizan las longitudes de los eslabones y los ángulos para un mecanismo de cuatro barras, con el que se reproduce el funcionamiento de una rodilla policéntrica.

Respecto a los antecedentes presentados, en general no existen trabajos similares a lo que se está proponiendo, porque todos han aplicado diferentes metodologías de diseño para prótesis de miembro inferior, si bien algunos utilizan optimización mediante metaheurísticas, ninguno ha considerado el diseño topológico del material compuesto para esta aplicación en particular.

Además, en los trabajos que han diseñado con *composites* en otras aplicaciones, las desventajas son que los resultados obtenidos son muy ideales y difíciles de llevar a la práctica, porque obtienen configuraciones topológicas del material muy complejas.

Una de las principales contribuciones de esta investigación es subsanar esta situación, mediante la implementación de restricciones que permitan obtener resultados factibles de materializar y que conlleven a utilizar esta metodología para diversos diseños con *composites* en ingeniería.

3. Metodología

De acuerdo a la literatura, las metaheurísticas pueden adaptarse al diseño con materiales compuestos, porque son métodos de optimización global y se utilizan para problemas no lineales o de variables discretas [2].

Para este caso se analiza el sistema como se observa en la Fig. 1, donde las consideraciones a realizar en el diseño de una pieza hecha con *composites* son, las cargas externas aplicadas como las fuerzas, momentos y apoyos que actúan sobre la plantilla para prótesis de miembro inferior para este caso, la teoría clásica de laminados que fundamenta los cálculos de la mecánica de materiales para *composites* bajo ciertas

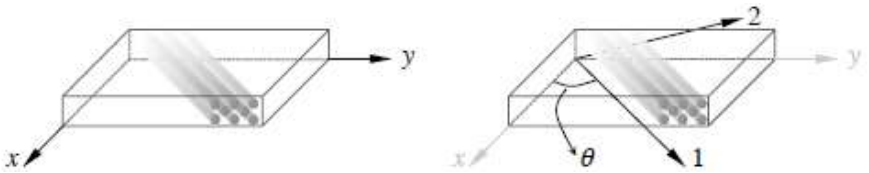


Fig. 3. Definición del sistema de coordenadas global (izquierda), local (derecha), en las ecuaciones anteriores c=cosθ y s=senθ [13].

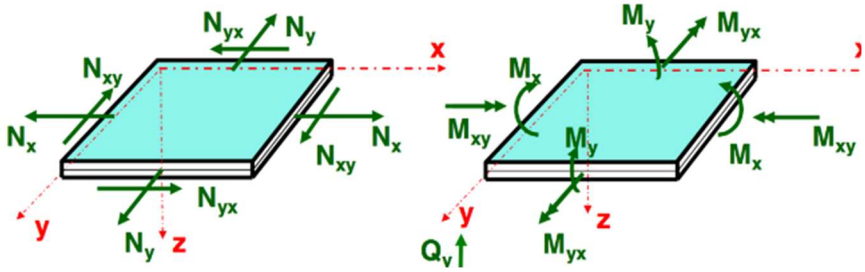


Fig. 4. Direcciones de las cargas y momentos que actúan sobre la lámina [12].

Tabla 1. Parámetros del ABC [16].

Nombre	Símbolo	Descripción
Número de soluciones	*SN*	Número de soluciones (fuentes de alimento)
Número de ciclos	*MCN*	Número total de ciclos (iteraciones) que ejecutará el ABC
Límite	*limit*	Número de ciclos que se conservará una solución sin mejorar antes de ser reemplazada por una nueva generada por una abeja exploradora

características, y el proceso de fabricación en el que se consideran aspectos importantes como las temperaturas de servicio y curado, y el porcentaje de humedad en la pieza.

3.1. Teoría clásica de laminados

Este trabajo considera un laminado simétrico, para que la pieza no se pandee después del proceso de curado, como se muestra en la Fig. 2. Para diseñar tomando en cuenta las cargas aplicadas a la plantilla para prótesis de pie y los esfuerzos internos que se producen en el material, se tiene que aplicar la teoría clásica de laminados.

Para crear una relación lineal entre esfuerzo-deformación para un material anisótropo se parte de la teoría de elasticidad como se muestra en la ecuación (1), mediante la ley de Hooke generalizada [4]:

$$\{\sigma_{ij}\} = [Q_{ij}]\{\varepsilon_{ij}\} \qquad \text{Siendo } i, j = 1, 2, 3, 4, 5, 6, \qquad (1)$$

donde $[Q_{ij}]$ recibe el nombre de la matriz de rigidez. Para un material genérico, esta matriz tiene 36 componentes para definir completamente el material como se muestra en la ecuación (2):

$$[Q_{ij}] = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} & Q_{14} & Q_{15} & Q_{16} \\ Q_{21} & Q_{22} & Q_{23} & Q_{24} & Q_{25} & Q_{26} \\ Q_{31} & Q_{32} & Q_{33} & Q_{34} & Q_{35} & Q_{36} \\ Q_{41} & Q_{42} & Q_{43} & Q_{44} & Q_{45} & Q_{46} \\ Q_{51} & Q_{52} & Q_{53} & Q_{45} & Q_{55} & Q_{56} \\ Q_{61} & Q_{62} & Q_{63} & Q_{46} & Q_{65} & Q_{66} \end{bmatrix}. \quad (2)$$

Para este tipo de diseños los laminados de materiales compuestos son delgados y se considera que la deformación fuera del plano es despreciable [13], entonces se analizan como problemas de deformación plana y se definen como en las ecuaciones (3) y (4):

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = [Q] \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix}, \quad (3)$$

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{21} & Q_{22} & 0 \\ 0 & 0 & Q_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix}. \quad (4)$$

Las ecuaciones para analizar las fibras respecto a una referencia se obtienen tomando en cuenta el esquema mostrado en la Fig. 3. Cada orientación de lámina demanda un sistema de coordenadas local, refiriendo la respuesta de cada lámina al sistema de coordenadas global o viceversa.

Según la teoría clásica de las placas laminadas, la ecuación constitutiva se puede escribir como en se muestra en la ecuación (5). Al considerar un laminado simétrico, resulta que $[B_{ij}]$=0, por lo que se simplifica la ecuación (5) y se obtiene las ecuaciones (6) y (7):

$$\begin{Bmatrix} N \\ M \end{Bmatrix} = \begin{bmatrix} A & B \\ B & D \end{bmatrix} \begin{Bmatrix} \varepsilon^0 \\ k \end{Bmatrix}, \quad (5)$$

$$\begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{16} \\ A_{21} & A_{22} & A_{26} \\ A_{16} & A_{26} & A_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_x^0 \\ \varepsilon_Y^0 \\ \gamma_{xY}^0 \end{Bmatrix}, \quad (6)$$

$$\begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & D_{16} \\ D_{21} & D_{22} & D_{26} \\ D_{16} & D_{26} & D_{66} \end{bmatrix} \begin{Bmatrix} k_x \\ k_y \\ k_{xy} \end{Bmatrix}, \quad (7)$$

donde:

ε^0 = Vector de deformaciones en el plano medio.

k = Curvaturas en la lámina.

Además, hay que calcular los coeficientes A y D respectivamente, y que dependerán del espesor de la pieza hasta la enésima lámina correspondiente. Las fuerzas y los momentos están acoplados por la matriz $[B]$ y están definidos por unidad de longitud del lado sobre el que actúan. Las direcciones de las cargas y momentos aplicados a cada lámina que constituye la pieza se muestran en la Fig. 4.

Para implementar la teoría clásica de laminados, las especificaciones del proceso de fabricación, las cargas externas a las que está sometida una plantilla para prótesis de

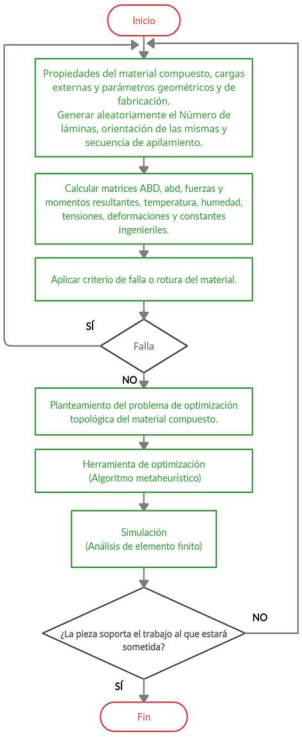


Fig. 5. Metodología propuesta para el diseño topológico con *composites*.

miembro inferior y buscar soluciones factibles en el diseño de la pieza, se propone la metodología presentada en la Fig. 5 mediante su diagrama de flujo, la cual se aplicará para el diseño topológico con *composites* en este trabajo.

La herramienta utilizada para la optimización mediante un algoritmo metaheurístico, fue la colonia artificial de abejas (ABC) perteneciente a los algoritmos de inteligencia colectiva. A continuación, se describen las consideraciones más significativas al respecto.

Se inicia con la selección de un *composite* comercial y se utilizan sus dimensiones y especificaciones mecánicas, además, se considera la geometría y las dimensiones de la pieza; después, se propone aleatoriamente un número de laminados, la orientación de las fibras y la secuencia de apilamiento; posteriormente, se realizan todos los cálculos requeridos según la teoría clásica de laminados y las constantes ingenieriles.

Se aplica un criterio de rotura y se pasa a la siguiente fase cuando la propuesta no falle, de lo contrario, se realizan más propuestas de forma aleatoria; a continuación, se optimiza mediante una metaheurística la propuesta, minimizando la masa de la plantilla según las restricciones y los requerimientos de diseño; finalmente, se valida el diseño topológico obtenido, mediante un análisis de elemento finito para verificar que la pieza soporte el trabajo al que será sometida, si pasa este análisis, el diseño es factible de fabricación. Cabe mencionar que, aunque se realice el análisis por elemento finito para validar el diseño, en la etapa de los cálculos mediante la teoría clásica de laminados ya

Algoritmo 1. Colonia Artificial de Abejas (ABC) adecuado para el diseño con composites.

1. **BEGIN** /*Inicio del algoritmo*/
2. Iniciar la población de fuentes de alimento $\mathbf{x}_{i,0}$, i=1, …, SN
3. Propiedades del material, cargas externas y parámetros geométricos y de fabricación
4. Cantidad de láminas, dirección de las mismas y sucesión de apilamiento
5. Calcular matrices ABD, abd, fuerzas y momentos resultantes, temperatura, humedad, tensiones, deformaciones y constantes ingenieriles
6. Evaluar la calidad de la población
7. Aplicar criterio de falla (TsaiWu)
8. $g=1$
9. **Repeat**
10. Generar soluciones nuevas $\boldsymbol{v}_{i,g}$ para las abejas empleadas mediante (8); aplicar todos los cálculos de la teoría clásica de laminados y criterio de rotura y evaluarlas
11. Mantener la mejor solución entre la actual y la candidata
12. Elegir las soluciones que serán visitadas por una abeja observadora según su calidad
13. Generar soluciones nuevas $\boldsymbol{v}_{i,g}$ mediante (8); aplicar todos los cálculos de la teoría clásica de laminados y criterio de rotura y evaluarlas
14. Mantener la mejor solución entre la actual y la candidata
15. Establecer si existe una fuente abandonada y sustituirla con una abeja exploradora
16. Guardar la mejor solución encontrada hasta el momento
17. $g=g+1$
18. **Until** $g=MCN$
19. Buscar entre todas las soluciones factibles, las que tengan mejor aptitud
20. Imprimir cantidad de láminas, sucesión de apilamiento y masa de la plantilla
21. **END**

están consideradas las deformaciones que sufrirá la pieza en función de las cargas externas que se le aplicarán, por lo que la pieza debe soportar el trabajo sometido sin ningún inconveniente.

3.2. Algoritmo de la colonia artificial de abejas (ABC)

El proceso de búsqueda de néctar en las flores por parte de abejas melíferas ha sido visto como un proceso de optimización. La forma en la que este tipo de insectos sociales logran centrar esfuerzos en zonas con altas cantidades de fuentes de alimento se ha modelado como una metaheurística.

A pesar de que existen diversos modelos basados en abejas [14], para efectos de este trabajo la explicación se basará en el modelo propuesto por Karaboga [15] que resuelve problemas de optimización numérica mediante dos comportamientos: El reclutamiento de abejas en una fuente de alimento y el abandono de una fuente.

El modelo biológico de recolección de alimento en abejas melíferas consta de fuentes de alimento, abejas recolectoras empleadas y recolectoras desempleadas. Una de las ventajas de este algoritmo es el bajo número de parámetros que requiere como puede verse en la Tabla 1. En el ABC, las abejas son vistas como operadores de variación, pues cuando una de ellas llega a una fuente de alimento, calcula una nueva solución candidata $\boldsymbol{v}_{i,g}$ utilizando la ecuación (8).

En donde $\boldsymbol{x}_{i,g}$ representa la solución en la que la abeja se encuentra en ese momento, $\boldsymbol{x}_{k,g}$ es una fuente de alimento aleatoria (y distinta de $\boldsymbol{x}_{i,g}$), g es el número de ciclo actual del programa y ϕ es un número real aleatorio en el intervalo [-1, 1]:

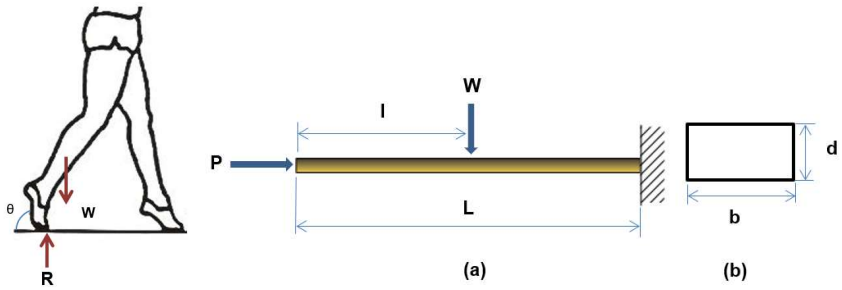


Fig. 6. Pie en fase de apoyo y etapa de despegue con las respectivas fuerzas analizadas. (a) Diagrama equivalente de la planta del pie; (b) sección de perfil para la viga.

Tabla 2. Material propuesto para el diseño de la pieza [17].

Concepto	Símbolo	Unidad	Carbono/epoxi (USN150)
Módulo de elasticidad en dirección 1	E_1	GPa	131.6
Módulo de elasticidad en dirección 2 y 3	E_2, E_3	GPa	8.20
Módulo cortante en plano 23	G_{23}	GPa	3.5
Módulo cortante en plano 12 y 13	G_{13}, G_{12}	GPa	4.5
Coeficiente de Poisson en plano 12 y 13	$\upsilon_{12}, \upsilon_{13}$	---	0.282
Coeficiente de dilatación térmica en dirección 1	α_1	x 10^{-6}/°C	-0.9
Coeficiente de dilatación térmica en dirección 2	α_2	x 10^{-6}/°C	27
Coeficiente de expansión higroscópico en dirección 1	β_1	---	0
Coeficiente de expansión higroscópico en dirección 2	β_2	---	0.4
Resistencia a la tracción en dirección 1	s_1^t	MPa	2000
Resistencia a la compresión en dirección 1	s_1^c	MPa	1400
Resistencia a la tracción en dirección 2 y 3	s_2^t, s_3^t	MPa	61
Resistencia a la compresión en dirección 2 y 3	s_2^c, s_3^c	MPa	130
Resistencia al cortante en plano 2 y 3	s_{23}	MPa	40
Resistencia al cortante en plano 12 y 13	s_{13}, s_{12}	MPa	70
Densidad	ρ	kg/m^3	1550
Espesor de la lámina	$t_{lámina}$	mm	0.25

Tabla 3. Datos de entrada al algoritmo ABC adecuado para el diseño con *composites*.

Descripción	Valor	Parámetro
Deflexión máxima	4 mm	*Cargas externas*
Carga de servicio	3880 N	
Ancho de la plantilla	90 mm	*Geométricos*
Longitud de la plantilla	260 mm	
Temperatura de curado	120 °C	*Fabricación de la pieza*
Temperatura de servicio	20 °C	
Contenido de humedad	0.5 %	
Coeficiente de seguridad	2.5	
Tamaño de población (*SN*)	50 abejas	*Algoritmo ABC*
Número máximo de iteraciones (*MCN*)	1000	
Límite (*limit*)	10	

$$V_{i,g} = X_{i,g} + \emptyset\left(X_{i,g} - X_{k,g}\right). \tag{8}$$

Tabla 4. Resultados del algoritmo ABC adecuado para el diseño de la plantilla.

i	n	$t_{lámina}$ (mm)	d (mm)	Secuencia óptima (θ_k)	m (kg)
1	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
2	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762
3	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762
4	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
5	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
6	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
7	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
8	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762
9	24	0.25	6	[45/-45/0/90/45/-45/0/90/-45/0/90/45]s	0.21762
10	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762
11	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762
12	24	0.25	6	[0/90/45/-45/0/90/45/-45/45/-45/0/90]s	0.21762

A continuación, se detalla el funcionamiento del algoritmo ABC adecuado para la metodología propuesta en la Fig. 5, este mismo puede observarse en el algoritmo 1. Se empieza con las soluciones que representarán las fuentes de alimento iniciales, siendo *SN* el número de soluciones (uno de los parámetros del algoritmo).

Se evalúan estas soluciones y se procede con un ciclo que se repetirá *MCN* veces, dónde *MCN* es el número de ciclos máximo. Dentro de este ciclo se comienza con enviar a las abejas empleadas a las fuentes de alimento y calcular nuevas soluciones candidatas utilizando la ecuación (8), posteriormente se utiliza una selección ambiciosa en la cual se conserva la mejor solución entre la fuente de alimento y su respectiva solución candidata.

Basándose en la aptitud de las fuentes de alimento que se conserven después del paso anterior se determina cuáles soluciones serán visitadas por abejas observadoras. Las abejas observadoras visitarán estas soluciones y generarán soluciones candidatas utilizando la ecuación (8). Posteriormente se realizará una selección ambiciosa entre las soluciones candidatas y la solución respectiva (de manera similar que con las abejas empleadas) [16].

Se decidió utilizar ABC debido a su facilidad de implementación, a las variables discretas características de problemas con laminados de *composites*, y a que en la literatura se han obtenido buenos resultados para este tipo de problemas.

3.3. Planteamiento del problema de optimización

Diseñar una plantilla para prótesis de miembro inferior con materiales compuestos, en donde las variables a optimizar son el número de láminas (espesor de la pieza) y la secuencia de apilamiento, para un laminado simétrico. La función objetivo establecida para este problema es la masa de la plantilla, como se muestra en la ecuación (9). Se propone analizar la planta del pie como una viga en voladizo, si se observa la Fig. 6, el comportamiento del pie en la fase de apoyo y etapa de despegue en el ciclo de marcha, es muy similar a el comportamiento de una viga de este tipo, en donde influyen

principalmente las fuerzas del peso de la persona y la reacción del suelo que son las cargas externas que sufre una plantilla de prótesis para miembro inferior.

En el modelo analizado que se presenta en la figura anterior, *l* es la distancia desde la punta de la viga hasta el punto de aplicación de la fuerza vertical, mientras que *P* y *W* son las proyecciones de la reacción del suelo en las direcciones definidas por la inclinación α del pie respecto a la superficie. También se aprecian *b* y *d*, ancho y espesor de la sección rectangular de la viga según corresponde.

Se precisa que este análisis no da un modelo definitivo, pero si un primer acercamiento al diseño que se pretende obtener, debido a que la plantilla de una prótesis para tobillo-pie no será un prisma rectangular como tal, sino que tendrá una geometría con arco como la de un pie humano, para reproducir de mejor forma la anatomía de este elemento:

$$\begin{gathered} min \qquad f(x) = m = \rho b d L \\ vector\ de\ las\ variables\ de\ diseño \quad x = [n,\ \theta_k, d], \end{gathered} \tag{9}$$

Cotas de diseño:

$$L = 26 \text{ cm},$$
$$\left(\frac{L}{2.4}\right) < b < \left(\frac{L}{2.9}\right).$$

La relación entre ancho y espesor no debe ser mayor a 8.

Sujeto a las restricciones de diseño siguientes:

$n > 10$, *donde* $n \in$ *números enteros*

Ángulos factibles de fabricación:

$$\theta_k = [0°, \pm 45°, 90°]$$

$Capas\ orientadas\ a\ 0° = 25\%$
$Capas\ orientadas\ a\ \pm 45° = 50\%$
$Capas\ orientadas\ a\ 90° = 25\%$

$$4 \text{ mm} \geq \delta_{máx}$$
$$3880 \text{ N} > \sigma$$

Si se ha producido rotura $\rightarrow m = \infty$

Donde:
ρ = densidad del material
n = número de láminas
θ_k = orientación de las fibras
d = espesor de la plantilla
L = longitud de la plantilla
b = ancho de la plantilla
m = masa
$\delta_{máx}$ = deflexión máxima
σ = esfuerzo de servicio

La Tabla 2 muestra las propiedades del material seleccionado para el diseño. El USN150 es un *composite* constituido de fibras de carbono y matriz epóxica (Carbono/Epoxi).

4. Implementación, resultados y discusión

La implementación del método se programó en el entorno de MATLAB® R2015a, los análisis de elemento finito fueron hechos en ANSYS®, y las ejecuciones se llevaron a cabo en un sistema de cómputo con: procesador Intel(R) Core (TM) i7-5600u CPU @ 2.60 GHz, 16 GB de memoria RAM y Sistema operativo Microsoft Windows 10. Los datos de entrada al algoritmo utilizado se presentan en la Tabla 3.

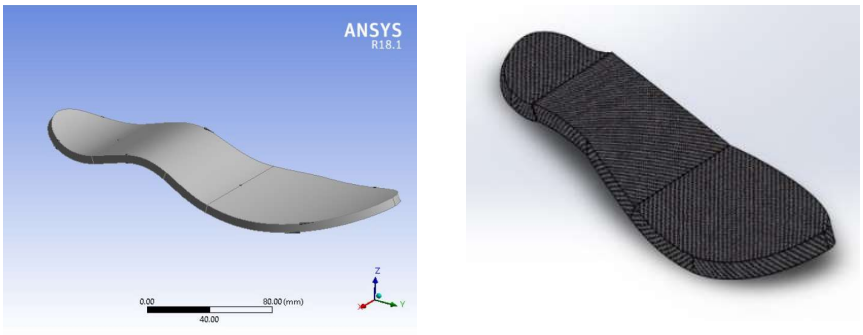


Fig. 7. Plantilla para prótesis de tobillo-pie hecha con *composites* resultante.

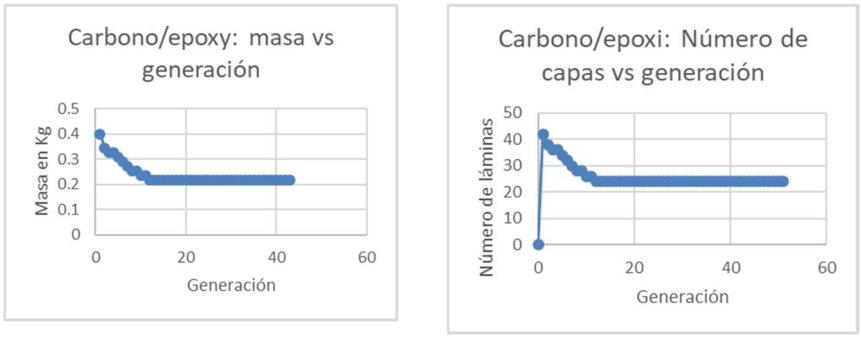


Fig. 8. Variación de la masa de la plantilla y del número de laminados respecto al número de generaciones del algoritmo ABC adecuado para el diseño con composites.

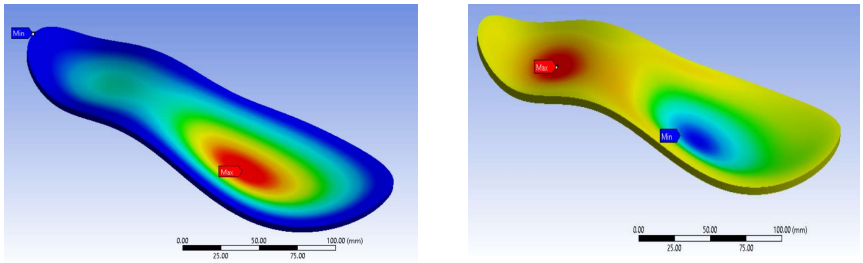


Fig. 9. Deformaciones total y axial de la plantilla obtenida y analizada mediante elemento finito.

Los resultados presentados son valores promedio de haber realizado múltiples corridas del algoritmo y de una selección de muestras de 30 corridas, las cuales convergieron a soluciones factibles.

La Tabla 4 muestra un resumen de los parámetros de diseño obtenidos para una plantilla hecha de fibra de carbono/epoxi USN150, mientras que en la Fig. 7 se aprecia el diseño obtenido y en la Fig. 8 la evolución de la búsqueda, mencionando que a cada

iteración se aumentan o disminuyen dos láminas para realizar una búsqueda exhaustiva, y de las soluciones factibles tomar la de menor valor de función objetivo, dónde n = número de láminas, $t_{lámina}$ = espesor de la lámina, d = espesor total de la pieza, θ_k = orientación simétrica de las fibras (la mitad de la pieza respecto al eje medio), m = masa de la plantilla. En los resultados presentados en la Tabla 4, la nomenclatura de la dirección de las fibras o sucesión óptima de apilamiento es simétrica, por lo que, la s al final del corchete quiere decir que es la primera parte de los laminados en función al eje medio y que la siguiente mitad tendrá la misma configuración de forma simétrica.

También se observa que a través del método ABC se ha encontrado un diseño óptimo que consta de 24 láminas y una masa de la plantilla de 0.21762 kg para todas las ejecuciones que convergen al óptimo global.

Además, las sucesiones de apilamiento varían, esto es entendible porque se trata de un método que genera soluciones aleatorias, el espacio de búsqueda es grande y hay bastantes posibles combinaciones que encaminan a encontrar el objetivo buscado, por lo que se puede considerar que todas las sucesiones de apilamiento obtenidas son factibles, siempre y cuando lleguen al mínimo peso, a la cantidad de láminas óptima y cumplan los criterios de la teoría clásica de laminados respecto al esfuerzo-deformación del material.

En la Fig. 9 se muestran las simulaciones por el método de elemento finito de la plantilla obtenida, en las cuales se da certeza de la funcionalidad del diseño, se presentan los resultados obtenidos de la deformación total y axial, observándose resultados aceptables y funcionales, porque la pieza soportará las cargas críticas a las que se le someterá.

5. Conclusiones y trabajo futuro

Las contribuciones más significativas de esta investigación son, la propuesta de una metodología que es capaz de optimizar a través del uso de una metaheurística, la secuencia de apilamiento y el número de láminas de fibra de carbono/epoxi en el diseño de una plantilla para prótesis de miembro inferior, la cual estará sometida a cargas específicas en su funcionamiento, también, en este trabajo se introdujo el utilizar ángulos de orientaciones de las fibras de los laminados que sean factibles de fabricación en la práctica, para este caso $0°, \pm 45°, 90°$.

Los resultados de las simulaciones por el método de elemento finito, muestran que el diseño obtenido mediante la metodología implementada es funcional y que resistirá de manera adecuada las deformaciones provocadas en la pieza durante su trabajo.

La propuesta realizada puede ser acoplada para otros diseños específicos de plantillas para prótesis de miembro inferior, de esta manera, el diseñador podrá ahorrar tiempo en el proceso cuando requiera de un análisis de este tipo de estructuras, variando dentro del código del método los parámetros necesarios como, la longitud de la plantilla, la carga de servicio, el ancho, la temperatura de curado, la humedad del material, así como las características de otro material que se proponga.

Como siguiente fase de la investigación, se deberá fabricar la plantilla y compararla con las comerciales, para determinar las ventajas y desventajas de la misma y poder mejorar en consecuencia.

Referencias

1. Miravete, A., Cuartero, J.: Materiales compuestos. Tomo 1, Barcelona: Reverté, vol. 2 (2003)
2. Suresh, S., Sujit, P. B., Rao, A. K.: Particle swarm optimization approach for multi-objetive composite box-beam desing. Composite Structures, vol. 81, pp. 598–605 (2007) doi: 10.101 6/j.compstruct.2006.10.008
3. Procópio de Paiva, F. A., Ferreira da Costa, J. A.: Muniz da Silva, C. R.: A serendipity-based approach to enhance particle swarm optimization using scout particles. IEEE Latin America Transactions, vol. 15, pp. 1101–1112 (2017) doi: 10.1109/TLA.2017.7932698
4. Shengyu, W.: Uso de materiales compuestos en el diseño de un árbol de transmisión. España, Universidad Carlos III de Madrid (2014) http://hdl.handle.net/10016/22682
5. Rangaswamy, T., Vijayrangan, S.: Optimal sizing and stacking sequence of composite drive shafts. Materials science, vol. 11, no 2, pp. 133–139 (2005)
6. Manjunath, S., Mohan, K., Channakeshava, R. K.: Optimization of ply stacking sequence of composite drive shaft using particle swarm algorithm. Journal of Engineering Science and Tecnology, vol. 6, no. 3, pp. 323–331 (2011)
7. Strbac, M., Popovic, D. B.: Software tool for the prosthetic foot, modeling and stiffness optimization. Computational and Mathematical Methods in Medicine, vol. 2012 (2012). doi: 10.1155/2012/421796
8. Figueroa, R.: Diseño y análisis mecánico de un pie protésico. Venezuela: Universidad Simón Bolivar (2009)
9. Ojeda-Granja, J., Mayo-Núñez, J.: Influencia de un modelo multicuerpo del pie en la estimación de los parámetros de un modelo de contacto pie-suelo durante la marcha. In: Proceeding XXI Congreso nacional de ingeniería mecánica, España (2016)
10. Páimes R., Villa, J. M., Font Llangunes, U., Lugris Cuadrado, J.: Estimación de los parámetros del modelo de contacto pie-suelo en la marcha humana. In: Proceedings of XIX Congreso Nacional de Ingeniería Mecánica (2012)
11. Lugo-González, E.: Diseño de mecanismos utilizando algoritmos genéticos con aplicación en prótesis para miembro inferior. CDMX, IPN, SEPI, ESIME (2010)
12. Navarro, C.: Elasticidad y resistencia de materiales II. España (2014)
13. Pérez, M. A., Sánchez, M.: Fundamentos de la mecánica de los materiales compuestos. Catalunya, Barcelona, Universidad Politécnica de Catalunya, pp. 19–50 (2014)
14. Baykasoglu, A. Ozbakir L., Tapkan, P.: Artificial bee colony algorithm and its application to generalized assignment problem. In Felix T.S. Chan and Manoj Kumar Tiwari, editors, Swarm Intelligence: Focus on Ant and Particle Swarm Optimization, pp. 113–144 (2007)
15. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization. Journal of Global Optimization, vol. 3, no. 39, pp. 459–471 (2007) doi: 10.10 07/s10898-007-9149-x
16. Silva-Ortigoza, R., Portilla-Flores E. A., Molina-Vilchis, M. A.: Mecatrónica. México (2010)
17. Gil-Lee, D., Pyo-Suh, N.: Axiomatic design and fabrication of composite-structures. Applications in Robots, Machine Tools, and Automobiles (2006)

ISSN 1870-4069

Sistema de teleoperación propioceptiva para la interacción con objetos virtuales

Francesco García Luna, Alma Rodríguez Ramírez,
Osslan Vergara Villegas, Elva Reynoso Jardón,
Manuel Nandayapa

Universidad Autónoma de Ciudad Juárez,
México

{francesco.garcia, alma.rodriguez.ram,
overgara, elva.reynoso, mnandaya}@uacj.mx

Resumen. En el presente trabajo se muestra el desarrollo de un sistema de teleoperación propioceptiva para la interacción con objetos virtuales. Aquí se consideran las relaciones cinemáticas y cinestésicas del teleoperador en el diseño del robot esclavo. El sistema consiste en un sistema de realidad aumentada, 2 marcadores 3D, 2 cámaras infrarrojas,un robot y una computadora para procesar la información. El robot consta de una cámara estéreo montada sobre un sistema de 2 GdL y un robot manipulador de 6 GdL sin pinza. El sistema funciona en 3 etapas: adquisición de señales del teleoperador, control de seguimiento de trayectoria del efector final y estimación de fuerzas sin sensor al contacto. El sistema se comparó contra teclado/mouse, y joystick, en una tarea de teleoperación que consiste en el acercamiento, contacto y alejamiento de dos zonas de contacto, en el menor tiempo posible y con la menor fuerza de contacto.

Palabras clave: Realidad aumentada, propiocepción, teleoperación, control, telecomunicación.

Proprioceptive Teleoperation System for Interaction with Virtual Objects

Abstract. The present work shows the development of a proprioceptive teleoperation system for interaction with virtual objects. In which the teleoperator's kinematics and kinesthetic relations are considered in the design of the slave robot. The system consists of an augmented reality system, 2 3D markers, 2 infrared cameras, a robot and a computer to process the information. The robot consists of a stereo camera mounted on a 2 GdL system and a 6 GdL manipulator robot without a gripper. The system works in 3 stages: Acquisition of teleoperator's signals, end-effector trajectory planning, and sensorless contact force estimation. The system was compared against keyboard / mouse and joystick, in a teleoperation task that consists of approaching, contacting and moving away from two contact zones, in the shortest possible time and with the least contact force.

Keywords: Augmented reality, propioception, teleoperation, control, telecomunication.

1. Introducción

Un sistema tradicional de teleoperación consiste en al menos un sistema maestro, un sistema esclavo y una interfaz entre ellos. Generalmente, una interfaz de teleoperación consta de uno o varios monitores con teclado/mouse (o joystick) o una combinación entre ellos [6], [3], algunas hacen uso de interfaces de realidad aumentada (RA) o virtual (RV) para compensar la experiencia del usuario [20]. Este tipo de sistemas le permiten al teleoperador operar de forma remota un robot y realizar una actividad en algún ambiente, en varias ocasiones, dañino para el ser humano.

En la literatura se encuentran numerosas aplicaciones de teleoperación, por ejemplo, para asistir a astronautas en el acoplamiento a la Estación Espacial Internacional [22], teleoperación cooperativa en una cirugía de invasión mínima [15], teleoperación con retroalimentación háptica [9], teleoperación bilateral [7], trilateral [21] o multilateral [12], teleoperación cooperativa para vehículos aéreos no tripulados (UAV, por sus siglas en inglés) [19], manejo automatizado [8], control en tiempo real de un brazo humanoide utilizando señales mioeléctricas [17], entre otros.

Particularmente, uno de los aspectos más importantes en los sistemas de teleoperación es la correcta identificación del efector final del robot remoto, con respecto a los movimientos del teleoperador y la latencia que existe entre el envío de comandos y su ejecución [10]. Además, los sistemas incluyen limitaciones técnicas en cuánto al campo de visión del sistema de visión y la iluminación utilizados en los sistemas tradicionales. También, el efector final o la región de interés queda ocluida por el mismo robot de forma parcial o completa, aumentando la complejidad de la tarea y generando un impacto negativo en el desempeño del sistema.

En algunas investigaciones se han generado diversas estrategias para compensar la falta de información visual, por ejemplo, en [24] se diseñó un algoritmo para detectar objetos en función de patrones ocluidos, en [4] utilizaron una técnica llamada oclusión ambiental fotométrica para calcular la oclusión ambiental y compensar la falta de iluminación en escenas fijas, por otro lado, [25] utilizó un sistema para detectar la profundidad de una imagen con oclusiones, y [2] reconstruyeron una cara parcialmente ocluida con un sistema neuronal semi-supervisado, y trabajos como el de [23], en donde realizaban un seguimiento de un objeto, a pesar de estar ocluido durante un tiempo en su desplazamiento.

Aún con las soluciones propuestas, no se han logrado estandarizar los modelos para resolver el problema de la oclusión del efector final. Lo anterior es debido a que no se ha tomado en consideración la experiencia del usuario, necesaria en sistemas de teleoperación. Por esta razón, un enfoque no cuantitativo, como la propiocepción, se considerará como una mejora en el desempeño de una tarea de teleoperación. La propiocepción es la capacidad del ser humano que le permite percibir la ubicación de las partes de su cuerpo sin ningún tipo de retroalimentación visual [16].

Se ha utilizado en diferentes aplicaciones, como en el diseño de prótesis [5], de pinzas híbridas [14], el mejoramiento de la percepción háptica [13], para predecir la configuración 3D de un robot suave [18], para compensar la falta de retroalimentación sensorial [11], evasión de obstáculos en robot manipuladores [1], entre otros. Tomando en cuenta lo anterior descrito, en el presente artículo se propone un sistema de teleoperación propioceptiva para la interacción con objetos virtuales, utilizando RA

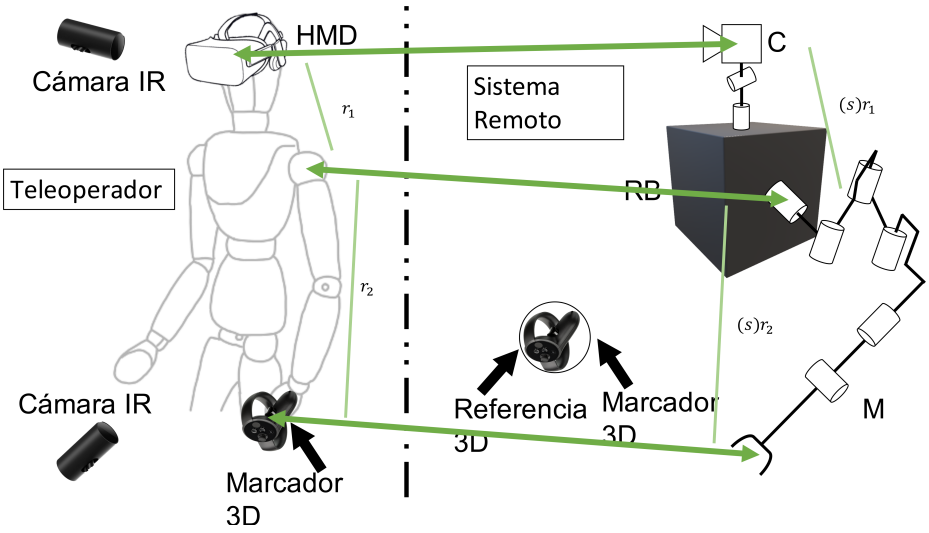


Fig. 1. Esquema del sistema de teleoperación propuesto, en donde se resaltan las relaciones cinemáticas en el diseño del robot remoto utilizando un factor de escala s.

mejorada con propiocepción para asistir a un teleoperador a interactuar con un objeto virtual de forma más eficiente, considerando el tiempo de convergencia y la fuerza de contacto. El sistema toma en consideración las proporciones relativas del ser humano en el diseño del robot remoto y se validó con una serie de experimentos en donde el teleoperador debe realizar la tarea de alcanzar dos puntos de forma precisa y consecutiva, en el menor tiempo posible y con la menor fuerza de impacto posible.

En la sección 2 se muestra el sistema propuesto. En la sección 3 se presentan los diferentes escenarios de teleoperación en donde se identifican los principales métodos de control remoto. Por último, en la sección 4 se presenta un análisis de resultados y en la sección 5 las conclusiones correspondientes.

2. Sistema de teleoperación propioceptiva

El sistema consiste en un sistema de RA, 2 marcadores 3D, 2 cámaras infrarrojas, un robot y una computadora para procesar la información (mostrado en la figura 1). El robot consta de una cámara estéreo montada sobre un sistema de 2 GdL y un robot manipulador de 6 GdL sin pinza. Esta misma cámara es utilizada para complementar, junto con un visor de RV (HMD), el sistema de RA.

Los cámaras infrarrojas se utilizaron, junto con los marcadores 3D que cuentan con una unidad de medición inercial (IMU) de 6 GdL y marcadores infrarrojos para obtener la posición relativa con respecto al HMD. El algoritmo utiliza una versión modificada del motor gráfico Unreal Engine 4.21 que permite la transmisión del video de la cámara estéreo al HMD en el sistema de RA. Además, el motor gráfico permite la comunicación con los Oculus Touch.

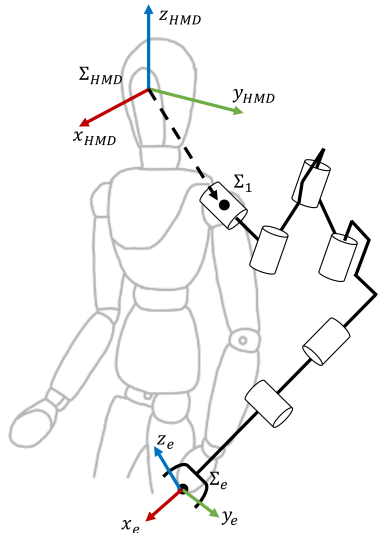


Fig. 2. Relaciones cinemáticas entre cabeza-hombro-mano del teleoperador y cámara-base-efector final del robot remoto.

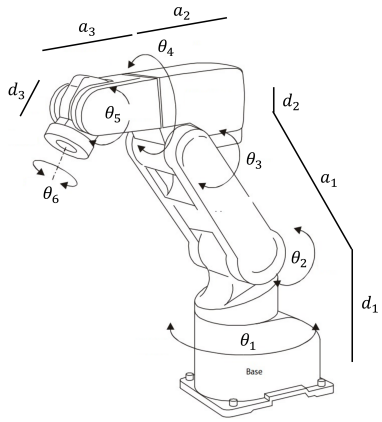


Fig. 3. Representación del robot Mitsubishi Melfa RV2A, en donde se describen los parámetros DH.

Sin embargo, el algoritmo de control de seguimiento de trayectoria se ejecuta de forma externa con Python vía socket y web-socket para el envío de las posiciones articulares al robot.

2.1. Modelo cinemático del robot remoto

El robot remoto es diseñado considerando las relaciones cinemáticas del ser humano cabeza-hombro-mano, de forma que el robot cuenta con una relación cámara-base-efector final (mostrado en la figura 2).

Tabla 1. Parámetros de Denavit-Hartenberg del robot propuesto.

Eslabón i	$R_z(.)$	$T_z(.)$	$T_x(.)$	$R_x(.)$
1	θ_1	d_1	0	0
2	0	0	a_1	$-\pi/2$
3	$\theta_2 - \pi/2$	0	a_2	0
4	θ_3	0	a_3	$-\pi 2$
5	θ_4	d_2	0	$\pi/2$
6	θ_5	0	0	$-\pi/2$
7	θ_6	d_3	0	0

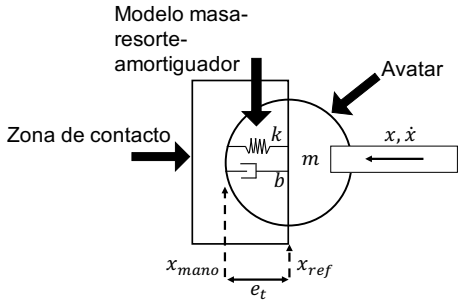


Fig. 4. Representación gráfica de la estimación de la fuerza de contacto utilizando el modelo masa-resorte-amortiguador.

Es importante denotar que, debido a que si bien se mantienen las relaciones cinemáticas cabeza-hombro-mano con cámara-base-efector final, la proporción dimensional cambiará en función de la tarea que se realiza, por ejemplo, una relación 1:1 para un sistema de telepresencia, una relación 1:10 para tareas de construcción o 1:0.01 para cirugías de precisión.

En la figura 3 se observa el manipulador utilizado, donde $\theta_i \in \mathbb{R}$ son las coordenadas generalizadas en radiantes, y $d, a \in \mathbb{R}$ son parámetros de Denavit-Hartenberg, los cuales se muestran en la tabla 1.

2.2. Estimación de las fuerzas de contacto sin sensor

Debido a que utilizar un sensor de fuerza implica una complicación en la configuración, montaje, resolución, costo y fragilidad, las fuerzas de contacto se estiman a partir de la penetración del avatar en la región de interés utilizando un modelo masa-resorte-amortiguador (ver Fig. 4), de modo que:

$$f_x = \begin{cases} ke_x + b\dot{e}_x & \text{si } e_x < 0, \\ 0 & \text{de otra forma,} \end{cases} \tag{1}$$

donde $f_x \in \mathbb{R}$ es la fuerza de contacto estimada a partir de un modelo masa-resorte-amortiguador en el eje x, $k \in \mathbb{R}$ es la constante elástica, $b \in \mathbb{R}$ es la constante de amortiguamiento, y $e_x, \dot{e}_x \in \mathbb{R}$ son el componente x del error de posición y velocidad entre la punta del avatar y la región de contacto respectivamente.

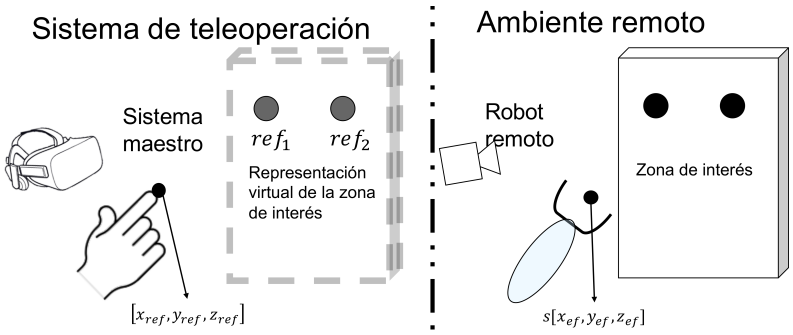


Fig. 5. Escenario experimental propuesto.

Tabla 2. Mapeo de movimientos articulares del avatar utilizando teclado/mouse.

Articulación	Evento
Desplazamiento en x_0	Movimiento vertical del mouse
Desplazamiento en y_0	Movimiento horizontal del mouse
Desplazamiento en z_0	Desplazamiento de la llanta del mouse
Rotación sobre $x_0(\phi)$	Movimiento vertical del mouse + SHIFT
Rotación sobre $y_0(\theta)$	Movimiento horizontal del mouse + SHIFT
Rotación sobre $z_0(\psi)$	Desplazamiento de la llanta del mouse + SHIFT

2.3. Control

Las cámaras infrarrojas utilizan fusión sensorial para identificar la posición de los marcadores 3D tomando en cuenta la información de la IMU y estereografía para obtener de forma precisa la posición y orientación de cada uno de los marcadores con respecto al HMD.

Siendo $\vec{x}_{mano} \in \mathbb{R}^6$ el vector que representa la distancia euclidiana y la orientación entre el HMD y la mano del teleoperador, $\vec{x}_{ref} \in \mathbb{R}^6$ es el vector que representa la posición y orientación del objetivo con respecto al marco coordenado inercial, y $\vec{x}_{HMD} \in \mathbb{R}^6$ es el vector que representa la posición y orientación del HMD con respecto al marco coordenado inercial Σ_0.

Debido a que el brazo del teleoperador y del manipulador pueden tener diferentes proporciones, se consideró un factor de escala $s \in \mathbb{R}$ en el esquema de control, de modo que:

$$\vec{u}_t = K_p\vec{e}_t + K_i \int \vec{e}_t dt + K_d\dot{\vec{e}}_t, \tag{2}$$

$$\vec{e}_t = \hat{\vec{x}}_{mano} - \vec{x}_{ref} = (s)\vec{x}_{mano} - \vec{x}_{ref}, \tag{3}$$

donde $K_p, K_i, K_d \in \mathbb{R}$ son las ganancias proporcional, integral y derivativa correspondientemente.

Tabla 3. Mapeo de movimientos articulares del avatar utilizando joystick.

Articulación	Evento
Desplazamiento en x_0	Movimiento vertical de la palanca
Desplazamiento en y_0	Movimiento horizontal de la palanca
Desplazamiento positivo en z_0	Presión del gatillo del dedo índice
Desplazamiento negativo en z_0	Presión del gatillo del dedo medio
Rotación sobre $x_0(\phi)$	Movimiento vertical de la palanca + botón A
Rotación sobre $y_0(\theta)$	Movimiento horizontal de la palanca + botón A
Rotación positiva sobre $z_0(\psi)$	Presión del gatillo del dedo índice + botón A
Rotación negativa sobre $z_0(\psi)$	Presión del gatillo del dedo medio + botón A

Tabla 4. Mapeo de movimientos articulares del avatar utilizando joystick.

Articulación	Evento
Desplazamiento en x_0	Desplazamiento en x_0 de la mano
Desplazamiento en y_0	Desplazamiento en y_0 de la mano
Desplazamiento en z_0	Desplazamiento en z_0 de la mano
Rotación sobre $x_0(\phi)$	Rotación sobre $x_0(\phi)$ de la mano
Rotación sobre $y_0(\theta)$	Rotación sobre $y_0(\theta)$ de la mano
Rotación sobre $z_0(\psi)$	Rotación sobre $z_0(\psi)$ de la mano

3. Diseño experimental

La tarea de teleoperación consiste en que el usuario debe de alcanzar dos regiones de interés (descrito en la figura 5) en el menor tiempo posible y con la menor fuerza de contacto, utilizando dos de las interfaces de control más utilizadas en sistemas de teleoperación (teclado/mouse y joystick) y se comparó con el sistema propuesto.

3.1. Método 1: teclado/mouse

El primer método consiste en controlar los 6 grados de libertad del avatar/efector-final utilizando únicamente el teclado y el mouse. De forma que es necesario realizar un mapeo de cada uno de los movimientos articulares del avatar a un evento de los dispositivos de entrada, como se observa en la Tabla 2.

3.2. Método 2: Joystick

En el segundo método, se utiliza como joystick únicamente un Oculus Touch sin usar los sensores inerciales o infrarrojos, es decir, únicamente botones y palancas. Para esto, se mapean los eventos del joystick a movimientos articulares del avatar, como se muestra en la Tabla 3.

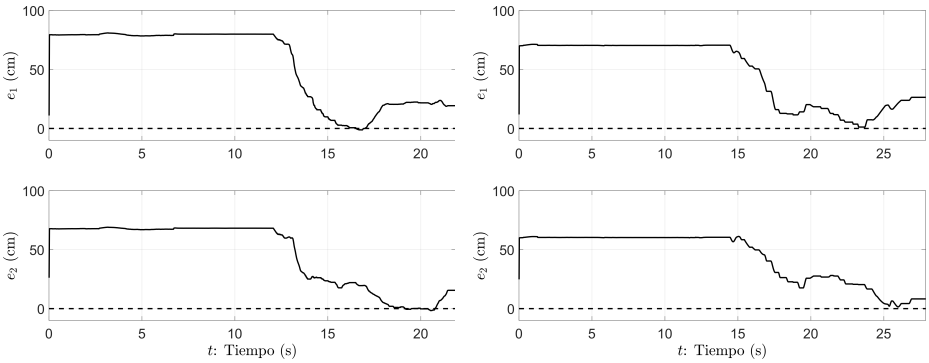


Fig. 6. Error de posición entre el avatar y la referencia 1 (arriba) y la referencia 2 (abajo) utilizando teclado/mouse (6a) y joystick (6b).

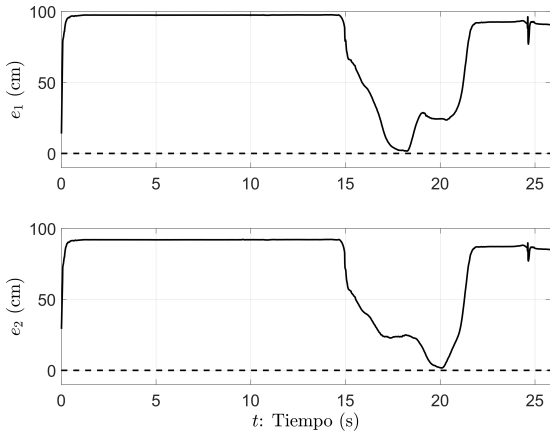


Fig. 7. Error de posición entre el avatar y la referencia 1 (arriba) y la referencia 2 (abajo) utilizando RA + Propiocepción.

3.3. Método 3: RA + propiocepción

Por último, para comprobar el método propuesto, propone un sistema de teleoperación, utilizando RA, propiocepción y considerando las relaciones cinemáticas del ser humano en el control del efector final. De este modo, los movimientos naturales de posición y orientación de la mano con respecto al hombro, son traducidos 1:1 como se muestra en la Tabla 4.

4. Resultados

A continuación se presentan los resultados del desempeño de 3 métodos de teleoperación en la tarea descrita, considerando el error de posición entre la representación de la mano del operador (avatar) y las referencias, las fuerzas de contacto y el error de posición del efector final del robot con las referencias de posición.

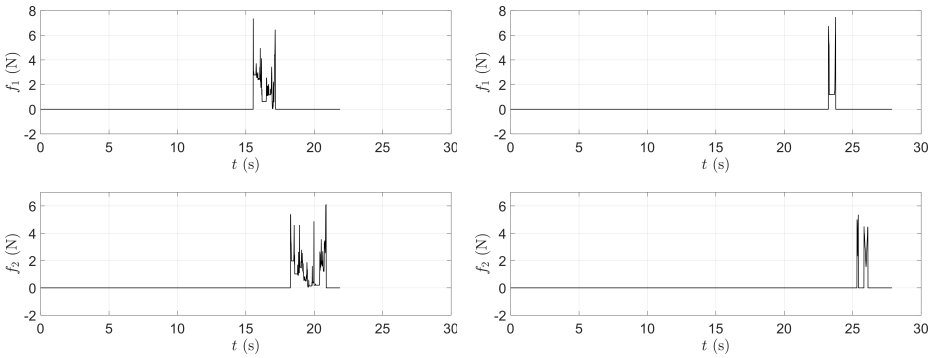


Fig. 8. Fuerzas de contacto estimadas entre el efector final y la referencia 1 (arriba) y la referencia 2 (abajo) utilizando teclado/mouse (8a) y joystick (8b).

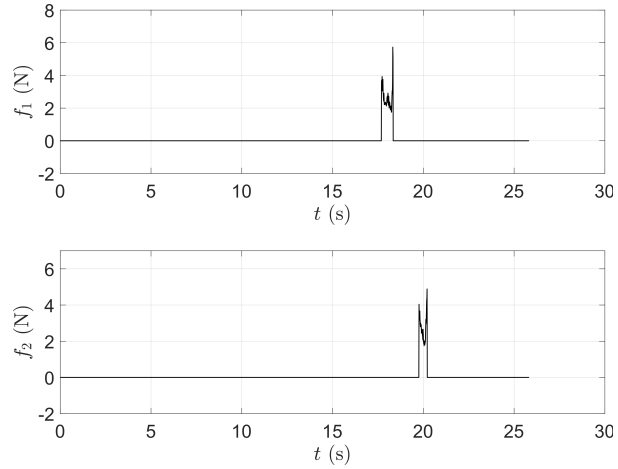


Fig. 9. Fuerza de contacto estimada entre el efector final y la referencia 1 (arriba) y la referencia 2 (abajo) utilizando RA + Propiocepción.

4.1. Tiempo de convergencia

En la Fig. 6a se observa la evolución del error de posición entre el avatar y las referencias en el tiempo utilizando teclado y mouse. El tiempo de convergencia o contacto con la referencia 1 se realizó aproximadamente en 17 segundos, mientras que con la referencia 2, partiendo del contacto con la región 1, sucedió aproximadamente 4 segundos después.

Lo que significa que el tiempo total del experimento de acercamiento y contacto con la referencia 1, distanciamiento o separación de la referencia 1, acercamiento y contacto con la referencia 2 y distanciamiento sucedió en 22 segundos.

Por su parte, utilizando un joystick, el tiempo de contacto con la referencia 1 se realizó después de 23 segundos, realizando un contacto con la referencia 2, 2 segundos después. Esto ocasionó que la duración del experimento completo sucediera en 26 segundos (ver Fig. 6b).

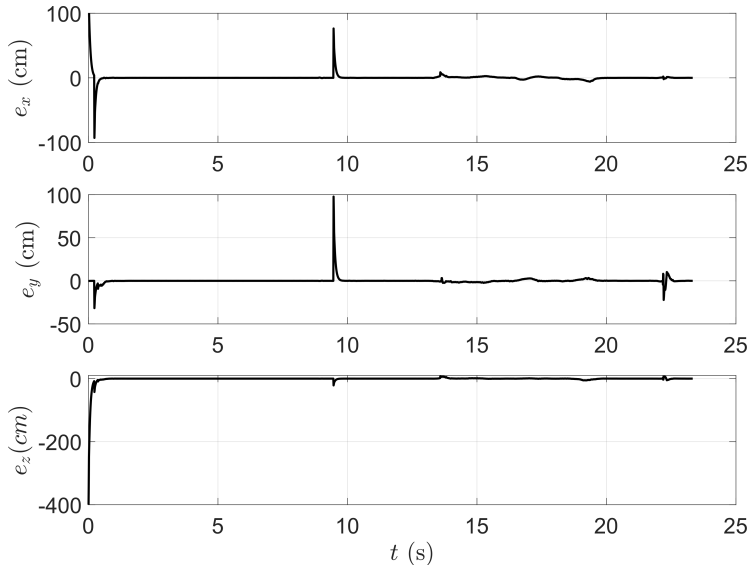


Fig. 10. Error de posición en seguimiento de trayectoria del efector final y la referencia móvil (avatar) utilizando RA + Propiocepción.

De la misma forma, pero utilizando el método propuesto, el tiempo de contacto con la referencia 1 sucedió igual que con el teclado/mouse, a los 17 segundos, con la diferencia de que el contacto con la referencia 2 ocurrió 3 segundos después. El experimento de acercamiento y contacto con la referencia 1, distanciamiento de la referencia 1, acercamiento y contacto con la referencia 2 y distanciamiento fue en 20 segundos (ver Fig. 7).

4.2. Fuerza de contacto

La fuerza de contacto fue modelada como un sistema masa-resorte-amortiguador. Los coeficientes dependen del material con el que se este en contacto. Para fines de simulación se establecieron de forma arbitraria como $k = 1$ y $b = 0{,}01$.

Utilizando teclado/mouse, la fuerza de contacto máxima que el efector final ejerció sobre la referencia 1 es de $7{,}3364N$, mientras que con la referencia 2 es de $6{,}1033N$ (ver Fig. 8a). A diferencia del uso del Joystick, las cuales son de $7{,}4573N$ y $5{,}3380N$ respectivamente (ver Fig. 8b). Por su parte, utilizando RA + Propiocepción, la fuerza de contacto con la referencia 1 alcanza un máximo de $5{,}7318N$ y con la referencia 2 de $4{,}8866N$ (ver Fig. 9).

4.3. Teleoperación

Como robot esclavo se utiliza un Mitsubishi Melfa RV2A, de 6 grados de libertad, y un control de seguimiento de trayectoria de posición y orientación para ocasionar que el efector final siga la referencia dinámica del avatar y alcance dos puntos en su espacio de trabajo.

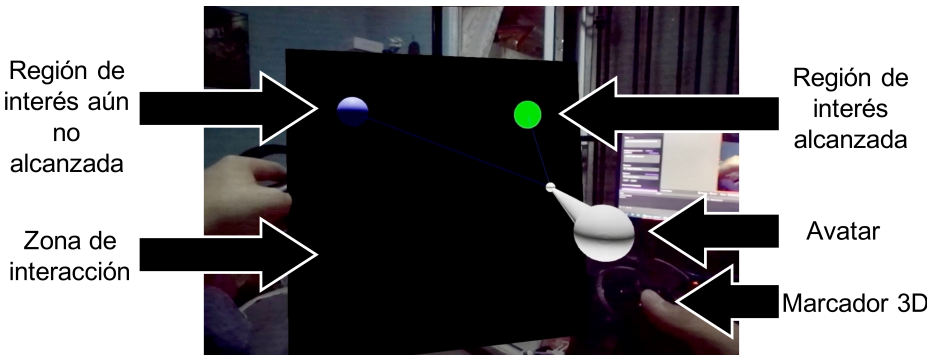


Fig. 11. Interfaz de RA y sus componentes.

Tabla 5. Tabla de desempeño.

Método	t_1 **(s)**	t_2 **(s)**	f_1 **(N)**	f_2 **(N)**
Teclado/Mouse	**17**	21	7.3364	6.1033
Joystick	23	25	7.4573	5.3380
RA + Propiocepción	**17**	**20**	**5.7318**	**4.8866**

En las figuras 10a, 10b y 11 se observa el error de seguimiento entre el efector final y el avatar utilizando teclado/mouse, joystick y la metodología propuesta respectivamente utilizando las ecuaciones (2) y (3). En los tres escenarios, el robot sigue correctamente la trayectoria descrita por el avatar, a pesar de que existe una diferencia en las proporciones cinemáticas entre ambos.

Se considera únicamente la posición del avatar con una orientación constante perpendicular a la zona de interacción. Finalmente, en la Fig. 11 se muestra una imagen de la interfaz de RA en donde se muestran los componentes en interacción.

5. Conclusiones

Considerando que la meta de la tarea era la de realizar contacto con dos áreas en el menor tiempo con la menor fuerza de contacto, existe una clara mejora en el desempeño si se utiliza RA + propiocepción como dispositivo de entrada en un sistema de teleoperación, logrando una mejora del 30 % con respecto al uso de joystick y un 10 % con respecto al teclado/mouse en cuanto al tiempo de convergencia.

Por otro lado, si se consideran las fuerzas de contacto, existe una mejora del 30 % en la referencia 1 y 9 % en la referencia 2 con respecto a utilizar un joystick, y una mejora del 28 % en la referencia 1 y 24 % en la referencia 2 con respecto a utilizar teclado/mouse (ver Tabla 5). Si se compara la Fig. 6a y 7 se observa una disminución en micro-movimientos, lo que significa un menor consumo energético.

Aunque la diferencia entre el uso de teclado/mouse y la metodología propuesta pareciera no ser considerable, es importante destacar que es más fácil controlar posición y orientación del efector final si se utiliza la metodología propuesta, debido a que son los movimientos naturales de la misma mano.

De igual forma, el uso de interfaces de teleoperación que no consideran las relaciones cinemáticas humanas en su diseño, supone un tiempo de adaptación y entrenamiento superior, sin mencionar que el grado de atención es mayor.

Referencias

1. Baradaran-Birjandi, S. A., Kuhn, J., Haddadin, S.: Observer-extended direct method for collision monitoring in robot manipulators using proprioception and IMU sensing. IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 954–961 (2020) doi: 10.1109/LRA.2020.2967287
2. Cai, J., Han, H., Cui, J., Chen, J., Liu, L., Zhou, S. K.: Semi-supervised natural face de-occlusion. IEEE Transactions on Information Forensics and Security, vol. 16, pp. 1044–1057 (2021) doi: 10.1109/TIFS.2020.3023793
3. Cizmeci, B., Xu, X., Chaudhari, R., Bachhuber, C., Alt, N., Steinbach, E.: A multiplexing scheme for multimodal teleoperation. ACM Transactions on Multimedia Computing, Communications, and Applications, vol. 13, no. 2, pp. 1–28 (2017) doi: 10.1145/3063594
4. Hauagge, D., Wehrwein, S., Bala, K., Snavely, N.: Photometric ambient occlusion for intrinsic image decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 4, pp. 639–651 (2016) doi: 10.1109/TPAMI.2015.2453959
5. Koukoulas, N., Bertos, G. A., Mablekos-Alexiou, A., Papadopoulos, E.: A biomechatronic EPP upper-limb prosthesis teleoperation system implementation using bluetooth low energy. IEEE Transactions on Medical Robotics and Bionics, vol. 2, no. 2, pp. 282–291 (2020) doi: 10.1109/EMBC.2018.8512634
6. Li, C., Jiang, Z., Li, Z., Fan, C., Liu, H.: A novel semi-autonomous teleoperation method for the tiangong-2 manipulator system. IEEE Access, vol. 7 (2019) doi: 10.1109/ACCESS.2019.2952762
7. Martinez, C. A. L., Polat, I., Molengraft, R. V. D., Steinbuch, M.: Robust high performance bilateral teleoperation under bounded time-varying dynamics. IEEE Transactions on Control Systems Technology, vol. 23, no. 1, pp. 206–218 (2015) doi: 10.1109/TCST.2014.2321522
8. Neumeier, S., Wintersberger, P., Frison, A. K., Becher, A., Facchi, C., Riener, A.: Teleoperation: The holy grail to solve problems of automated driving? sure, but latency matters. In: Proceedings of the 11th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, pp. 186–197 (2019) doi: 10.1145/3342197.3344534
9. Quan-Zen, A., Horan, B., Nahavandi, S.: Multipoint haptic mediator interface for robotic teleoperation. IEEE Systems Journal, vol. 9, no. 1, pp. 86–97 (2015) doi: 10.1109/JSYST.2013.2283955
10. Rakita, D., Mutlu, B., Gleicher, M.: Effects of onset latency and robot speed delays on mimicry-control teleoperation. In: Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction, pp. 519–527 (2020) doi: 10.1145/3319502.3374838
11. Rossi, M., Bianchi, M., Battaglia, E., Catalano, M. G., Bicchi, A.: HapPro: A wearable haptic device for proprioceptive feedback. IEEE Transactions on Biomedical Engineering, vol. 66, no. 1, pp. 138–149 (2019) doi: 10.1109/TBME.2018.2836672
12. Shahbazi, M., Atashzar, S. F., Patel, R. V.: A systematic review of multilateral teleoperation systems. IEEE Transactions on Haptics, vol. 11, no. 3, pp. 338–356 (2018) doi: 10.1109/TOH.2018.2818134
13. Sornkarn, N., Nanayakkara, T.: Can a soft robotic probe use stiffness control like a human finger to improve efficacy of haptic perception? IEEE Transactions on Haptics, vol. 10, no. 2, pp. 183–195 (2017) doi: 10.1109/TOH.2016.2615924

14. Su, Y., Fang, Z., Zhu, W., Sun, X., Zhu, Y., Wang, H., Tang, K., Huang, H., Liu, S., Wang, Z.: A high-payload proprioceptive hybrid robotic gripper with soft origamic actuators. IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 3003–3010 (2020) doi: 10.1109/LRA.2020.2974438
15. Takhmar, A., Polushin, I. G., Talasaz, A., Patel, R. V.: Cooperative teleoperation with projection-based force reflection for MIS. IEEE Transactions on Control Systems Technology, vol. 23, no. 4, pp. 1411–1426 (2015) doi: 10.1109/TCST.2014.2369344
16. Taylor, J.: Proprioception. In: Squire, L. R. (ed) Encyclopedia of Neuroscience, pp. 1143–1149 (2009)
17. Tortora, S., Moro, M., Menegatti, E.: Dual-myo real-time control of a humanoid arm for teleoperation. In: Proceedings of the 14th ACM/IEEE International Conference on Human-Robot Interaction, pp. 624–625 (2019) doi: 10.1109/hri.2019.8673259
18. Truby, R. L., Santina, C. D., Rus, D.: Distributed proprioception of 3d configuration in soft, sensorized robots via deep learning. IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 3299–3306 (2020) doi: 10.1109/LRA.2020.2976320
19. Vitor, R., Keller, B., D'Angelo, T., Azpurua, H., Bianchi, A. G. C., Delabrida, S.: Collaborative teleoperation evaluation for drones. In: Proceedings of the 18th Brazilian Symposium on Human Factors in Computing Systems, Association for Computing Machinery (2019) doi: 10.1145/3357155.3358439
20. Walker, M. E., Hedayati, H., Szafir, D.: Robot teleoperation with augmented reality virtual surrogates. In: Proceedings of the 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (2019) doi: 10.1109/hri.2019.8673306
21. Weihua, L., Haibo, G., Liang, D., Mahdi, T.: Trilateral predictor-mediated teleoperation of a wheeled mobile robot with slippage. IEEE Robotics and Automation Letters, vol. 1, no. 2, pp. 738–745 (2016) doi: 10.1109/LRA.2016.2522503
22. Wilde, M., Chua, Z. K., Fleischner, A.: Effects of multivantage point systems on the teleoperation of spacecraft docking. IEEE Transactions on Human-Machine Systems, vol. 44, no. 2, pp. 200–210 (2014) doi: 10.1109/THMS.2013.2295298
23. Yang, Y., Sundaramoorthi, G.: Shape tracking with occlusions via coarse-to-fine region-based sobolev descent. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 5, pp. 1053–1066 (2015) doi: 10.1109/TPAMI.2014.2360380
24. Zhou, C., Yuan, J.: Occlusion pattern discovery for object detection and occlusion reasoning. IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 7, pp. 2067–2080 (2020) doi: 10.1109/TCSVT.2019.2909982
25. Zhu, H., Wang, Q., Yu, J.: Occlusion-model guided antiocclusion depth estimation in light field. IEEE Journal of Selected Topics in Signal Processing, vol. 11, no. 7, pp. 965–978 (2017) doi: 10.1109/JSTSP.2017.2730818

ISSN 1870-4069

Clasificación de complicaciones en diabetes mellitus mediante algoritmos genéticos

Mario Daniel Cervantes-Guerrero[1], Miguel Cruz [2],
Adan Valladares-Salgado[2], Jorge Issac Galván-Tejada[1],
Tania A. Gutiérrez-García[3], Carlos Eric Galván-Tejada[1]

[1] Universidad Autónoma de Zacatecas,
México

[2] Unidad de Investigación Médica en Bioquímica,
Centro Médico Nacional Siglo XXI,
México

[3] Universidad de Guadalajara,
Centro Universitario de Ciencias Exactas e Ingenierías,
México

{danielcervantesguerrero, gatejo, ericgalvan}@uaz.edu.mx,
{adan.valladares, miguel.cruzlo}@imss.gob.mx
tania.gutierrez@academicos.udg.mx

Resumen. La diabetes tipo 2 es una enfermedad metabólica que se caracteriza por la incapacidad de regular los niveles de glucosa en la sangre debido a la resistencia a la insulina y una limitada producción de la misma en el cuerpo, causando graves complicaciones a largo plazo, las cuales incluyen retinopatía, neuropatía y nefropatía. En este trabajo se presenta una clasificación de complicaciones de pacientes con diabetes tipo 2 realizada a partir de una base de datos con información clínica del Hospital de Especialidades del Centro Médico Nacional Siglo XXI, México. Se identificaron las características más relevantes mediante algoritmos genéticos (GA), para luego usarlos como información de entrada para la estimación de un modelo de clasificación basado en SVM (Support Vector Machine). Los resultados muestran la correlación entre las características estudiadas y las complicaciones de los pacientes, con un área bajo la curva ROC (Receiver Operating Characteristic) de 0.69. Así, se propone el uso de ésta combinación de estimadores para el futuro desarrollo de herramientas predictivas y de diagnóstico para la enfermedad.

Palabras clave: Máquina de soporte de vectores, algoritmos genéticos, diabetes tipo 2.

Classification of Complications in Diabetes Mellitus Using Genetic Algorithms

Abstract. Type 2 diabetes is a metabolic disease characterized by the inability to regulate blood glucose levels due to insulin resistance and limited insulin production in the body, causing serious long-term complications, which they include retinopathy, neuropathy, and nephropathy. This paper presents a

classification of complications of patients with type 2 diabetes based on a database with clinical information from the Hospital de Especialidades del Centro Médico Nacional Siglo XXI, Mexico. The most relevant features were identified using genetic algorithms (GA), to later use them as input information for the estimation of a classification model based on SVM (Support Vector Machine). The results show the correlation between the characteristics studied and the complications of the patients, with an area under the ROC curve (Receiver Operating Characteristic) of 0.69. Thus, the use of this combination of estimators is proposed for the future development of predictive and diagnostic tools for the disease.

Keywords: Support vector machine, genetic algorithms, diabetes type 2.

1. Introducción

De acuerdo a la Organización Mundial de la Salud (WHO), aproximadamente más de 422 millones de personas padecían diabetes en 2014; se estima que para el 2030, esta enfermedad será la séptima causa de mortalidad a nivel mundial [15]. La diabetes es una enfermedad que se desarrolla cuando el páncreas no sintetiza o secreta la insulina que el cuerpo humano necesita o cuando la insulina no se usa de manera eficiente [13], lo que provoca un aumento en los valores normales de glucosa en la sangre.

Como consecuencia, este cambio de niveles produce un aumento excesivo en los niveles de glucosa en la sangre, esto se relaciona con complicaciones a largo plazo tales como la disfunción e insuficiencia de órganos, especialmente ojos, riñones, nervios, corazón y vasos sanguíneos [1].

La diabetes tipo 2 es una afección común y grave relacionada con la reducción de la esperanza de vida, representa a más del 90 % de los pacientes con diabetes y causa diferentes complicaciones microvasculares y macrovasculares. A pesar del conocimiento cada vez mayor sobre los factores de riesgo de la diabetes tipo 2 y la evidencia de programas de prevención exitosos, la incidencia y prevalencia de la enfermedad continúa en aumento a nivel mundial.

La detección temprana mediante programas de detección y la disponibilidad de terapias seguras y eficaces reducen la morbilidad y la mortalidad al prevenir o retrasar las complicaciones asociadas a la enfermedad [3]. En otros trabajos se ha identificado que algunas características como el tipo de insulina suministrada al paciente y el tiempo transcurrido desde que el paciente contrajo diabetes, ayudan a identificar el riesgo de retinopatía diabética [21].

En otros estudios, utilizando parámetros no invasivos que no se basan en el laboratorio, pudo predecir con éxito el desarrollo de diabetes e hipertensión en pacientes [5]. En trabajos similares, se muestra cómo la extracción de datos y los métodos computacionales se pueden adoptar de manera efectiva en la medicina clínica para calcular el factor de riesgo de un paciente con diabetes tipo 2 para desarrollar una complicación microvascular [4]. Se han probado diferentes clasificadores para medir su desempeño en la clasificación de pacientes con diabetes mellitus [8, 21, 5].

Fig. 1. Diagrama de la metodología.

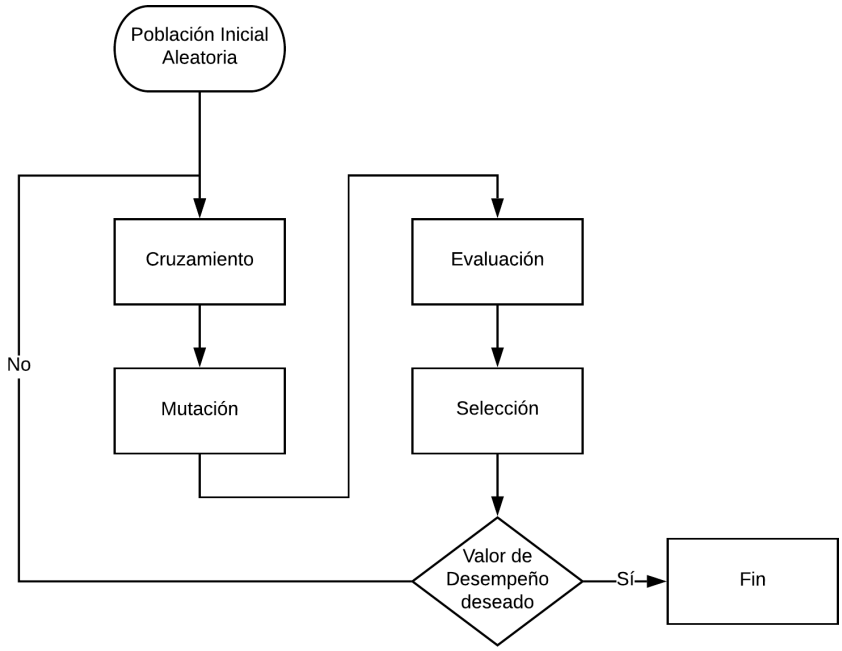


Fig. 2. Diagrama general de un algoritmo genético.

El objetivo de este trabajo es diseñar un modelo predictivo para pacientes diabéticos y anticipar la aparición de complicaciones de la enfermedad, asi como detectar parámetros que proporcionen información relevante a un modelo de clasificación usando algoritmos genéticos como enfoque para la selección de características de un conjunto de datos clínicos. El modelo es útil para notificar el riesgo a los pacientes de forma temprana y evitar complicaciones que aumentan el riesgo de muerte.

2. Materiales y métodos

La metodología seguida se presenta en la Figura 1. Se inicia con la base de datos pura (sin procesar), que es la información sin ninguna modificación y que contiene todas las variables y observaciones capturadas en el Hospital de Especialidades del Centro Médico Nacional Siglo XXI, del Instituto Mexicano del Seguro Social (IMSS). Luego, el pre-procesamiento, consiste en la selección de muestras y observaciones del conjunto de datos original, la normalización de datos, la eliminación de casos incompletos y la eliminación de características sin suficientes observaciones.

Tabla 1. Parámetros eliminados.

Parámetro	Número de observaciones	Relación respecto al total
ACARBO.MG.DIA	10	1.12
ROSIGLI.MG.DIA	12	1.34
PIOGLI.MG.DIA	16	1.79
INSUL.MG.DIA	235	26.34
GLIBEN.MG.DIA	385	43.16
METFOR.MG.DIA	701	78.58

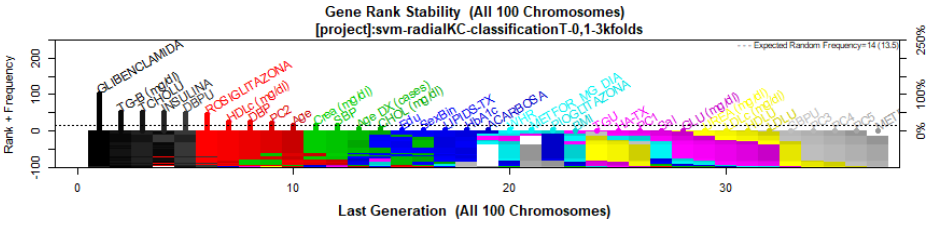


Fig. 3. Resultados de salida de la selección de parámetros por frecuencia de aparición.

A continuación se lleva a cabo la selección de parámetros, se realiza con el fin de extraer las variables más relevantes, para posteriormente diseñar un modelo de clasificación mediante un enfoque de máquina de soporte de vectores (SVM). Finalmente, se realiza un paso de validación del modelo para evaluar los resultados obtenidos midiendo la curva característica operativa del receptor (ROC) y su área bajo la curva (AUC) correspondiente.

2.1. Base de datos

Los datos utilizados en este trabajo incluyen 48 variables y 1,787 observaciones. Cada variable o característica proporciona información sobre los pacientes, tales como su edad, sexo, tipo de complicación, etc. La edad media de los pacientes es de 52.77 años, con una desviación estándar de 10.12, 892 varones y 895 mujeres.

El conjunto de datos contiene 1.787 observaciones, de las cuales 889 son controles, es decir, pacientes sin diabetes y 898 son casos, es decir, pacientes con presencia de diabetes de tipo 2. Considerando que el objetivo de este trabajo es clasificar correctamente a los pacientes que desarrollaron una complicación, los casos control no se tomaron en cuenta para este estudio.

2.2. Preprocesamiento de datos

Reducir la dimensionalidad del estudio es recomendable en bases de datos con una alta cantidad de variables, debido a que existen variables que no aportan información [14] o a que no contienen suficientes observaciones para realizar un análisis, por eso, cuando un parámetro contiene menos del 80 % del total de observaciones, se descarta del estudio [6].

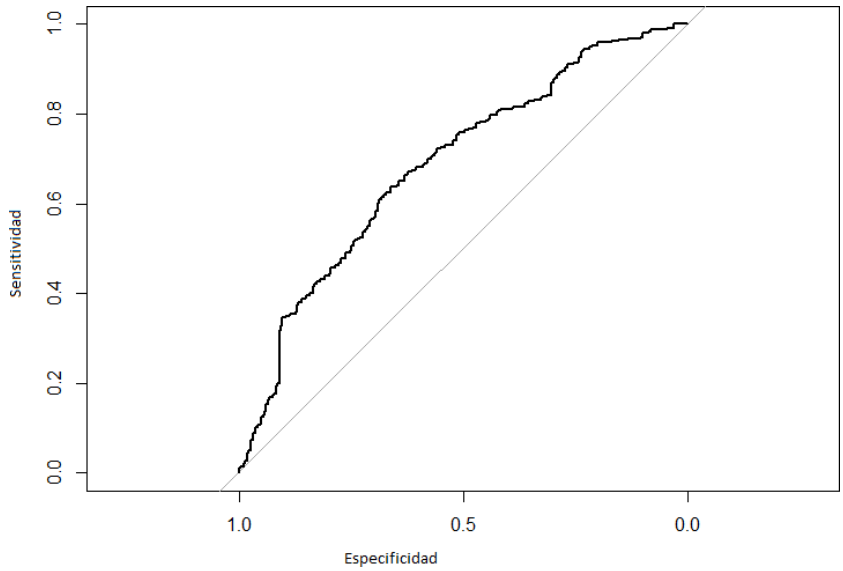


Fig. 4. Curva ROC del modelo de clasificación.

Existen variables con suficientes observaciones para su análisis, pero con valores ausentes, un método sencillo para enfrentar este problema es la imputación por promedio aritmético de la variable [24], por lo tanto se realiza la selección de parámetros y observaciones, así como la eliminación de casos incompletos y carecterísticas sin suficientes observaciones.

Finalmente, los datos totales se dividen en dos conjuntos, uno conformado por el 70 % de los datos, el cual es designado para el entrenamiento del modelo y el 30 % restante de los datos, que se utilizará para probar su eficiencia de clasificación [22, 18].

2.3. Selección de características

La selección de características se realiza con algoritmos genéticos [20]. Los GA son algoritmos inspirados en la teoría de la evolución. Los GA buscan una posible solución a un problema específico en una estructura de datos similar a un cromosoma, y luego estas estructuras se recombinan para preservar la información más relevante [23]. El diagrama general de un algoritmo genético se muestra en la figura 2 [19].

El procedimiento parte de una población aleatoria de subconjuntos variables de un tamaño determinado (definidos como cromosomas). Cada cromosoma se evalúa por su capacidad para predecir una variable dependiente, en este caso la presencia o ausencia de complicaciones.

El principio general es reemplazar la población inicial con una nueva población que incluye variantes de cromosomas con mayor precisión de clasificación y repetir el proceso suficientes veces para lograr el nivel deseado de precisión [20]. El objetivo de este proceso es encontrar aquellas características que describan mejor las observaciones en el conjunto de datos en relación con la salida.

Tabla 2. Parámetros de evaluación del modelo.

Parámetro de evaluación	Value
Exactitud	0.6528
Sensitividad	0.6325
Especificidad	0.6696
PPV	0.6130
NPV	0.6877
Exactitud balanceada	0.6510

Una vez que se han determinado las características más significativas, estas se pueden utilizar como entrada para el paso de clasificación. R [17] es un lenguaje de programación de licencia abierta y un entorno de software libre para análisis estadístico respaldado por R Foundation. El lenguaje R se usa universalmente entre los científicos de datos para desarrollar modelos estadísticos.

Este lenguaje fue seleccionado por su accesibilidad y su desempeño para el modelado estadístico. Para el análisis de algoritmos genéticos se utilizó GALGO [20], un paquete de software R para entrenar algotitmos genéticos y así seleccionar subconjuntos de variables.

2.4. Modelo de clasificación

Se utilizan las características más relevantes como entrada para un modelo de clasificación de máquina de soporte de vectores (SVM).

Un SVM es un algoritmo de aprendizaje automático supervisado en el cual se entrena un modelo de aprendizaje mediante un ejemplo de datos, los datos deben clasificarse como una de muchas categorías y el algoritmo SVM construye un modelo que predice la categoría de un nuevo ejemplo dado, un SVM construye un hiperplano óptimo para clasificar patrones que maximizan la distancia desde el hiperplano al punto más cercano de cada patrón.

Su objetivo principal es maximizar el margen para que pueda clasificar correctamente los patrones dados; cuanto mayor sea el tamaño del margen, mejor será el rendimiento [16].

2.5. Validación del modelo

La validación de modelos matemáticos y computacionales es un paso importante porque sin este procedimiento, las razones para creer que el modelo funciona de manera óptima son inexistentes [22].

Para medir la capacidad de clasificación del modelo de este estudio, se calculan las métricas estadísicas que normalmente son útiles para tal fin, como son especificidad, sensibilidad, curva ROC, AUC, valor predictivo positivo (VPP), valor predicho negativo (VPN) y precisión equilibrada. La sensibilidad está ligada a la verdadera razón positiva, que en este caso sería la capacidad del modelo para clasificar correctamente a los pacientes con presencia de complicaciones.

Tabla 3. Métodos de clasificación.

Método de clasificación	**Desempeño**
Máquina de soporte de vectores	0.694
Random forest	0.607
Regresión lineal	0.6308
Regresión logística	0.6384

Por otro lado, la especificidad está ligada a la razón negativa, que se referiría a la capacidad del modelo para clasificar correctamente a los pacientes sin complicaciones. La especificidad está ligada a la capacidad de los modelos para clasificar correctamente a los pacientes sin complicaciones.

Los valores graficados de la sensibilidad y la especificidad se denominan curva ROC, y se ha utilizado ampliamente para medir o visualizar el rendimiento de un clasificador junto con su valor de área bajo la curva (AUC), con el fin de seleccionar un punto de operación adecuado, llamado como umbral de decisión [10]. El análisis ROC proporciona dos resultados principales: la precisión de la clasificación de complicaciones de la prueba y el valor de punto de corte óptimo para la prueba.

El valor VPP es la probabilidad de que un paciente con clasificación positiva en el modelo realmente presente una complicación mientras que el valor de VPN es la probabilidad de que un paciente con una prueba de clasificación negativa realmente no tenga una complicación.

La precisión equilibrada se reduce al rendimiento de predicción promedio entre VPP y VPN. Una validación hold-out es un tipo de validación cruzada. Esta separa las observaciones en dos conjuntos, uno utilizado para entrenar al modelo y otro conjunto de datos diferentes para realizar su prueba de validación [2]. Para calcular estas métricas, se utilizó el paquete Caret [9] para el software estadístico R [17].

3. Resultados

Las caraterísticas utilizadas para este estudio se muestran en el Anexo 1. De estas, las que se eliminaron durante el Pre-procesamiento por presentar una ausencia de observaciones de más del 80 %, están en la Tabla 1. Tampoco se muestran las variables que se refieren al identificador del paciente o su estatus como caso o control.

Como se mencionó anteriormente, se utilizó GALGO para implementar el algoritmo genético, los parámetros de este se configuraron para que tome de forma aleatoria una muestra entre todo el conjunto de datos, el método de clasificación evaluador fue por máquinas de soporte de vectores.

Para reducir la dimensionalidad del modelo se eligió un tamaño de cromosoma de 5 genes, probando la eficiencia de clasificación del modelo usando conjuntos de cinco parámetros hasta encontrar el conjunto con el mayor desempeño.

El ajuste a alcanzar es de 0.7 a lo largo de 100 generaciones. Esto quiere decir que el algoritmo genético repitió 100 veces sus interacciones hasta acercarse lo más posible al valor deseado. En la Figura 3 se muestra la frecuencia de aparición de los parámetros más significativos en todas las generaciones entrenadas del GA.

Tabla 4. Parámetros eliminados.

Parámetro	Definición
ACARBO	Miligramos de acarbosa tomados al día
ROSIGLI	Miligramos de rosiglitazona tomados al día
PIOGLI	Miligramos de pioglitazona tomados al día
INSUL	Miligramos de insulina tomados al día
GLIBEN	Miligramos de glibenclamina tomados al día
METFOR	Miligramos de metmorfina tomados al día

La Figura presenta el cromosoma más fuerte luego de 100 generaciones en color negro, las características que explican el gen más fuerte del algoritmo son GLIBENCLAMIDA, TG-B (mg.dl), CHOLU, INSULINA y DBPU. Para la validación hold-out utilizando el 30 % de los datos para evaluar el rendimiento, los sujetos se dividieron aleatoriamente en dos grupos, utilizando un total de 446 observaciones para el conjunto de prueba.

El conjunto de datos de validación hold-out de pruebas contenía 446 observaciones, 192 de los pacientes en el conjunto de datos de prueba presentaron una complicación y 254 no presentaron ninguna complicación. El modelo de predicción tuvo un rendimiento promedio de 118 observaciones correctamente clasificadas como pacientes con complicaciones y 175 observaciones correctamente clasificadas como pacientes sin complicaciones. La curva ROC del modelo se muestra en la Figura 4:

El AUC de la curva ROC de la Fig 4. es de 0.69. Un AUC de 0.5 se traduce en la posibilidad de clasificar aleatoriamente a los pacientes; un valor mayor indica que el modelo tiene capacidad predictiva [11]. Los parámetros de calidad para evaluar el modelo validado se muestran en la Tabla 2.

4. Discusión

La selección de características en el algoritmo genético usada en estudio permitió identificar las principales variables correlacionadas con el desarrollo de complicaciones en pacientes con diabetes tipo 2, por ejemplo, el tipo de hormona que el paciente usa.

En conjunto, el resultado de clasificación y la estimación del modelo permiten resaltar, al paciente o al médico, la prioridad que tienen las variables, por ejemplo, administrar el tipo correcto de la hormona al paciente o monitorear la presión vascular para el desarrollo de síntomas crónicos de gravedad.

Otras de las variables correlacionadas con el desarrollo de complicaciones fueron la presión alterial diastólica, los niveles de triglicéridos y los niveles de colesterol. De esta forma, es posible aportar directamente a la prevención y además al conocimiento del efecto a largo plazo de las características de la base de datos utilizada.

Un valor de 0.5 para AUC indica que la salida del modelo es aleatoria y, por lo tanto, el modelo no tiene capacidad para clasificar a los pacientes [12]. En este estudio, un número importante de pacientes se clasifican correctamente. Los resultados muestran un área ROC bajo la curva de 0.69.

Tabla 5. Lista de variables.

Nombre de la variable	Definición
Age	Edad del paciente
Age.Dx	Tiempo desde el diagnóstico de diabetes al paciente
BMI	Índice de masa corporal
Chol.mg.dl	Esteroide que modula la fluidez de las membranas biológicas, se encuentra en los tejidos. Precursor de hormonas esteroides, vitamina D y ácidos biliares. Valor de referencia colesterol total normal inferior a 200 mg / dL.
CHOLU	Concentraciones plasmáticas de colesterol total
Crea.Mg.Dl	Un anhídrido de creatina desechado en la orina. Su valor de referencia es de 0,5 a 1,3 mg / dL.
DBPU	Presión arterial diastólica Valor de referencia: 80 mm Hg
Gen.Bi	El sexo biológico del paciente
GLIBENCLA	La glibenclamida, también conocida como gliburida, es un medicamento que se usa para tratar la diabetes mellitus tipo 2. Presencia o ausencia en el tratamiento del paciente
Glu.Mg-Dl	Monosacarido, hexosa con eficiencia energética. Usado por los tejidos como una forma de energía. Valores de referencia 70 a 100 mg / dL
HA.TX	Hipertensión arterial. Se reduce a la presión arterial. Es una medida de la fuerza que se ejerce contra las paredes de las arterias cuando el corazón bombea sangre al cuerpo. Hipertensión es el término utilizado para describir la presión arterial alta. Presión normal y rango: 120/80 mm Hg.
HbA1c	Hemoglobina glucosilada, es una forma de hemoglobina que está químicamente ligada a un azúcar
HDLC.mg.dl	lipoproteínas de alta densidad. Limpian el colesterol de los tejidos del hígado y lo lleva al hígado. Valor de referencia mayor $<$ 50 mg / dL en mujeres y 40 mg / dL en hombres.
HDLU	Lipoproteinas de alta densidad. Niveles antes de recibir medicamentos para controlarlos
INSULINA	Tipo de insulina que el paciente recibe
LDLC.mg.dl	Lipoproteína de baja densidad que transporta el colesterol desde el hígado a los tejidos extrahepáticos. Valor de referencia inferior a 100 mg / dL
LDLU	Lipoproteinas de baja densidad. Niveles antes de recibir medicamentos para controlarlosl
Lipids.Tx	Moléculas orgánicas, insolubles en agua pero solubles en algunos disolventes polares. Tienen funciones estructurales, regulatorias y de reserva energética.
TG-B	Es un lípido que actúa como reserva energética. Valor de referencia de menos de 150 mg / dL
METFORMI	Metformina, es el medicamento más común para el tratamiento de la diabetes tipo 2, especialmente en personas con sobrepeso. hace referencia a si el paciente está siendo tratado con Metformina o no
PC1- PC5	Puntuaciones de PC (PC corresponde a la ascendencia NAM / EUR). Es una herramienta útil para el análisis de datos genéticos
SBPU	Presión arterial sistólica Valor de referencia: 120 mm Hg. La U se refiere al valor ajustado debido al tratamiento.
TG-B (mg/dl)	Niveles de triglicéridos

El número de pacientes que se clasificaron correctamente en este estudio indican que los algoritmos genéticos son una alternativa para realizar una clasificación. Los genes en el cromosoma de mayor desempeño encontrado por el algoritmo coinciden con las variables estudiadas en otros trabajos para predecir el desarrollo de algunas complicaciones tales como cardiopatías o retinopatías [8, 21].

Considerando distintos métodos de clasificación, las máquinas de soporte de vectores tuvieron un valor más alto de clasificación a comparación de la regresión lineal, regresión logística o random forest . Se hace la comparación del desempeño de estos métodos de clasificación considerando las variables del cromosoma más fuerte como características de entrada para cada modelo en la Tabla 3.

5. Trabajo futuro

Se planea diseñar una herramienta de predicción con este modelo estadístico, con el fin de hacer una detección temprana y/o predecir. También, se propone utilizar la metodología presentada para entrenar un modelo con un conjunto de datos más grande y considerando un tamaño de cromosoma mayor, con el fin de usar más parámetros al momento de la clasificación y esperando un desempeño aún mejor. Finalmente, se sugiere la comparación de diferentes algoritmos de clasificación para mejorar el rendimiento y además separar el tipo de complicaciones en diferentes grupos.

Referencias

1. American Diabetes Association: Diagnosis and classification of diabetes mellitus. Diabetes Care, vol. 36, pp. 62–67 (2009) doi: 10.2337/dc09-S062
2. Celisse, A., Robin, S.: A cross-validation based estimation of the proportion of true null hypotheses. Journal of Statistical Planning and Inference, vol. 140, no. 11, pp. 3132–3147 (2010) doi: 10.1016/j.jspi.2010.04.014
3. Chatterjee, S., Khunti, K., Melanie J. D.: Type 2 diabetes.The Lancet, vol. 53, pp. 2239–2251 (2017) https://www.thelancet.com/clinical/diseases/diabetes-type2
4. Dagliati, A., Marini, S., Sacchi, L., Cogni, G., Teliti, M., Tibollo, V., De Cata, P., Chiovato, L., Bellazzi, R.: Machine learning methods to predict diabetes complications. Journal of Diabetes Science and Technology, vol. 12, no. 2, pp. 295–302 (2017) doi: 10.1177/1932296817706375
5. Farran, B., Channanath, A. M., Behbehani, K., Thanaraj, T. A.: Predictive models to assess risk of type 2 diabetes, hypertension and comorbidity: Machine-learning algorithms and validation using national health data from Kuwait-a cohort study. BMJ Open, vol. 3, no. 5 (2013) doi: 10.1136/bmjopen-2012-002457
6. Ghorbani, A., Zou, J.: Data Shapley: Equitable valuation of data for machine learning. In: Proceedings of the International Conference on Machine Learning (2019) doi: 10.48550/ARXIV.1904.02868
7. Gutierrez, D. D.: Machine learning and data science: An introduction to statistical learning methods with R. Technics Publications (2015)
8. Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., Chouvarda, I.: Machine learning and data mining methods in diabetes research. Computational and Structural Biotechnology Journal, vol. 15, pp. 104–116 (2017) doi: 10.1016/j.csbj.2016.12.005
9. Kuhn, M.: Caret: Classification and regression training. Astrophysics Source Code Library (2015)
10. Kumar, R., Indrayan, A.: Receiver operating characteristic (ROC) curve for medical researchers. Indian Pediatrics, Springer Science and Business Media LLC, vol. 48, no. 4, pp. 277–287 (2011) doi: 10.1007/s13312-011-0055-4

11. Lasko, T. A., Bhagwat, J. G., Zou, K. H., Ohno-Machado, L.: The use of receiver operating characteristic curves in biomedical informatics. Journal of Biomedical Informatics, vol. 38, no. 5, pp. 404–415 (2005) doi: 10.1016/j.jbi.2005.02.008
12. Mandrekar, J. N.: Receiver operating characteristic curve in diagnostic test assessment. Journal of Thoracic Oncology, vol. 5, no. 9, pp. 1315–1316 (2010) doi: 10.1097/jto.0b013e3181ec173d
13. Nathan, D. M.: Long-Term complications of diabetes mellitus. New England Journal of Medicine, vol. 328, no. 23, pp. 1676–1685 (1993) doi: 10.1056/nejm199306103282306
14. Nguyen, L. H., Holmes, S.: Ten quick tips for effective dimensionality reduction. PLOS Computational Biology, vol. 15, no. 6, (2019) doi: 10.1371/journal.pcbi.1006907
15. Ogurtsova, K., da Rocha Fernandes, J. D., Huang, Y., Linnenkamp, U., Guariguata, L., Cho, N. H., Cavan, D., Shaw, J. E., Makaroff, L. E.: IDF Diabetes atlas: Global estimates for the prevalence of diabetes for 2015 and 2040. Diabetes Research and Clinical Practice, vol. 128, pp. 40–50 (2017) doi: 10.1016/j.diabres.2017.03.024
16. Pradhan, A.: Support vector machine-a survey. International Journal of Emerging Technology and Advanced Engineering 2.8, pp. 82–85 (2012).
17. R Core Team: A language and environment for statistical computing. R Foundation for Statistical Computing (2018) www.R-project.org/
18. Refaeilzadeh, P., Tang, L., Liu, H.: Cross-validation.encyclopedia of database systems, pp. 532–538 (2009) doi: 10.1007/978-0-387-39940-9_565
19. Scrucca, L.: GA: A package for genetic algorithms in R. Journal of Statistical Software, Foundation for Open Access Statistic, vol. 53, no. 4 (2013) doi: 10.18637/jss.v053.i04
20. Trevino, V., Falciani, F.: GALGO: An R package for multivariate variable selection using genetic algorithms. Bioinformatics, vol. 22, no. 9, pp. 1154–1156 (2006) doi: 10.1093/bioinformatics/btl074
21. Tsao, H. Y., Chan, P. Y., Su, E. C.: Predicting diabetic retinopathy and identifying interpretable biomedical features using machine learning algorithms. BMC Bioinformatics, vol. 19, no. S9 (2018) doi: 10.1186/s12859-018-2277-0
22. Vanslette, K., Tohme, T., Youcef-Toumi, K.: A general model validation and testing tool. Reliability Engineering and System Safety, vol. 195 (2020) doi: 10.1016/j.ress.2019.106684
23. Whitley, D.: A genetic algorithm tutorial. Statistics and Computing, Springer Science and Business Media LLC, vol. 4, no. 2 (1994) doi: 10.1007/bf00175354
24. Zhang, Z.: Missing data imputation: Focusing on single imputation. Annals of Translational Medicine, vol. 4, no. 1 (2016) doi: 10.3978/j.issn.2305-5839.2015.12.38

ISSN 1870-4069

Mask-net: Identificación del uso correcto de mascarilla mediante visión por computador

Alexander Kalen-Targa, Alberto Landi-Cortiñas,
Nicolas Araque-Volk, Alejando Marcano Van-Grieken

Universidad Metropolitana,
Facultad de Ingeniería,
Venezuela

{alexanderkalen, albertolandi}@correo.unimet.edu.ve,
{naraque, amarcano}@unimet.edu.ve

Resumen. Este trabajo se enfoca en crear un sistema de reconocimiento de uso correcto de una mascarilla mediante técnicas de visión por computador. Se realizó una investigación con el objetivo de establecer los lineamientos para el uso correcto e incorrecto de la mascarilla, los cuales permitieron determinar los criterios para la creación de los conjuntos de los datos "KLD" e "IMKLD". Dichos conjuntos sirvieron para entrenar, validar y probar los modelos "Mask-net" e "I-Mask-net" respectivamente. Estos son modelos de arquitectura de Aprendizaje Profundo, donde se aplicó la técnica de "Aprendizaje por Transferencia" al utilizar "MobileNet" como una base para la extracción de atributos. Los resultados de los entrenamientos arrojaron que el ajuste de hiperparámetros realizado fue el adecuado en ambos casos, mientras que las pruebas hechas demostraron que los modelos tienen un buen porcentaje de exactitud.

Palabras clave: Mascarilla, visión por computador, inteligencia artificial, aprendizaje profundo, redes neuronales.

Mask-net: Detection of Correct Use of Masks through Computer Vision

Abstract. This paper focuses on creating a system for recognizing the correct use of a mask through computer vision techniques. Research was carried out with aims of establishing guidelines for the correct and incorrect use of a mask, which allowed for determining the criteria for the creation of the "KLD" and "IMKLD" datasets. These datasets were used to train, validate and test the "Mask-net" and "I-Mask-net" models respectively. The results given by training showed that the fine tuning carried out was adequate in both cases, while the tests carried out showed that the models have an acceptable level of accuracy.

Keywords: Masks, computer vision, artificial intelligence, deep learning, neural networks.

1. Introducción

El 11 de marzo del 2020, la Organización Mundial de la Salud (OMS), declaró el COVID-19, también conocido como coronavirus, como una pandemia a nivel mundial. Actualmente, miles de millones de personas han sido afectadas directa o indirectamente por el coronavirus. A nivel mundial, se han notificado más de 100 millones de casos confirmados de COVID-19, incluidas más de 2 millones de muertes, notificadas a la OMS [1]. Las medidas de emergencia obligatorias para controlar y evitar una mayor propagación del virus han causado cambios importantes en el estilo de vida de los seres humanos.

Uno de los problemas respecto al cumplimiento de medidas de bioseguridad es el uso incorrecto de la mascarilla; estudios preliminares realizados indican que [2] "la enfermedad se propaga principalmente de persona a persona a través de las gotículas que salen despedidas de la nariz o la boca de una persona infectada al toser, estornudar o hablar". Por lo que el uso incorrecto de la misma pudiese generar más contagios, saturando los sistemas de salud y causando una mayor cantidad de fallecimientos.

Actualmente, el uso de la mascarilla es una de las medidas de prevención más efectivas a nivel mundial, y los primeros estudios realizados respecto a su uso arrojan resultados prometedores, como el realizado por Mitze et al. [3], donde afirman en un estudio hecho en Jena, Alemania, que el uso de la mascarilla ayudó a reducir en un 60 % la tasa de crecimiento diario de infecciones.

Ante estos beneficios, es indispensable extender la normativa de uso obligatorio de la mascarilla, fundamentalmente en espacios cerrados y/o de gran concentración de personas, y controlar que sean utilizadas correctamente. Los modelos de visión computacional son utilizados para resolver problemáticas complejas relevantes a imágenes o videos y pudiesen ser diseñados y entrenados para predecir el uso correcto de una mascarilla en una persona y así ayudar a solucionar este problema.

Seguidamente, se conoce que las redes neuronales convolucionales (CNN) son las redes más comunes en el campo de la visión por computador. Para Goodfellow et al. [4], son un tipo de red neuronal para el procesamiento de la data con una topología matricial conocida en forma de cuadrícula, siendo tremendamente exitosas en aplicaciones prácticas. La red emplea una operación matemática llamada convolución, siendo tremendamente exitosa para esta aplicación.

Utilizar un modelo de inteligencia artificial para detectar el uso correcto de la mascarilla sin intervención humana puede ser de gran utilidad en espacios públicos y privados de alto tránsito de personas, ya que se evitaría la revisión manual de la misma, promoviendo así las medidas de distanciamiento social con una revisión fiable. Según Johns Hopkins Medicine & Maragakis [5], se considera el uso correcto de la mascarilla aquellos casos en los que se tapan la nariz, la boca y la barbilla de una cara en su totalidad, consiguiendo así la mayor eficacia de la mascarilla facial.

Por otro lado, se considera uso incorrecto de la mascarilla cualquier caso en el que la nariz, la boca o la barbilla de un rostro estén expuestos, parcial o totalmente. Partiendo del argumento anterior, el objetivo de la presente investigación es implementar un sistema de reconocimiento de uso correcto de una mascarilla médica mediante técnicas de visión por computador.

Fig. 1. Cambios de color de mascarillas en muestras.

Para ello, se desarrolló el modelo de aprendizaje profundo "Mask-net" utilizando la arquitectura de red neuronal convolucional para reconocer el uso correcto, incorrecto, o no uso de la mascarilla. Adicionalmente, se creó otro modelo de aprendizaje profundo "I-Mask-Net", que utiliza la salida del modelo anterior cuando se identifica uso incorrecto de la misma, este modelo indica si la mascarilla está en la barbilla, en la boca y la barbilla, o en la nariz y la boca, para poder brindar una sugerencia al usuario sobre cómo portarla correctamente.

2. Metodología

2.1. Conjunto de datos

Puesto que no existe un conjunto de datos preconstruido que se adhiera a los parámetros preestablecidos, la creación y diseño de conjuntos de datos personalizados de alta calidad se convirtió en uno de los enfoques de mayor importancia para el desarrollo de los modelos de aprendizaje automático Mask-net e I-Mask-net. Dicho lo anterior, se construyeron dos conjuntos de datos: KLD e IMKLD.

KLD es un conjunto de datos compuesto por un total de 11.003 imágenes, con el objetivo de entrenar y validar el modelo Mask-net. Las imágenes seleccionadas se originaron de múltiples conjuntos de datos de dominio público disponibles en internet, y algunas de autoría propia, entre las cuales se incluyen:

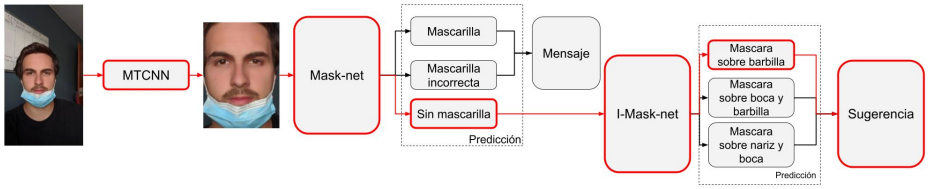


Fig. 2. Implementación combinada de los modelos Mask-net e I-Mask-net.

Tabla 1. Matriz de confusión de Mask-net.

		PREDICCIÓN		
		Mascarilla	**Mascarilla incorrecta**	**Sin mascarilla**
CLASE REAL	**Mascarilla**	55	1	0
	Mascarilla incorrecta	13	42	2
	Sin mascarilla	4	3	50

– Medical Mask Dataset [6], que proporcionó imágenes de personas reales usando mascarilla correctamente.

– Flickr-Faces-HQ [7], que proporcionó imágenes de personas reales, sin mascarilla.

– MaskedFace-net [8], que proporcionó imágenes de personas usando mascarilla médica de manera correcta e incorrecta.

– Autoría propia, un conjunto de imágenes que captaron situaciones del mundo real, tomando en cuenta ángulos de toma, fondos y uso de la mascarilla, así como bordados, colores y tipos utilizadas en el conjunto de pruebas. Dichas imágenes fueron proporcionadas por familiares, amigos y conocidos.

IMKLD es un conjunto de datos compuesto por un total de 1.841 imágenes, con el objetivo de entrenar y validar el modelo I-Mask-net. Las imágenes seleccionadas se originaron en el conjunto de datos MaskedFace-net, el cual proporcionó las tres clases de imágenes de uso incorrecto de la mascarilla. Además, algunas imágenes de autoría propia fueron incorporadas a dicho conjunto para incrementar la cantidad de casos reales.

Las imágenes están clasificadas en 3 clases según las partes del rostro cubiertas por la mascarilla: barbilla, boca y barbilla, y nariz y boca. Los conjuntos de datos KLD e IMKLD fueron almacenados localmente, donde los archivos fueron separados en conjuntos de entrenamiento, validación y prueba, cada uno subdividido en subdirectorios según las clases de dicho conjunto de datos.

KLD se distribuyó en 8.804 imágenes para entrenamiento, 2.199 imágenes para validación y 170 imágenes para prueba. Mientras que el conjunto de imágenes IMKLD se distribuyó en 1.610 imágenes de entrenamiento, 231 imágenes de validación y 57 imágenes de prueba.

Tabla 2. Métricas de presición (presicion), exhaustividad (recall) de Mask-net.

Clase	Presición	Exhaustividad
Mascarilla	0,7638	0,9821
Mascarilla incorrecta	0,9130	0,7368
Sin mascarilla	0,9615	0,8771

2.2. Preprocesamiento de imágenes

Con el objetivo de mantener un orden en los conjuntos de datos, es necesario establecer una nomenclatura común. Para ello, se creó un algoritmo con el que se enumeraron y renombraron cada una de las miles de imágenes pertenecientes a ambos conjuntos de datos. También se consideró fundamental reducir la información innecesaria (tales como fondos) que pudiese generar sesgo en los datos, así como una reducción en la velocidad de entrenamiento del modelo.

Esta problemática se resolvió al establecer una Región de Interés (RoI) que envolviese los rostros de las muestras suministradas. Utilizando un algoritmo en Python, se procesaron las imágenes de ambos conjuntos de datos. Alimentando iterativamente cada una de las imágenes a MTCNN [9], se marcaron los bordes detectados de la Región de Interés, procediendo a recortar y posteriormente guardar la imagen de salida, teniendo como resultado solo los rostros de los sujetos para cada una de las muestras.

Por último, un problema de sesgo encontrado en ambos conjuntos corresponde al color azul de la mascarilla quirúrgica, ya que estas imágenes forman un porcentaje importante de la totalidad disponible. Para mitigar esto, fue necesario suministrar la mayor variación en colores y patrones de mascarillas de manera que el modelo a entrenar generalice lo mejor posible.

Una solución a esta problemática involucró la construcción de un algoritmo, que detecta zonas con la presencia de un rango de color seleccionado (en el espacio de color HSV), en este caso, tonalidades de color azul claro presente en las mascarillas quirúrgicas. Luego de detectar dichas zonas de coincidencia en las imágenes suministradas, se comenzó a plasmar sobre tales áreas otras coloraciones y tonalidades proporcionadas por valores aleatorios del espacio de color BGR, de tal manera que la variedad de mascarillas encontradas fuese más cercanas a la realidad.

2.3. Arquitectura de la red neuronal

Mask-net e I-Mask-net fueron diseñados con arquitecturas similares haciendo uso del Aprendizaje por Transferencia, lo que implicó el uso de una arquitectura predeterminada, siendo MobileNet el modelo preferido para construir la arquitectura base.

El modelo Mask-net cuenta con 85 capas profundas, de las cuales 82 son capas de extracción de atributos de MobileNet y 3 son capas de ajuste propio (fine tuning), que corresponden a una capa oculta de 512 nodos, seguida de una capa de Dropout con un índice de 0,2, culminando con una capa de salida para las tres clases determinadas, con función de activación Softmax.

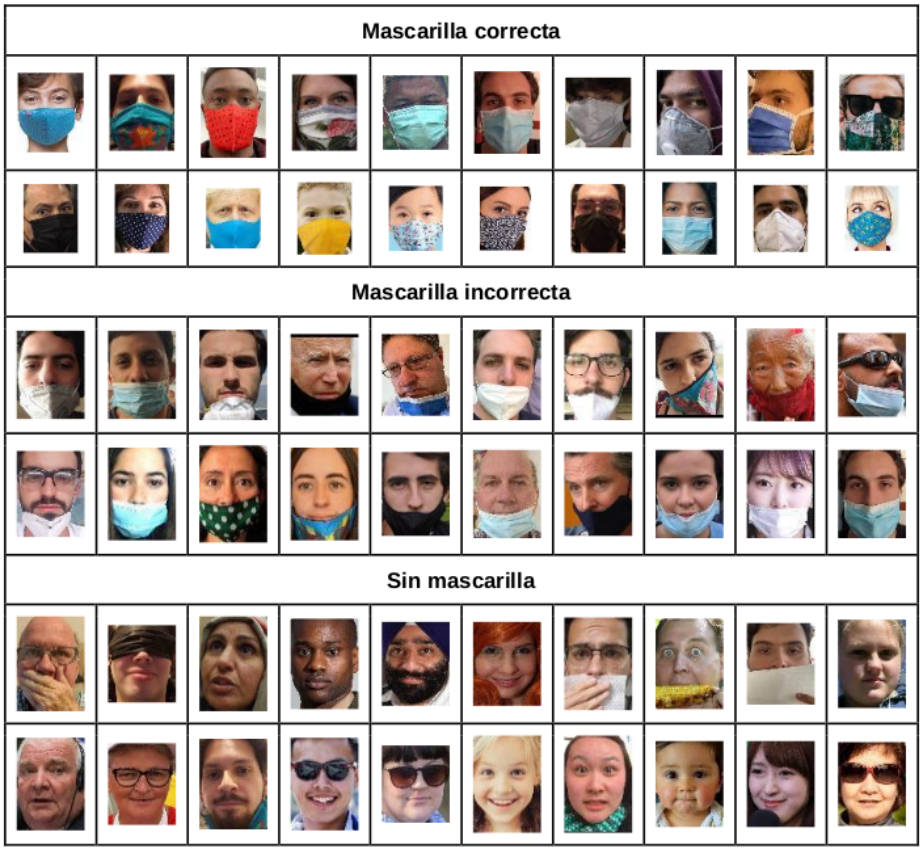


Fig. 3. Predicciones correctas del Modelo Mask-net por clases.

Por otra parte, I-Mask-net cuenta con 84 capas profundas y 2 de ajuste propio, donde se conectó una capa de Dropout con un índice de 0,6, culminando con una capa de salida de función de activación Softmax, para las tres clases determinadas. Posteriormente, se congelaron ambos modelos con pesos correspondientes al preentrenamiento realizado por MobileNet con el conjunto de datos ImageNet, siendo entrenables las últimas 23 capas de Mask-net, y las últimas 34 capas en el caso de I-Mask-net.

Al congelar las capas, se permitió al modelo retener conocimientos de atributos generales extraídos de ImageNet, conjunto de datos que presenta clases similares a KLD (mask, respirator, gas mask, gas helmet, ski mask, oxygen mask), y diferentes en el caso de IMKLD, debido a que no existen clases específicas.

2.4. Entrenamiento de los modelos

Para el modelo Mask-Net, el entrenamiento final fue realizado localmente en Jupyter Notebooks, haciendo uso de aceleramiento de hardware por GPU “Nvidia GTX 1050ti with Max-Q Design”.

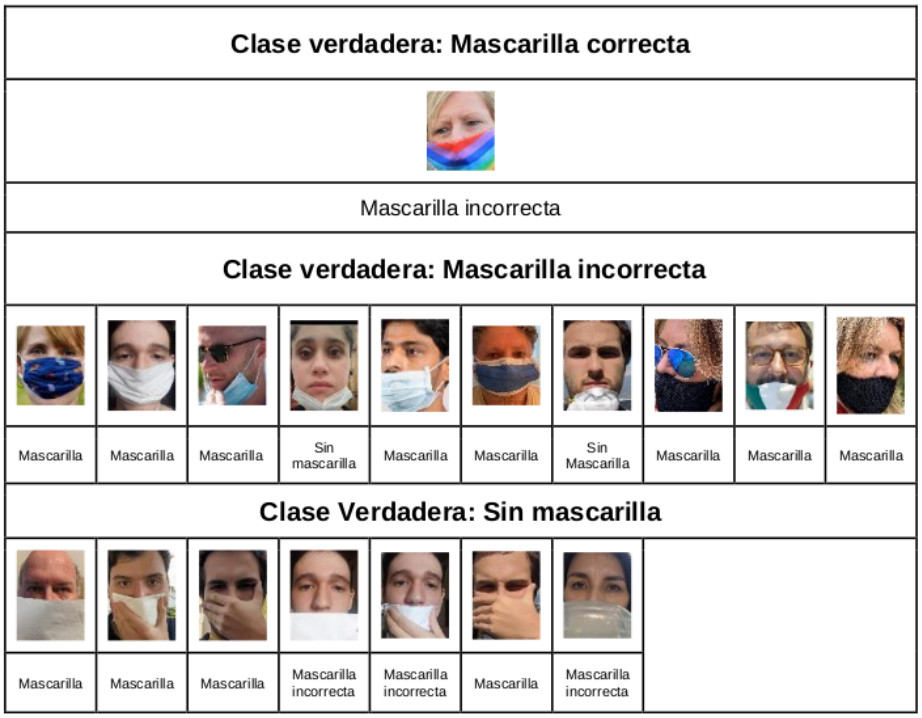


Fig. 4. Predicciones incorrectas del Modelo Mask-net por clases.

Tabla 3. Matriz de confusión de I-Mask-net.

		PREDICCIÓN		
		Barbilla	**Boca y barbilla**	**Nariz y boca**
CLASE REAL	**Barbilla**	16	2	1
	Boca y barbilla	2	14	3
	Nariz y boca	0	0	19

Se usó la inicialización de parámetros “random uniform”, con media 0 y desviación estándar 1, implementado sobre la capa lineal de 512 nodos; además, se regularizaron los pesos de dicha capa con regularizador L2 igualado a 0,01. El modelo se compiló utilizando el optimizador “Adam” con tasa de aprendizaje igualada en 0,0001 y función de pérdida “Categorical Cross Entropy”.

Se hizo uso de técnicas de aumento de datos implementando volteado horizontal y la rotación de imágenes respecto a su eje vertical hasta 20°. Además, el uso de la capa de Dropout que prevendría el sobreajuste durante el entrenamiento de Mask-net, el cual se llevó a cabo durante 10 iteraciones para evitar el sobreaprendizaje del modelo, tomando así 18 minutos para completarse.

Matriz de Confusión del modelo Mask-net, en I-Mask-net, se compiló utilizando el optimizador “Adam” con tasa de aprendizaje igualada a 0,00001 para frenar la velocidad de entrenamiento y con función de pérdida “Categorical Cross Entropy”.

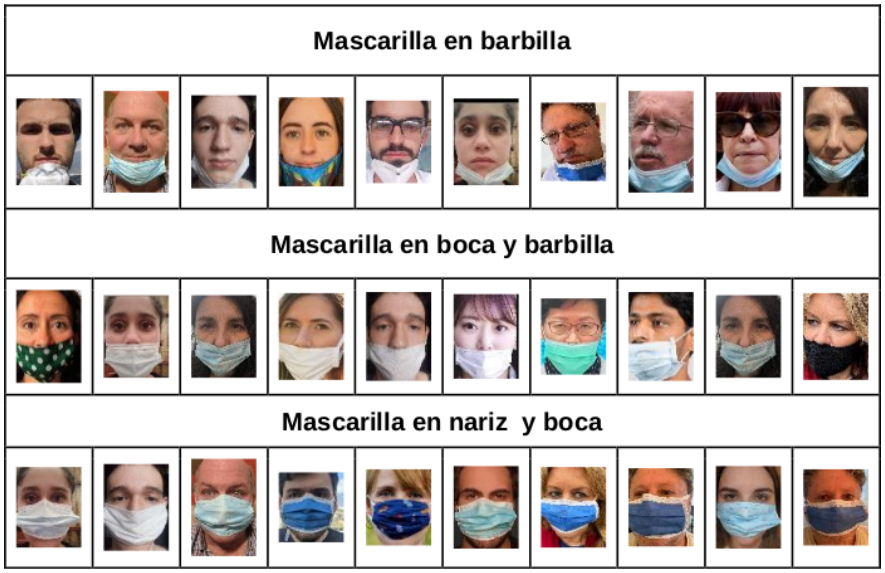


Fig. 5. Predicciones correctas del Modelo I-Mask-net por clases.

Se compiló utilizando el optimizador “Adam” con tasa de aprendizaje igualada a 0,00001 para frenar la velocidad de entrenamiento y con función de pérdida “Categorical Cross Entropy”.

3. Resultados

Utilizando un grupo de 170 imágenes correspondientes al conjunto de prueba de KLD, se corrieron predicciones sobre el modelo Mask-net. De la matriz de confusión visible en la Tabla 1 y de los datos de exhaustividad de la Tabla 2, se infiere que Mask-net predijo acertadamente en el 98,21 % de las veces que se le presenta un rostro portando una mascarilla correctamente. Cuando se le presentaron rostros portando la mascarilla incorrectamente, Mask-net fue capaz de predecir ciertamente dicho caso en el 73,68 % de las veces.

Cuando se le presentó uno sin mascarilla, Mask-net predijo correctamente dicho caso en el 87,71 % de las veces, lo que da un porcentaje de exactitud del 86,47 % sobre todo el conjunto de prueba (overall accuracy). Como se observa en la Figura 3, Mask-net es capaz de predecir correctamente imágenes diversas, con personas de diferentes etnicidades y edades, portando variedad de mascarillas en patrones y colores, accesorios y objetos obstruyendo su rostro.

Incluso, se notó que Mask-net puede acertar en algunas ocasiones aunque usuarios intenten engañarlo. Como se observa en la Figura 4, Mask-net presentó algunas fallas particularmente en las clases “Mascarilla incorrecta” y “Sin mascarilla”. Las fallas en mascarilla incorrecta se dieron sobre todo en imágenes donde la barbilla estaba descubierta e imágenes cuyos sujetos no muestran su rostro de manera completamente frontal.

Tabla 4. Métricas de presición (presicion), exhaustividad (recall) de I-Mask-net.

Clase	Presición	Exhaustividad
Barbilla	0,8888	0,8421
Boca y barbilla	0,8750	0,7368
Nariz y boca	0,8260	1,0000

Las fallas en rostros sin mascarilla son pertinentes a individuos intentando engañar al modelo utilizando algún objeto o inclusive su mano, situaciones de carácter particular. Por otro lado, Utilizando un grupo de 57 imágenes correspondientes al conjunto de prueba de IMKLD, se corrieron las siguientes predicciones sobre el modelo Mask-net. Ver Tabla 2. De la matriz de confusión de la Tabla 3 y de los datos de exhaustividad de la Tabla 4, se entiende que I-Mask-net predijo acertadamente en el 84,21 % de las veces que se le presentó un rostro portando una mascarilla sobre la barbilla.

Cuando se le presentó al modelo situaciones donde se cubría la boca y barbilla y se dejaba expuesta la nariz, I- Mask-net predijo correctamente dicho caso en el 73,68 % de las veces. Cuando se le presentaron rostros portando la mascarilla sobre la nariz y boca y se dejaba la barbilla expuesta, I-Mask-net fue capaz de predecir ciertamente dicho caso en el 100 % de las veces. Lo que da un porcentaje de exactitud del 85,96 % sobre todo el conjunto de prueba (overall accuracy).

Como se observa en la Figura 5, I-Mask-net es capaz de predecir correctamente imágenes de las diferentes clases en la vasta mayoría de los casos estudiados. Como se observa en la Figura 6, I-Mask-net cometió errores en la clase “barbilla”, cuando el sujeto tiene la mascarilla cercana a su boca.

Por otra parte, la clase “boca y barbilla” sufrió mayor cantidad de errores en aquellas imágenes donde la mascarilla estaba superpuesta sobre la boca del individuo. Tomando en cuenta lo expuesto anteriormente, vale la pena acotar que el resto de los errores del modelo I-Mask-net se debe a una falta de generalización en algunas de las imágenes de prueba. Esto probablemente ocurre debido a que el conjunto de entrenamiento y validación de IMKLD no es lo suficientemente extenso para generalizar correctamente.

4. Estado del arte

A continuación se comparan los resultados experimentales de Mask-net con resultados de trabajos publicados destacados en la identificación de mascarillas. Es importante acotar que estos trabajos no resuelven la misma problemática que resuelve Mask-net, ya que la misma se enfoca no solamente en detectar si una persona tiene la mascarilla puesta, sino tambien si la tiene puesta de manera correcta.

Sin embargo, sirven como referencia para medir el rendimiento de los modelos debido a su similaridad al utilizar Aprendizaje por Transferencia y librerías como MTCNN para la detección de los rostros. Las investigaciones disponibles a continuación son comparadas en la Tabla 5:

– RetinaFaceMask: A Face Mask detector [10]. RetinaFaceMask utiliza múltiples mapas de funciones y luego utiliza funciones red piramidal (FPN) para fusionar

Fig. 6. Predicciones incorrectas del Modelo I-Mask-net por clases.

la información semántica de alto nivel. La investigación hace uso del algoritmo del estado del arte para la detección de rostros MTCNN [9], de manera similar a Mask-net, con el cual se presenta mejor precisión al detectar rostros con mascarilla con ángulos de toma diferentes en comparación al usar Haard Cascade [10].

– Multi-Stage CNN Architecture for Face Mask Detection [11]. Es un sistema que consta en una arquitectura CNN de dos etapas, la cual es capaz de detectar rostros con y sin mascarilla. En esta investigación se hace uso de el Aprendizaje por Transferencia con arquitecturas distintas, preentrenando dichas redes con el conjunto de datos ImageNet, de manera similar a Mask-net.

Comparando a RetinaFaceMask, se halla que Mask-net es similar en precisión a RetinaFaceMask en el caso de predecir sobre la clase mascarilla, alcanzando hasta un 98,21 % de exhaustividad. Cuando se presentan rostros descubiertos (sin mascarilla), Mask-net supera a RetinaFaceMask alcanzando un 96,15 % de precisión, manteniendo una exhaustividad similar a este último.

Por otra parte, los resultados de Multi-Stage CNN fueron superiores utilizando un camino parecido al de MaskNet, utilizando el mismo detector de rostros MTCNN y al utilizar una arquitectura CNN con Aprendizaje Por transferencia.

5. Conclusiones

Procesar las muestras utilizando la detección de rostros por MTCNN aumentó la exactitud de Mask-net e I-Mask-net ante casos reales. Igualmente, efectuar cambios de color a un grupo extenso de mascarillas introdujo variedad en los conjuntos de datos KLD e IMKLD, lo cual disminuyó el sesgo de predicción y optimizó la generalización de los modelos entrenados. Esto permitió a los mismos clasificar correctamente sin importar color o bordado de la mascarilla.

Tabla 5. Tabla comparativa de Precisión (precision) y Exhaustividad (recall) de Mask-net con RetinaFaceMask y Multi-Stage CNN para clases equivalentes.

Investigación	Base	Rostro		Mascarilla	
		Precision	**Recall**	**Precision**	**Recall**
RetinaFaceMask	Mobilenet	79,00 %	92,80 %	78,90 %	89,10 %
	Resnet	91,50 %	95,60 %	93,30 %	94,40 %
Multi-Stage CNN	MobilenetV2	N/D	N/D	99,12 %	99,20 %
	Densenet121	N/D	N/D	99,70 %	99,12 %
Mask-net	Mobilenet	96,15 %	87,71 %	76,38 %	98,21 %

Se desarrolló y entrenó el modelo de visión por computador Mask-net para predecir el uso correcto de la mascarilla, presentando una exactitud en predicción de 86,47 % sobre el conjunto de prueba KLD. Además, se desarrolló y entrenó un modelo adicional, I-Mask-net, para complementar a este.

Dicho modelo manifestó una exactitud en predicción de 85,96 % sobre el conjunto de prueba IMKLD. Uno de los principales aportes de Mask-net a diferencia de otros modelos desarrollados es no solo poder identificar mascarillas médicas, sino también identificar diversidad de mascarillas de tela de variedad de colores y bordados, y respiradores como el tipo N95.

Ambos modelos mostraron un comportamiento altamente satisfactorio, con algunas excepciones. Mask-net no pudo predecir correctamente imágenes de rostros donde la mascarilla era portada incorrectamente con nariz y boca cubiertas y barbilla expuesta, siendo erróneamente predichas como mascarilla.

Por otra parte, Mask-net e I-Mask-net presentaron ciertas diferencias en precisión debido principalmente a la diferencia en extensión en los conjuntos de datos KLD e IMKLD, así como sus susodichos conjuntos de prueba utilizados.

Considerando las conclusiones realizadas anteriormente, se evidencia que es posible detectar el uso correcto e incorrecto de la mascarilla a través de un sistema de reconocimiento facial utilizando visión por computador. Dicho esto, se efectúan las siguientes recomendaciones:

– Se propone incrementar la cantidad de muestras reales en los conjuntos de datos KLD e IMKLD, para abarcar la mayor cantidad posible de situaciones que se puedan presentar, como más ángulos de toma, y casos particulares, como, por ejemplo: casos de individuos utilizando una mayor variedad de accesorios, estilos de cabello distintos y más opciones de vello facial.

– Se sugiere modificar el conjunto de datos KLD, desglosando la clase "mascarilla incorrecta" en las subsiguientes clases: "barbilla", "boca y barbilla" y "nariz y boca", además de las existentes "mascarilla" y "sin mascarilla", totalizando en cinco clases; con el objetivo de entrenar un modelo multiclase, como Mask-net, y así comparar si esta alternativa es más eficiente que utilizar dos modelos independientemente entrenados.

– Se recomienda probar con diferentes modelos de Aprendizaje por Transferencia para la extracción de características, con el fin de evaluar si el desempeño del modelo mejora, caso similar al de RetinaFaceMask, ya que se vió beneficiado en un incremento de sus niveles de precisión y exhaustividad al implementar el modelo de Aprendizaje por Transferencia Resnet.

Referencias

1. Cabani, A., Hammoudi, K., Benhabiles, H., Mahmoud, M.: Maskedface-net – A dataset of correctly/incorrectly masked face images in the context of COVID-19, vol. 2019, no. 2021, pp. 2352–6483 (2020) doi: 10.1016/j.smhl.2020.100144
2. Chavda, A., Dsouza, J., Badgujar, S., Damani, A.: Multi-Stage CNN architecture for face mask detection. In: Proceedings of the 6th International Conference for Convergence in Technology (2021) doi: 10.1109/i2ct51068.2021.9418207
3. Fan, X., Jiang, M.: Retinafacemask: A single stage face mask detector for assisting control of the COVID-19 pandemic (2020) doi: 10.48550/arXiv.2005.03950
4. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT Press (2016) www.deeplearningbook.org
5. Hopkin, J., Maragakis, L. L.: How to properly Wear a face mask (2020) www.hopkinsmedicine.org/health/conditions-anddiseases/coronavirus/proper-mask-wearing-coronavirus-prevention-infographic
6. Humans in the loop.: Medical Mask Dataset (2020) humansintheloop.org/resources/datasets/medical-mask-dataset/
7. Jebril, N.: Declared a pandemic public health Mmnace: A systematic review of the coronavirus disease (COVID19). World Health Organization, pp. 12 (2020) www.psychosocial.com/article/PR290311/25748/
8. Karras, T., Laine, S. Aila, T.: A style-based generator architecture for generative adversarial networks. Conference on Computer Vision and Pattern Recognition, pp. 4401–4410 (2019) doi: 10.48550/arXiv.1812.04948
9. Mitze, T., Kosfeld, R., Rode, J., Wälde, K.: Face masks considerably reduce COVID-19 cases in Germany: A synthetic control method approach, vol. 177, no. 51, pp. 32293–32301 (2020) doi: 10.1073/pnas.2015954117
10. World Health Organization: Coronavirus (COVID19) dashboard with vaccination data (2020) covid19.who.int/?gclid
11. Zhang, K., Zhang, Z. Li, Z., Qiao, Y.: Joint face detection and alignment using multi-task cascaded convolutional networks. IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503 (2016) doi: 10.1109/LSP.2016.2603342

Electronic edition
Available online: http://www.rcs.cic.ipn.mx

http://rcs.cic.ipn.mx

Centro de Investigación
en Computación