

## 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 processing. In this article we present extraction methods based on machine learning and dictionaries for the identification 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 Daturks<sup>1</sup>. 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 cardíaca (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.

<sup>1</sup> <https://docs.daturks.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.

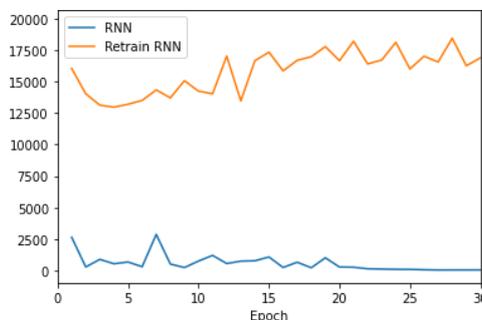
Tomando en cuenta los errores humanos al escribir las notas médicas se propuso utilizar la distancia de *Levenshtein* [6], que toma la distancia entre dos cadenas de caracteres como el número mínimo de ediciones (es decir, inserciones, eliminaciones o sustituciones) de un solo carácter que son necesarias 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 "."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 "."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<sup>2</sup> los detalles de la arquitectura pueden encontrarse en su página 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 pérdidas que presenta son mucho mayores.

<sup>2</sup> <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<sup>3</sup>.

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
```

<sup>3</sup> <https://code.google.com/archive/p/negex/>

Al pase de visita se refiere con **disnea de pequeños esfuerzos, niega cefalea, fiebre, dolor.**



**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.

positivo razon de 1 botella al mes. ¿? Exposición biomasa: preguntado negado. Inicia su padecimiento actual el día 14/07/2020 al presentar **cefalea SINTOMA** EVA 8/10 de predominio frontotemporal bilateral, **fiebre SINTOMA** cuantificada en 38.9°C, **tos SINTOMA** en accesos no cianozante, no disneizante, no hemetizante, malestar general, **conjuntivitis SINTOMA**, **anosmia SINTOMA**, **disgeusia SINTOMA**, **hiporexia SINTOMA**, **disnea medianos esfuerzos SINTOMA**, **mialgias SINTOMA**, **artralgias SINTOMA**, **diaforesis nocturna SINTOMA**, refiere manejo con paracetamol, refiere mejoría en la sintomatología sin embargo aun no ceden los síntomas, por lo que acude al Triage de Tlalpan el día de hoy 24/07/2020 donde se decide su ingreso nuestra unidad temporal COVID 19. El paciente refiere prueba de RT-PCR para SARS COV2 el día 22/07/2020 en Clínica del IMSS en Tlalpan, comenta que los resultados se esperan el día 27/07/2020 mediante forma telefonica. Actualmente el paciente se refiere **asintomático SINTOMA**. Objetivo Se recibe < paciente> de edad aparente similar la cronológica.

**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 <i>Levenshtein</i>	Sin distancia de <i>Levenshtein</i>
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 está 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 language 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)

*Dalia Cruz-Aguirre, Helena Gómez-Adorno, Armando Rios-Lastiri*

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