EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

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

Research in Computing Science

Vol. 154 No. 2
February 2025

Research in Computing Science

***RESEARCH IN COMPUTING SCIENCE**, Año 25, Volumen 154, No. 2, Febrero de 2025, 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 Othón 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, otorgado por el Instituto Nacional del 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 Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Fecha de última modificación 07 de Febrero de 2025.*

***RESEARCH IN COMPUTING SCIENCE**, Year 25, Volume 154, No. 2, February, 2025, is a monthly publication edited by the National Polytechnic Institute through the Center for Computing Research. Av. Juan de Dios Bátiz S/N, Esq. Miguel Othón de Mendizábal, Nueva Industrial Vallejo, C.P. 07738, Mexico City, Tel. 57 29 60 00, ext. 56571. https://www.rcs.cic.ipn.mx. Editor in charge: Dr. Grigori Sidorov. Reservation of Exclusive Use Rights of Title No. 04-2019-082310242100-203. ISSN: pending, granted by the National Copyright Institute. Responsible for the latest update of this issue: the Computer Research Center, Dr. Grigori Sidorov, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Last modified on February 7, 2025.*

Advances in Computing Science and Applications

Ana María Magdalena Saldaña Pérez (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2025

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

Reconocimiento de habla interna usando señales EEG: Estado del arte y metodología propuesta

Luciana Espindola-Ulibarri, Olga Kolesnikova,
Rolando Menchaca-Méndez

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
México

{lespindolau2024, kolesnikova, rmen}@cic.ipn.mx

Resumen. En este trabajo se abordarán las principales características del habla interna que mostrarán un panorama general para comprender los elementos que la conforman, aunado a la exposición de una técnica llamada electroencefalografía (EEG), apropiada para el desciframiento y reconocimiento del habla interna debido a que posee rasgos convenientes para esta tarea, como lo es su alta resolución temporal y su practicidad al ser un método no invasivo. Además, se propone una metodología para analizar, procesar y reconocer el habla interna a partir de señales EEG usando dos conjuntos de datos basados en un vocabulario reducido de palabras que representan comandos.

Palabras Clave: Habla interna, electroencefalografía (EEG) y alta resolución temporal.

Inner Speech Recognition Using EEG Signals: State of the Art and Proposed Methodology

Abstract. In this work we will address the main characteristics of internal speech that will show a general overview to understand the elements that make it up, together with the exposure of a technique called electroencephalography (EEG), appropriate for the decipherment and recognition of internal speech because it has convenient features for this task, such as its high temporal resolution and its practicality as it is a non-invasive method. In addition, a methodology is proposed to analyze, process and recognize internal speech from EEG signals using two data sets based on a reduced vocabulary of words that represent commands.

Keywords: Inner speech, electroencephalography (EEG), high temporal resolution.

1. Introducción

Actualmente se presenta el problema de llegar a descifrar el habla interna mediante técnicas que consisten en registrar la actividad neuronal, como lo es la electroencefalografía (EEG, por sus siglas en inglés) esto con el principal fin de crear mejores medios de comunicación para quienes padecen enfermedades como la esclerosis lateral amiotrófica (ELA), la embolia (ictus cerebral), las lesiones de médula espinal o cerebral, la parálisis cerebral, las distrofias musculares, la esclerosis múltiple, entre otros padecimientos. Por esta razón es de gran importancia conocer lo que caracteriza al habla interna, así como sus componentes y propiedades principales, para que partiendo de esta información sea posible tomar decisiones informadas acerca de los métodos y técnicas más pertinentes para su predicción. Destacando que el procesamiento, desciframiento y decodificación del habla interna puede ser la llave para introducir una nueva era de comunicación humana pasando de la comunicación común por voz o basada en texto a la comunicación basada en el cerebro [1].

Se ha demostrado que las señales EEG son apropiadas para descifrar el habla interna gracias a que tienen una alta resolución temporal, lo que les permite captar los cambios en las señales sobre el tiempo a nivel de milisegundos y producir una imagen de la actividad eléctrica en el cerebro, la cual es representada como ondas con variación de frecuencia, amplitud y forma basada en parámetros, además de ser un método no invasivo, que no daña el cerebro y que no requiere cirugía [2-5].

Para poder llegar a comprender los principales componentes y características conocidos hasta ahora del habla interna este trabajo está estructurado de la siguiente manera: La sección II introduce los antecedentes del habla interna, en la sección III se establecen un conjunto de definiciones del habla interna, la sección IV expone la relación entre el habla interna y el habla externa, en la sección V se abordan los principales componentes del habla interna, la sección VI presenta el habla interna en el cerebro y las principales zonas involucradas, en la sección VII se describen las ventajas y desventajas de la técnica de EEG para el problema de la predicción del habla interna, la sección VIII muestra el estado del arte con los resultados obtenidos de algunos trabajos basados en señales EEG para la predicción del habla interna, en la sección IX se describen dos conjuntos de datos para el reconocimiento del habla interna, en la sección X se presenta la metodología propuesta, la sección XI expone los resultados que se esperan obtener después de aplicada la metodología propuesta y finalmente en la sección XII se formulan las conclusiones.

2. Antecedentes

De acuerdo con el enfoque vygotskiano (Vygotsky 1896-1934), el pensamiento no se expresa simplemente en palabras, sino que existe a través de ellas, es decir, el lenguaje, el cual, si bien es adquirido en la interacción social, se interioriza en el curso del desarrollo, primero transformándose en un habla privada, que sigue siendo audible pero ya no dirigida a otros, y finalmente convirtiéndose en (inaudible) habla interna [6, 7].

3. Definiciones

Vygotsky (1896-1934) define el habla interna como la reutilización controlada e interna de una capacidad que se manifiesta primero en forma de discurso abierto. Oppenheim y Dell (2010) proponen que el habla interna "se caracteriza típicamente como la activación de representaciones lingüísticas abstractas o una simulación articulatoria detallada a la que sólo le falta la producción de sonido". Alderson-Day y Fernyhough (2015) proponen que el habla interna puede definirse como "la experiencia subjetiva del lenguaje en ausencia de la articulación manifiesta y audible". Grandchamp et al. (2019) también respaldan esta definición, ajustándola ligeramente a: "la experiencia subjetiva de la verbalización en ausencia de articulación o signo manifiesto". Peter Langland-Hassan (2020) define al habla interna como un tipo de uso mental del lenguaje, o simulación del habla, que, con bastante frecuencia, ocurre conscientemente y en ausencia de cualquier articulación manifiesta [8]. Anna M. Borghi y Charles Fernyhough (2021) proponen que el habla interna es el resultado de la internalización de interacciones mediadas lingüísticamente que regulan la cognición y el comportamiento [7]. Nikola A. Kompa (2023) describe al habla interna como un episodio interno que involucra sustancialmente el sistema de producción del habla [9].

4. Relación entre habla interna y habla externa

El habla interna está fonéticamente "empobrecida" en comparación con el habla externa, ocurriendo en un nivel más "abstracto" de procesamiento del lenguaje (es decir, antes de que se seleccionen las características fonémicas completas correspondientes al mensaje deseado). Puede ser que haya un componente articulatorio separado del habla interna: una representación o simulación [8]. Se ha demostrado que el habla interna involucra áreas similares del cerebro, al igual que la producción del habla manifiesta o externa [7]. Debido a las transformaciones que acompañan a la internalización, el discurso privado es funcional y formalmente distinto de su precursor, el habla social, lo que requerirá el refinamiento de los modelos de procesamiento predictivo del habla interna que la conciben como un discurso social atenuado. La evidencia de la abstracción proviene de los hallazgos de que el habla interna puede ocurrir en formas más condensadas y menos ricas en características. Un ejemplo reciente que apoya la abstracción mostró una disociación entre las velocidades en la lectura en voz alta y la lectura silenciosa en participantes adultos.

Mientras que la velocidad de lectura en voz alta se correlacionaba con la velocidad de articulación, la lectura silenciosa se correlacionaba más fuertemente con el vocabulario y el conocimiento conceptual [10]. Del mismo modo que el habla externa implica movimientos articulatorios complejos de la boca, los labios, la lengua y la caja vocal, el habla interna también puede implicar imágenes motoras de esos movimientos; o, si no se trata de imágenes, puede implicar la generación de un plan motor necesario para producir tal enunciado. Podemos llamar a esto el componente articulatorio del habla interna. La influyente teoría de Levelt (1989) sobre la producción del habla señala que el hablante comienza con un "mensaje" a transmitir, donde el mensaje se realiza en una estructura conceptual no lingüística de algún tipo, y luego lleva a cabo etapas

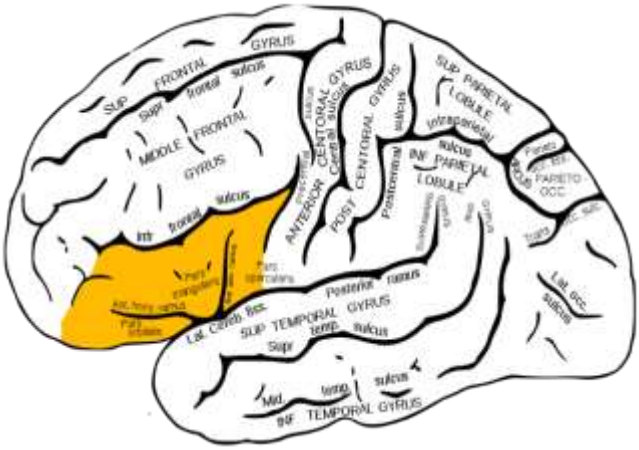

Fig. 1. Circunvolución frontal inferior izquierda (IIFGT**)** [13]**.**

sucesivas de procesamiento lingüístico con el fin de generar palabras de lenguaje natural adecuadas para expresar el mensaje en voz alta [11].

5. Componentes del habla interna

Dentro de la categoría del habla interna hay otra distinción importante entre hablar (donde la experiencia subjetiva es la de producir un enunciado, análogo a hablar en una grabadora de audio) y la audición interna (donde la experiencia es la de percibir una expresión que ha sido hablada internamente, como al escuchar la propia voz en una grabación) [10]. Al menos a menudo, nuestros episodios internos del habla ("ISE", por sus siglas en inglés) parecen tener un componente auditivo y debido a esto podemos decir que los ISE tienen asociado un carácter sensorial auditivo-fonológico [11]. Scott presentó evidencia de que la generación del habla interna atenúa la percepción de los sonidos externos, lo que es consistente con la opinión de que una predicción sensorial del enunciado en cuestión interfiere con la percepción de un sonido externo [12].

6. Habla interna en el cerebro

Las áreas de producción del lenguaje, como la circunvolución frontal inferior izquierda (lIFG, Figura. 1), son las que se activan de manera más significativa durante el habla interna, aunque, en algunas condiciones, hay una activación significativa de la percepción del lenguaje (como la circunvolución temporal superior (STG, Figura 2) [8].

Es posible que el paso del habla interna condensada a la expandida implique la traducción de tales representaciones en algo más expresado, a través de la articulación en el giro frontal inferior izquierdo también llamado circunvolución frontal inferior izquierda y la representación fonológica en las estructuras del giro temporal superior o también conocido como circunvolución temporal superior [12].

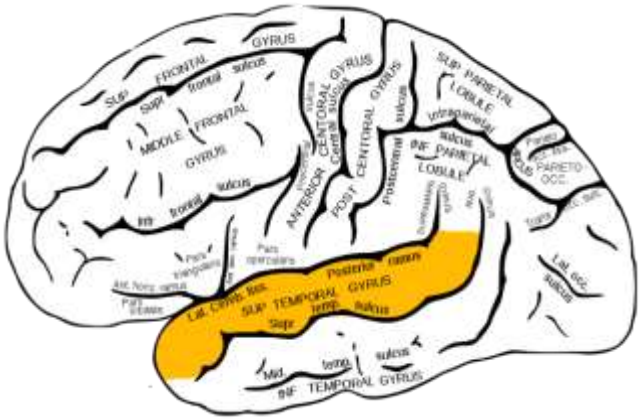

Fig. 2. Circunvolución temporal superior (STG) [13]**.**

7. Electroencefalografía

La electroencefalografía es una técnica de imagen médica que lee la actividad eléctrica del cuero cabelludo generada por las estructuras cerebrales. El electroencefalograma (EEG) se define como la actividad eléctrica de tipo alterno registrada en la superficie del cuero cabelludo después de ser captada por electrodos metálicos y medios conductores [2]. El EEG medido directamente desde la superficie cortical se denomina electrocortigrama, mientras que cuando se utilizan sondas de profundidad se denomina electrograma. Por lo tanto, la lectura electroencefalográfica es un procedimiento completamente no invasivo que se puede aplicar repetidamente a pacientes, adultos normales y niños, prácticamente sin riesgo ni limitación. Los patrones cerebrales forman formas de onda que suelen ser sinusoidales. Por lo general, se miden de pico a pico y normalmente oscilan entre 0,5 y 100 μV de amplitud. Por medio de la transformada de Fourier se deriva el espectro de potencia de la señal de EEG en bruto.

En el espectro de potencia, la contribución de las ondas sinusoidales con diferentes frecuencias es visible. Aunque el espectro es continuo, desde 0 Hz hasta la mitad de la frecuencia de muestreo, el estado cerebral del individuo puede hacer que ciertas frecuencias sean más dominantes. Las ondas cerebrales se han clasificado en cuatro grupos básicos [14]:

- Ondas Gamma (>35 Hz): Los cambios en la alta frecuencia gamma (70-150 Hz) se asocian con el habla abierta y encubierta. Durante el habla abierta el lóbulo temporal, el área de Broca, el área de Wernicke, la corteza premotora y la corteza motora primaria presentan cambios gamma elevados. Por otro lado, este estudio también presenta evidencia de altos cambios gamma durante el habla encubierta en la circunvolución supramarginal y el lóbulo temporal superior.

Tabla 1. Estado del arte.

Autor, Año	**Características y métodos**	**Resultados**
Moctezuma et al. (2017) [15]	Clasificación binaria entre actividad e inactividad lingüística imaginada utilizando dos conjuntos de datos, el primero que contiene las palabras en español: arriba, abajo, izquierda, derecha y seleccionar. Mientras que el segundo contiene las palabras en español: arriba, abajo, izquierda y derecha. Métodos: Bosques aleatorios	-92% de exactitud
Balaji et al. (2017) [16]	Clasificación multiclase de las palabras imaginadas "yes","no", "haan"y "na" en el idioma hindi e inglés como primer y segundo lenguaje respectivamente. Métodos: Máquinas de soporte vectorial Árboles aleatorios AdaBoost Redes neuronales artificiales	-92.18% de exactitud
Sereshkeh et al. (2017) [17]	Clasificación binaria de las palabras imaginadas "yes" y "no". Métodos: Máquina de soporte vectorial	-69.3% de exactitud
Alsaleh et al. (2018) [18]	Clasificación multiclase de las palabras imaginadas en español: arriba, abajo, izquierda, derecha y seleccionar. Métodos: Máquina de soporte vectorial Naive Bayes Bosques aleatorios Análisis discriminante lineal	-87.4% de exactitud
D. Vorontsova et al. (2021) [4]	Clasificación multiclase de 8 palabras imaginadas en el idioma ruso que expresan los comandos: forward, backward, up, down, help, take, stop y reléase. Métodos: Redes neuronales convolucionales Redes neuronales recurrentes	-84.5% de exactitud
Varshney and Khan (2022) [19]	Clasificación multiclase de seis palabras imaginadas: could, yard, give, him, there y toe. Métodos: Bosques aleatorios Máquinas de soporte vectorial	-28.61% de exactitud
Agarwal and Kumar (2022) [20]	Clasificación multiclase de cuatro palabras y una frase imaginadas, las cuales son, "sos", "stop", "medicine", "washroom"y "come-here" respectivamente. Métodos: Redes neurales recurrentes de tipo memoria a corto y largo plazo	-73.56% de exactitud

	Redes neuronales convolucionales Unidades recurrentes cerradas	
Einizade et al. (2022) [21]	Clasificación multiclase de tres palabras imaginadas las cuales son “piedra”, “papel” o “tijeras” y el estado de reposo. Métodos: Clasificadores basados en grafos	-50.10% de exactitud
Liwicki et al. (2022) [22]	Clasificación multiclase de dos grupos de palabras imaginadas en español, el primer grupo se compone de las vocales y el segundo grupo de seis palabras que expresan comandos: arriba, abajo, izquierda, derecha, atrás y adelante. Acceso abierto. Métodos: Redes neurales recurrentes de tipo memoria a corto y largo plazo Redes neuronales convolucionales Unidades recurrentes cerradas	-35.20% de exactitud para las vocales -29.21% de exactitud para las palabras que expresan comandos

- Ondas Beta (12–35 Hz): Estas ondas a menudo están relacionadas con el movimiento muscular y la retroalimentación. Por lo tanto, se puede considerar que están involucrados durante las tareas auditivas y la producción del habla.
- Ondas Alfa (8–12 Hz): Durante el procesamiento del lenguaje, estas ondas están involucradas en la retroalimentación auditiva y la percepción del habla. Además, la frecuencia alfa durante el habla encubierta se ha identificado como débil en comparación con su comportamiento durante el discurso abierto.
- Ondas Theta (4–8 Hz): Estas ondas se activan durante la restauración fonémica y el procesamiento de las señales de coarticulación para componer palabras. Además, otro estudio identificó que las ondas theta pueden ayudar a identificar las consonantes en las sílabas.
- Ondas Delta (0.5–4 Hz): Se ha encontrado que la entonación y el ritmo durante la percepción del habla caen en rangos de frecuencia que pertenecen a la banda de oscilación delta inferior. Asimismo, diversos estudios han encontrado otros procesos del habla en los que intervienen las ondas delta, como el fraseo prosódico, la estructura de sílabas, las sílabas largas, entre otros [2].

8. Estado del arte

En la Tabla 1 se muestra el estado del arte, el cual está organizado por año y autor, características de los conjuntos de datos y los modelos utilizados para cada caso, y finalmente los resultados por porcentaje de exactitud.

Tabla 2. Características del primer conjunto de datos.

Características	
Palabras	**Arriba, abajo, izquierda, derecha y seleccionar**
Sujetos	27
Número de ensayos por palabra	33
Técnica	EEG
Canales	14
Frecuencia de muestreo	128 Hz
Número de registros	1 569 207

9. Conjuntos de datos para el reconocimiento del habla interna

Existen una serie de conjuntos de datos para el reconocimiento del habla interna, sin embargo, actualmente se cuenta con escasos conjuntos de acceso abierto que puedan ser utilizados para el análisis de dicho problema y el mejoramiento de los modelos. Entre estos, se encuentran los siguientes dos conjuntos.

Conjunto 1: Este trabajo tiene como objetivo interpretar las señales de EEG registradas durante la pronunciación imaginada de palabras de un vocabulario reducido, sin emitir sonidos ni articular movimientos (habla interna o no pronunciada) con la intención de controlar un dispositivo. Específicamente, el vocabulario permitiría controlar el cursor de la computadora, y consta de las palabras del lenguaje español: “arriba”, “abajo”, “izquierda”, “derecha”, y “seleccionar”. Para ello, se registraron las señales de EEG de 27 individuos utilizando un protocolo básico para saber a priori en qué segmentos de la señal la persona imagina la pronunciación de la palabra indicada.

Para la adquisición de la actividad cerebral se registraron las señales EEG utilizando un kit EMOTIV. Este kit es inalámbrico y consta de catorce electrodos (canales) de alta resolución (más las referencias CMS/DRL en las posiciones P3/P4 respectivamente) cuya frecuencia de muestreo es de 128 Hz. Los nombres de los canales, de acuerdo con el sistema internacional 10- 20, son: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4 (ver Figura 3).

El protocolo consiste en colocar a la persona cómodamente sentada con los ojos abiertos cerca de un escritorio, y con la mano derecha sobre el mouse de una computadora. Con un clic al mouse, el usuario delimita tanto el inicio como el fin de la pronunciación imaginada de alguna de las cinco palabras del vocabulario. La pronunciación imaginada de cada una de las cinco palabras fue repetida 33 veces consecutivas durante el registro del EEG, es decir, cinco bloques de 33 repeticiones por palabra. Antes de cada bloque se le indicó al individuo cuál es la palabra que debía pronunciar internamente. Asimismo, todas las épocas de las cinco palabras

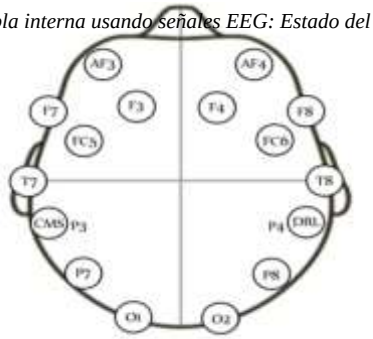

Fig. 3. Localización de los electrodos en el kit Emotiv [23].

pertenecientes a un individuo fueron registradas en una sola sesión. Además, al inicio del registro de las señales de EEG, se le indicó a la persona que evitara parpadear o realizar movimientos corporales mientras imaginaba la pronunciación de la palabra indicada, ya que después de cada marcador de fin podía tomarse un tiempo de descanso para dichos movimientos.

Con la finalidad de que el individuo no supiera cuantas veces se repetiría una palabra, en la sala de experimentos otra persona, alejada del campo visual y guardando el debido silencio, se encargó de realizar el conteo de repeticiones e indicó al individuo cuándo debía concluir. Esto con la finalidad de que el individuo no se distrajera contando el número de repeticiones ni se predispusiera a saber que le faltaba poco o mucho para concluir el experimento. Las sesiones se registraron en un laboratorio alejado de ruido audible externo, ruido visual (como la transición del día y la noche, y distracciones), entre otras. Para formar un corpus de datos para los experimentos, se registraron las señales de EEG de 27 individuos sanos (S1-S27), de los cuales 2 de ellos son zurdos y el resto son diestros. Sin embargo, de acuerdo con el modelo Geschwind-Wernicke, las señales de EEG relacionadas con la producción del habla afectan diferentes áreas en la parte izquierda del cerebro (a excepción de algunas personas zurdas) [23]. Las características dichas anteriormente, así como el número de registros que contiene el conjunto se pueden apreciar más claramente en la Tabla 2.

Conjunto 2: El objetivo principal de este trabajo es proporcionar a la comunidad científica una base de datos de electroencefalografía multiclase de acceso abierto de comandos de voz interna que podría utilizarse para una mejor comprensión de los mecanismos cerebrales relacionados. Cada participante realizó entre 475 y 570 ensayos en un solo día de registro, obteniendo un conjunto de datos con más de 9 horas de registro continuo de datos de EEG, que resultaron siendo más de 5600 ensayos en total.

El protocolo experimental fue aprobado por el Comité Asesor de Ética y Seguridad en el Trabajo Experimental (CEySTE, CCT-CONICET, Santa Fe, Argentina). Diez participantes diestros sanos, cuatro mujeres y seis hombres con edad media de 34 años, sin pérdida de audición, sin pérdida del habla y sin trastornos neurológicos, del movimiento o psiquiátricos, se unieron al experimento y dieron su consentimiento informado por escrito. Todos los participantes eran hablantes nativos de español.

Ninguno de los individuos tenía experiencia previa en BCI y participaron en aproximadamente dos horas de grabación. El estudio se llevó a cabo en una habitación

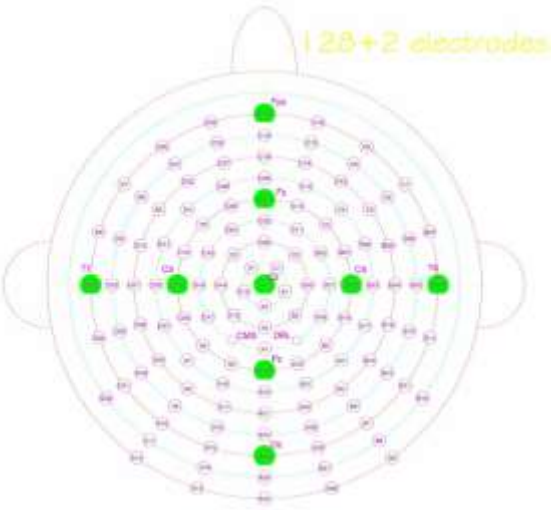

Fig. 4. Distribución de los electrodos en el casco BioSemi cap_128_layout_medium.jpg (970×900) (biosemi.com)[24].

con blindaje eléctrico. Los participantes se sentaron en una silla cómoda frente a una pantalla de computadora donde se presentaron las señales visuales. Con el fin de familiarizar al participante con el procedimiento experimental y el ambiente de la sala, se explicaron todos los pasos del experimento, mientras se colocaba el casco del EEG y los electrodos externos. El proceso de configuración duró aproximadamente 45 minutos. En la Figura 4 se muestra la configuración principal del experimento.

El fondo de la pantalla era de color gris claro para evitar el deslumbramiento y la fatiga ocular. Cada individuo participó en un solo día de grabación que comprendió tres sesiones consecutivas. Se dio un período de descanso autoseleccionado entre sesiones, para evitar el aburrimiento y la fatiga. Al comienzo de cada sesión, se registró una línea de base de segundos en la que se indicaba al participante que se relajara y permaneciera lo más quieto posible. Dentro de cada sesión, se presentaron cinco corridas de estimulación. Las ejecuciones corresponden a las diferentes condiciones propuestas: habla pronunciada, habla interna y condición visualizada. Al comienzo de cada corrida, la condición se anunciaba en la pantalla de la computadora durante un período de 3 segundos. En todos los casos, el orden de las corridas fue: un discurso pronunciado, dos discursos internos y dos condiciones visualizadas. Se dio un descanso de un minuto entre corridas. Las clases se seleccionaron específicamente considerando una aplicación de control BCI natural con las palabras en español: "arriba", "abajo", "derecha", "izquierda".

La clase (palabra) del ensayo se presentó al azar. Cada participante tuvo 200 ensayos tanto en la primera como en la segunda sesión. Sin embargo, dependiendo de la voluntad y el cansancio, no todos los participantes realizaron el mismo número de ensayos en la tercera sesión. Para evaluar la atención de cada participante, se agregó aleatoriamente un control de concentración al habla interna y se ejecutó la condición visualizada. La tarea de control consistió en preguntar al participante, después de algunos ensayos seleccionados al azar, cuál era la dirección de la última clase mostrada.

El participante tenía que seleccionar la dirección usando las flechas del teclado. No se dio un límite de tiempo para responder a estas preguntas y el protocolo continuó después de que el participante presionara cualquiera de las cuatro teclas de flecha. Se proporcionó información visual que indicaba si la pregunta se había respondido correcta o incorrectamente.

Los datos de electroencefalografía (EEG), electrooculografía (EOG) y electromiografía (EMG) se adquirieron utilizando un sistema de medición de alta resolución BioSemi ActiveTwo. Para la adquisición de datos, se utilizaron 128 canales activos de EEG y 8 canales externos activos de EOG/EMG con una resolución de 24 bits y una frecuencia de muestreo de 1024 Hz. BioSemi también proporciona gorros de cabeza de EEG estándar de diferentes tamaños con posiciones de electrodos pre-fxed. Se eligió un gorro de tamaño adecuado para cada participante midiendo la circunferencia de la cabeza con una cinta métrica.

Con el objetivo de localizar las principales fuentes de activación y analizar sus conexiones, pedimos a los participantes que realizaran el experimento bajo tres condiciones diferentes: habla interna, habla pronunciada y condición visualizada:

- *Habla interna.* El habla interna es la condición principal en el conjunto de datos y tiene como objetivo detectar la actividad eléctrica del cerebro relacionada con el pensamiento de un participante sobre una palabra en particular. En las ejecuciones de discurso interno, se indicó a cada participante que imaginara su propia voz como si estuviera dando una orden directa a la computadora, repitiendo la palabra correspondiente hasta que el círculo blanco se volviera azul. A cada participante se le pidió explícitamente que no se centrara en los gestos de articulación. Además, a cada participante se le indicó que permaneciera lo más quieto posible y que no moviera la boca ni la lengua. En aras de la imaginación natural, no se proporcionó ninguna señal rítmica.

- *Habla pronunciada.* La condición pronunciada del habla se propuso con el propósito de encontrar regiones motoras involucradas en la pronunciación que coincidan con las activadas durante la condición interna del habla. En las ejecuciones de discurso pronunciado, se indicó a cada participante que pronunciara repetidamente en voz alta la palabra correspondiente a cada señal visual, como si estuviera dando una orden directa a la computadora. Al igual que en las ejecuciones de voz internas, no se proporcionó ninguna señal rítmica.

- *Condición visualizada.* Dado que las palabras seleccionadas tienen un alto componente visual y espacial, y con el objetivo de encontrar cualquier actividad relacionada que se produce durante el habla interior, se propuso la condición visualizada. Es oportuno mencionar que los principales centros neuronales relacionados con el pensamiento espacial se localizan en las regiones occipital y parietal. En las corridas de condiciones visualizadas, se indicó a los participantes que se concentraran en mover mentalmente el círculo que aparecía en el centro de la pantalla en la dirección indicada por la señal visual [24].

Las características dichas anteriormente, así como el número de registros que contiene el conjunto se pueden apreciar más claramente en la Tabla 3.

Tabla 3. Características del segundo conjunto de datos.

Características	
Palabras	**Arriba, abajo, izquierda y derecha**
Sujetos	10
Número de ensayos por palabra	Habla externa: 25-30 Habla interna: 50-60
Técnica	EEG
Canales	128
Frecuencia de muestreo	1024 Hz
Número de registros	47 775 570

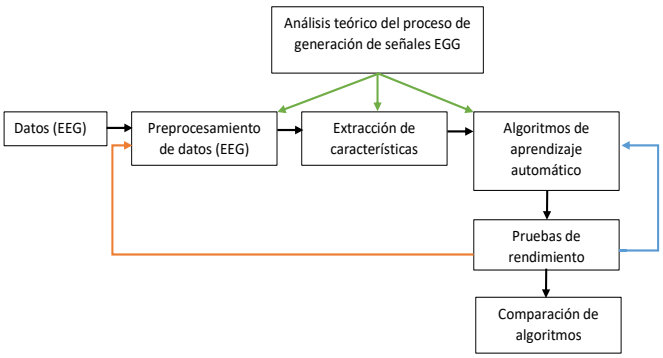

Fig. 5. Diagrama de la metodología propuesta.

10. Metodología propuesta

La metodología que se propone en este trabajo y con base en la información y conjuntos de datos descritos en las secciones anteriores, es la siguiente:

Como se aprecia en el diagrama de la Figura 5, la propuesta comienza con la adquisición de los datos, posteriormente se realiza una investigación teórica sobre el proceso de generación de las señales EEG, y para este caso particular en qué consiste el habla interna, así como las zonas del cerebro involucradas y las características y peculiaridades que la distinguen. Lo anterior, con el objetivo de conocer sus principales propiedades y tomar decisiones informadas para la selección del modelo de clasificación (flechas verdes →). El paso siguiente es la realización del preprocesamiento de los datos recaudados, así como la extracción de características, que bien puede ser mediante medidas estadísticas [15] aplicando además un método que por medio de raíces aplicadas a éstas medidas se saca una relación entre todas ellas a partir de la sustitución consecutiva del índice y el radicando por cada uno de los

valores de las medidas estadísticas [25] u otros métodos afines para luego aplicar clasificadores basados en algoritmos de aprendizaje automático como redes neuronales recurrentes de tipo memoria a corto y largo plazo (LSTM, por sus siglas en inglés) o técnicas de aprendizaje con grafos. Posteriormente se propone realizar pruebas de rendimiento para comparar los resultados con trabajos anteriores, de forma que, si el porcentaje no supera a los anteriores o no se obtiene un resultado relevante en ningún aspecto se vuelva a la etapa de algoritmos de aprendizaje automático (representado por la flecha azul →) o en su peor defecto al preprocesamiento de los datos (representado por la flecha naranja →).

Existe también una segunda alternativa de comparación que consiste en utilizar Transformers [26][27] específicamente un transformador generativo pre-entrenado para series de tiempo, mejor conocido como TimeGPT por sus siglas en inglés, que es el primer modelo base para series de tiempo, capaz de generar predicciones precisas para diversos conjuntos de datos sin necesidad de entrenamiento adicional. Lo anterior, gracias a que TimeGPT se entrenó con la mayor colección de series de tiempo disponibles públicamente (alrededor de 100 billones de datos) y puede pronosticar series de tiempo no vistas sin volver a entrenar sus parámetros. El entrenamiento de TimeGPT, con este rico conjunto de datos lo equipa para manejar una amplia gama de escenarios, lo que mejora su solidez y capacidades de generalización. Además de las características temporales, el conjunto de datos muestra variabilidad en los niveles de ruido y la presencia de valores atípicos, lo que proporciona una condición de entrenamiento sólida.

Esto es importante porque algunos conjuntos de datos exhiben patrones ordenados y predecibles, mientras que otros exhiben picos de ruido significativos o eventos inesperados, lo que proporciona una amplia gama de escenarios para la asimilación de modelos lo que permite a TimeGPT pronosticar con precisión series temporales no vistas, al tiempo que elimina la necesidad de entrenamiento y optimización de modelos individuales. TimeGPT se beneficia del entrenamiento previo en conjuntos de datos de series temporales masivos y diversos, incluso en escenarios con datos escasos, puede aprovechar este conocimiento previo para tener un buen rendimiento. En contraste, los modelos tradicionales de aprendizaje automático tienen dificultades para capturar patrones complejos debido a la insuficiencia de datos de entrenamiento, lo que resulta en una menor precisión de pronóstico. Con este nuevo modelo se hace innecesaria la implementación del preprocesamiento a los datos, la optimización de híper parámetros y la selección de modelos [26, 27].

11. Resultados esperados

Los resultados que se esperan obtener dada la metodología propuesta, son, en primera instancia, una métrica de exactitud que supere la obtenida en trabajos anteriores relacionados o al menos los que se sirven de los mismos conjuntos de datos citados en la sección IX, así como también sustraer conocimientos significativos acerca de las características más primordiales y fundamentales de las señales del habla interna para el caso particular de un vocabulario reducido de palabras que expresan comandos, sean estas características propiedades de las señales como frecuencia, amplitud, formas

basadas en parámetros, caracterización de patrones, etc.. De igual manera llegar a conclusiones importantes sobre los modelos aplicables a problemas de series de tiempo basados en señales electroencefalográficas para la clasificación del habla interna o incluso para construir una hipótesis bien fundamentada para el reconocimiento del habla interna fluida o el discurso interno.

12. Conclusiones

Las señales electroencefalográficas poseen características útiles para el análisis del habla interna que pueden aportar información relevante a cerca de las propiedades que distinguen al habla interna del habla externa, debido a esto los modelos de procesamiento predictivo del habla interna que la conciben como un discurso social atenuado requerirán su refinamiento. Por lo que se espera que la presente metodología propuesta aporte información relevante a tal fin, con la que sea posible la clasificación más precisa de un vocabulario reducido de palabras que expresan comandos, así como sentar algunas bases hipotéticas para el reconocimiento del habla interna fluida a partir de señales electroencefalográficas.

Agradecimientos. El trabajo se realizó con el apoyo de la Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional mediante los proyectos SIP 20240951 y 20240950. Además, los autores agradecen al CONAHCYT por su apoyo.

Referencias

1. Lee, Y.E., Lee, S. H., Kim, S. H.: Towards Voice Reconstruction from EEG During Imagined Speech. In: Proc. 37th AAAI Conf. Artif. Intell., 37, pp. 6030–6038 (2023) doi: 10.1609/aaai.v37i5.25745.
2. Lopez-Bernal, D., Balderas, D., Ponce, P.: A State-of-the-Art Review of EEG-Based Imagined Speech Decoding. Front. Hum. Neurosci., 16, pp. 1–14 (2022) doi: 10.3389/fnhum.2022.867281.
3. Gu X., *et al.*: EEG-Based Brain-Computer Interfaces (BCIs): A Survey of Recent Studies on Signal Sensing Technologies and Computational Intelligence Approaches and Their Applications. In: IEEE/ACM Trans. Comput. Biol. Bioinforma., 18(5), pp. 1645–1666 (2021) doi: 10.1109/TCBB.2021.3052811.
4. Vorontsova, D., *et al.*: Silent Eeg-Speech Recognition Using Convolutional And Recurrent Neural Network With 85% Accuracy Of 9 Words Classification. Sensors, 21(20), pp. 1–19 (2021) doi: 10.3390/s21206744.
5. Nurhadi, J., Rahma, R., Rahayu, W.: Mechanism of Inner Speech in Silent Reading Activity. In: Proc. Fifth Int. Conf. Lang. Lit. Cult. Educ. (ICOLLITE), 595, pp. 379–385 (2022) doi: 10.2991/assehr.k.211119.059.
6. L. S. Vigotsky, Pensamiento y lenguaje. México: Ediciones Quinto Sol (1988)
7. Kompa, N. A.: Inner Speech and ‘Pure’ Thought – Do we Think in Language?. Rev. Philos. Psychol (2023) doi: 10.1007/s13164-023-00678-w.
8. Langland-Hassan, P.: Inner speech. Wiley Interdiscip. Rev. Cogn. Sci., 12(2), pp. 1–18 (2021) doi: 10.1002/wcs.1544.
9. Borghi A. M., Fernyhough, C.: Concepts, abstractness and inner speech. Philos. Trans. R. Soc. B Biol. Sci., 378(1870) (2023) doi: 10.1098/rstb.2021.0371.

10. Fernyhough, C., Borghi, A. M.: Inner speech as language process and cognitive tool. Trends Cogn. Sci., 27 (12), pp. 1180–1193 (2023) doi: 10.1016/j.tics.2023.08.014.
11. Gregory, D.: Inner Speech: New Voices. Anal. (United Kingdom), 80(1), pp. 164–173 (2020) doi: 10.1093/ANALYS/ANZ096.
12. Fernyhough C., Alderson-Day, B.: Inner Speech: Development, Cognitive Functions, Phenomenology, and Neurobiology. Psychol. Bull., 141(5), pp. 931–965 (2015)
13. Wikimedia: Wikimedia Commons (2020) url: https://commons.wikimedia.org/wiki/MainPage.
14. Teplan, M.: xxFundamentals of EEG Measurement. Meas. Sci. Rev., 2, pp. 1–11 (2002)
15. Alfredo-Moctezuma, L., Carrillo, M., Villaseñor Pineda, L.: Hacia la clasificación de actividad e inactividad lingüística a partir de señales de electroencefalogramas (EEG). Res. Comput. Sci., 140 (1), pp. 135–149 (2017) doi: 10.13053/rcs-140-1-11.
16. Balaji A., *et al.*: EEG-based Classification of Bilingual Unspoken Speech Using ANN. In: Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS, pp. 1022–1025 (2017) doi: 10.1109/EMBC.2017.8037000.
17. Sereshkeh, A.R., Trott, R., Bricout, A.: Online EEG Classification of Covert Speech for Brain-Computer Interfacing. Int. J. Neural Syst., 27(8), pp. 1–16 (2017) doi: 10.1142/S0129065717500332.
18. Alsaleh, M., Moore, R., Christensen, H.: Examining Temporal Variations in Recognizing Unspoken Words Using EEG Signals. In: Proc. IEEE Int. Conf. Syst. Man, Cybern. SMC 2018, pp. 976–981 (2018) doi: 10.1109/SMC.2018.00173.
19. Varshney Y.V., Khan, A.: Imagined Speech Classification Using Six Phonetically Distributed Words. Front. Signal Process., 2, pp. 1–11 (2022) doi: 10.3389/frsip.2022.760643.
20. Agarwal P., Kumar, S.: Electroencephalography-Based Imagined Speech Recognition Using Deep Long Short-Term Memory Network. ETRI J. 44(4), pp. 672–685 (2022) doi: 10.4218/etrij.2021-0118.
21. Einizade, A., Mozafari, M., Jalilpour, S.: Neural Decoding of Imagined Speech From EEG Signals Using the Fusion of Graph Signal Processing and Graph Learning Techniques. Neurosci. Informatics, 2(3), pp. 100091 (2022) doi: 10.1016/j.neuri.2022.100091.
22. Simistira-Liwicki, F., Gupta, V., Saini, R.: Rethinking the Methods and Algorithms for Inner Speech Decoding and Making Them Reproducible. NeuroSci, 3(2), pp. 226–244 (2022) doi: 10.3390/neurosci3020017.
23. Torres, A., Reyes, C.A., Villaseñor, L.: Análisis de Señales Electroencefalográficas para la Clasificación de Habla Imaginada. Rev. Mex. Ing. Biomédica, 34(1), pp. 23–39 (2013)
24. Nieto, N., Peterson, V., Rufiner, H. L.: Thinking Out Loud, an Open-Access EEG-Based BCI Dataset for Inner Speech Recognition. Sci. Data, 9(1) (2022) doi: 10.1038/s41597-022-01147-2.
25. Cerón Figueroa, S.: Classificador asociativo basado en conceptos lineales y no lineales. Instituto Politécnico Nacional, CIC (2018) https://tesis.ipn.mx/jspui/bitstream/123456789/26705/1/T1977.pdf.
26. Garza, A., Mergenthaler-Canseco, M.: TimeGPT-1. pp. 1–12, (2023) http://arxiv.org/abs/2310.03589.
27. Liao, W., et al.: TimeGPT in Load Forecasting: A Large Time Series Model Perspective. (2024) http://arxiv.org/abs/2404.04885.

ISSN 1870-4069

Árbol de decisiones en prototipo de sistema de gestión de riesgos para proyectos de software

Axel Omar Alarcón Padilla, Jessie Paulina Guzmán Flores, Juan Jesús Gutiérrez García

Instituto Politécnico Nacional,
Escuela Superior de Cómputo,
México

aalarconp1600@alumno.ipn.mx, jguzmanf@ipn.mx, jgutierrezg@ipn.mx

Resumen. El prototipo se centra en desarrollar una aplicación web que utiliza técnicas de machine learning para la gestión de riesgos en proyectos de software, con un enfoque en la seguridad de la información. Se basa en árboles de decisiones para evaluar y mitigar riesgos, y cuenta con un módulo de plan de seguridad para asignar acciones específicas en respuesta a los riesgos identificados. Además, el prototipo incluye la capacidad de realizar un inventario de activos que clasifica los riesgos según la amenaza, impacto, probabilidad y vulnerabilidad. También implementa un algoritmo para priorizar los riesgos y proporciona informes detallados sobre los riesgos detectados, los planes de seguridad y sus actualizaciones. Adicionalmente, se ofrece una bitácora de seguimiento para registrar los análisis de riesgos y el progreso de los planes de seguridad. Este enfoque integra los estándares del Project Management Institute (PMI) y OWASP, utilizando métricas de análisis de vulnerabilidades, amenazas, probabilidad e impacto para proyectos web, generando una solución integral para identificar, evaluar y priorizar riesgos en función de su probabilidad e impacto.

Palabras clave: Gestión de riesgos, seguridad de la información, machine learning.

Decision Tree in a Risk Management System Prototype for Software Projects

Abstract. The prototype focuses on developing a web application that utilizes machine learning techniques for risk management in software projects, with a particular emphasis on information security. It is based on decision trees to evaluate and mitigate risks and includes a security plan module to assign specific actions in response to identified risks. Additionally, the prototype has the capability to perform an asset inventory that classifies risks according to threat, impact, probability, and vulnerability. It also implements an algorithm to prioritize risks and provides detailed reports on detected risks, security plans, and their updates. Furthermore, a tracking log is available to record risk analyses and

the progress of security plans. This approach integrates the standards of the Project Management Institute (PMI) and OWASP, using metrics for analysing vulnerabilities, threats, probability, and impact for web projects, generating a comprehensive solution to identify, evaluate, and prioritize risks based on their probability and impact.

Keywords: Risk management, information security, machine learning.

1. Introducción

Un software de análisis de riesgos en seguridad de la información es una herramienta esencial para el plan de acción y mantenimiento de cualquier proyecto de software que busque proteger la información, la tecnología y la comunicación, lo cual conlleva a cumplir sus objetivos y asegurar la información de sus usuarios.

A lo largo de la vida escolar, se han realizado proyectos de software, en los cuales se ha observado repetidamente la necesidad de una herramienta robusta de análisis de riesgos en seguridad de la información. Cada proyecto enfrentó desafíos relacionados con la protección de la información y la gestión de amenazas potenciales, lo cual resaltó la importancia de contar con un sistema que facilite la identificación y mitigación de riesgos.

En la experiencia acumulada durante estos proyectos, se evidenció que las soluciones manuales son laboriosas, lo que puede comprometer una correcta evaluación y clasificación de los riesgos. Por lo tanto, se hace imperativo el uso de un software especializado que apoye a los alumnos en estos procesos durante el ciclo de vida de desarrollo de sus proyectos.

La implementación de la herramienta desarrollada se probó con un proyecto de investigación que involucraba una aplicación móvil destinada a apoyar la lectura y comprensión lectora [1], con el fin de garantizar la seguridad de la información dentro del desarrollo de este proyecto. Esta prueba permitió validar la efectividad del software en un entorno real, evaluando su capacidad para identificar, clasificar y gestionar riesgos en las diferentes fases del ciclo de vida del proyecto.

La implementación de un sistema de seguridad de la información requiere el uso de software especializado. Este software utiliza una variedad de técnicas y métodos para identificar, clasificar y evaluar los riesgos asociados a la seguridad de la información en una organización. Este tipo de herramientas es esencial para garantizar la integridad, la confidencialidad y la disponibilidad de los datos almacenados y procesados por los sistemas informáticos. Con su uso, las organizaciones pueden tomar medidas preventivas y correctivas para mitigar los riesgos y proteger su información e infraestructura.

El problema que abordaremos a continuación es la forma en la que los equipos de desarrollo de proyectos realizan su gestión de riesgos, pues cada parte se realiza a mano, lo cual es más tardado y susceptible al error humano, ya que las herramientas gratuitas que se encuentran en línea son plantillas que en el usuario introduce los datos, además de usar parámetros que no son claros y no se adaptan a las necesidades del gestor ni del proyecto.

Toda la información procesada y almacenada por un software está sujeta a amenazas físicas, lógicas y organizacionales. Dicha información puede representar datos importantes de clientes y usuarios, desde un nombre hasta una cuenta bancaria, considerada por una organización como un activo del negocio que tiene un valor, por lo que no se puede escatimar en su protección frente a amenazas. En el proceso de desarrollo de un sistema de software existen riesgos potenciales que amenazan la seguridad de la información, dicha información puede perder confidencialidad, integridad y disponibilidad [2].

El perder confidencialidad indica que la información pase a estar disponible o proporcionada a entidades o procesos no autorizados. El que la información pierda integridad hace que deje de ser precisa y completa, mientras que perder disponibilidad significa que deje de ser accesible y usable en petición de una entidad o proceso autorizados. [3] Un riesgo de seguridad de la información es un efecto de incertidumbre en los objetivos de la seguridad de la información. La causa de los riesgos es un incidente, el cual perjudica al ciclo de vida del desarrollo software en cualquier etapa, por lo que un equipo de desarrollo necesita tener un plan de acción frente a incidentes, compuesto por valoración, identificación, administración y tratamiento de riesgos. [3]

La necesidad de abordar eficazmente los riesgos en seguridad de la información es cada vez más apremiante en un entorno digital en constante evolución. A medida que las organizaciones dependen cada vez más de la tecnología y la interconexión, se vuelven más susceptibles a una variedad de amenazas cibernéticas.

1.1. Trabajos relacionados

La gestión de riesgos en el ámbito específico de los proyectos de software, en lo que respecta a la seguridad de la información, no se ha establecido un enfoque que integre los árboles de decisiones para la evaluación de riesgos. En la mayoría de los casos, las soluciones existentes se basan en conjuntos de reglas fijas y pocas de estas ocupan datos previamente recopilados por la herramienta. En su mayoría las herramientas disponibles no contemplan la aplicación de los estándares del Project Management Institute (PMI) y o del OWASP. Sin embargo, al revisar la literatura actual, se observa que muchos de los enfoques no están alineados claramente con estos estándares, ni detallan de manera específica el uso de árboles de decisiones como herramienta clave para la priorización y gestión de riesgos. Algunas de las herramientas analizadas son:

- Plantillas de matrices y gestión de riesgos en Excel [9]. Estas son las más accesibles y se encuentran en una breve búsqueda en internet, si bien estas son personalizables todas son a la experiencia que tenga el gestor a la hora de categorizar y evaluar un riesgo.
- Pirani [10]. Gestiona procesos, controles, eventos, planes de acción, indicadores y evaluaciones relacionados con el gobierno corporativo, no implementa los estándares del Project Management Institute (PMI) tampoco los de OWASP así como no deja en claro si utiliza arboles de decisiones.
- Aperisoft [11]. Imita cómo funciona el riesgo y la gestión de riesgos en el mundo real. Analiza los riesgos cuantitativamente con un motor de simulación Monte Carlo,

pero no utiliza arboles de decisiones para la evaluación de los riesgos y no implementa los estándares del Project Management Institute (PMI).

2. Módulos

Se ha concebido el desarrollo de un prototipo de aplicación Web de gestión de riesgos altamente especializada, con un enfoque primordial en proyectos de software y tomando como referencia las buenas prácticas de los marcos de trabajo. A diferencia de las alternativas disponibles en el mercado que carecen de su integración y se encuentran diseñadas para organizaciones con una infraestructura de sistemas robusta, el prototipo de aplicación Web está diseñado para ser una solución enfocada en proyectos.

Al centrarnos en la gestión de proyectos de software y aprovechar los principios de la guía de gestión de riesgos del Instituto de Gestión de Proyectos (Project Management Institute, PMI en sus siglas en inglés), en base al PMI se tomó en cuenta el proceso de gestión de riesgos en el ciclo de vida del proyecto. Este enfoque permite integrar la gestión de riesgos en todas las fases del proyecto.

Asimismo, se ha considerado la metodología de valoración de riesgos del Proyecto Abierto de Seguridad de Aplicaciones Web (Open Web Application Security Project, OWASP en sus siglas en inglés). Con el OWASP Risk Rating se utilizó para la evaluación de los riesgos, adaptándolo con un enfoque de inventario de activos. Esta adaptación implica identificar y clasificar los activos del proyecto, lo que nos ayuda a evaluar los riesgos en función de su impacto y probabilidad, nuestro prototipo de aplicación Web brinda una metodología y una estructura para realizar la evaluación de riesgos, lo que nos ayuda a crear planes para mitigar y controlar los riesgos.

El sistema ofrece herramientas para evaluar la probabilidad y el impacto de los riesgos identificados, permitiendo a los usuarios asignar valores numéricos a estos parámetros para calcular la prioridad de los riesgos y establecer estrategias de mitigación.

Mantiene un registro completo de todos los riesgos identificados, incluyendo su descripción y nivel de prioridad.

Los módulos del sistema son los siguientes:

– Módulo de gestión de proyectos y usuarios: En este módulo se manejarán los proyectos de software y las sesiones de los usuarios gestores y participantes, así como el alta de estos en el sistema. El usuario gestor tiene el acceso completo del sistema, junto con todos sus módulos y la capacidad de añadir distintos usuarios participantes que se encarguen de las acciones del módulo del plan de seguridad. Además, se centra en proporcionar a estos usuarios participantes la capacidad de acceder a su sesión personalizada para reportar el progreso de las acciones asignadas en todos los proyectos en los que participan. Los usuarios participantes, que pueden ser supervisores de área, jefes de área, técnicos de área, u otros roles responsables de mantener la integridad y el orden de los reportes de tareas dentro de la organización, tendrán la responsabilidad de reportar el porcentaje de avance de los

planes de acción asignados y proporcionar información detallada sobre dicho avance para el usuario gestor.
- Módulo de inventario de activos: Tendrá formularios para dar de alta, baja o modificar cada activo que tenga el proyecto de software. La sección de activos ahora consta de dos partes: inventario y evaluación. En el inventario de activos el gestor puede visualizar a detalle y editar los activos del proyecto además de poder agregar otros, de acuerdo con las necesidades específicas del proyecto. Los datos de los activos en el inventario se componen de una ficha técnica sin métricas de valoración para el análisis de riesgos. Mientras que, en la sección de evaluación, se le dan valores a las variables de confidencialidad, disponibilidad e integridad de los activos del inventario, dando un valor de sensibilidad del activo.
- Módulo de gestión de riesgos: Este módulo tendrá desglosados cada riesgo identificado con sus activos asociados, y su amenaza, vulnerabilidad, probabilidad de ocurrencia e impacto. Además de la matriz de riesgos con un semáforo colorimétrico en el que se visualizará la valoración de cada riesgo calculada por la aplicación.

2.1. Procesos del módulo de gestión de riesgos

1. Identificación de amenazas: Una vez que se ha establecido un inventario de activos, se procede a identificar las amenazas potenciales que podrían afectar a estos activos. Las amenazas pueden manifestarse en forma de ataques de hackers, malware, ataques de denegación de servicio (DDoS) y otros peligros que podrían comprometer la seguridad de la aplicación web.

2. Evaluación de vulnerabilidades: Este paso implica la evaluación de las vulnerabilidades en la aplicación que podrían ser explotadas por las amenazas identificadas en el paso anterior. Estas vulnerabilidades pueden incluir problemas de seguridad en el código, configuraciones incorrectas en servidores, falta de control de acceso y otras debilidades potenciales.

3. Valoración de riesgos: Implica asignar valores a factores de confidencialidad, integridad y disponibilidad de los activos asociados. Además, se considera la probabilidad e impacto de que ocurra dicho riesgo. Este proceso permite entender la magnitud del riesgo asociado con cada activo [4]. Para calcular la probabilidad, se tienen cuatro factores de amenaza y cuatro de vulnerabilidad. Los factores de amenaza son: Nivel de habilidad, motivación, oportunidad y tamaño.

El valor de la amenaza del riesgo está dado por [4]:

$$\textbf{Amenaza} = \frac{\mathrm{N_H} + M + O + T}{8} \ . \tag{1}$$

donde:

N_H= Nivel de habilidad de la amenaza en un valor del 1 al 9.
M= Motivación de la amenaza en un valor del 1 al 9.
O= Oportunidad de la amenaza en un valor del 1 al 9.

T= Tamaño de la amenaza en un valor del 1 al 9.

Los factores de vulnerabilidad son: Facilidad de descubrimiento, facilidad de explotación, conciencia y detección de intrusiones. El valor de la vulnerabilidad del riesgo está dado por [4]:

$$\textbf{Vulnerabilidad} = \frac{F_D + F_E + C + D_I}{8} \ . \tag{2}$$

donde:

F_D= Facilidad de descubrimiento de la vulnerabilidad en un valor del 1 al 9.
F_E= Facilidad de explotación de la vulnerabilidad en un valor del 1 al 9.
C= Conciencia de la vulnerabilidad en un valor del 1 al 9.
D= Detección de intrusiones en la vulnerabilidad en un valor del 1 al 9.

Con dichos factores, el valor de la probabilidad del riesgo está dado por [4]:

$$\textbf{Probabilidad} = A + V \ . \tag{3}$$

donde:

A= Valor de amenaza derivado de sus factores
V= Valor de vulnerabilidad derivado de sus factores.

Para calcular el impacto, se tienen cuatro factores de impacto empresarial y tres de impacto técnico derivado de los activos asociados al riesgo. Los factores de impacto empresarial son: Daño financiero, daño a la reputación, incumplimiento y violación a la privacidad. El valor del impacto empresarial del riesgo está dado por [4]:

$$\boldsymbol{I_E} = \frac{D_F + D_R + N_C + V_P}{4} \ . \tag{4}$$

donde:

I_E= Valor del impacto empresarial del riesgo.
D_F= Daño financiero a la organización en un valor del 1 al 9.
D_R= Daño a la reputación de la organización en un valor del 1 al 9.
N_C= Incumplimiento de la organización en un valor del 1 al 9.
V_P= Violación de la privacidad de la organización en un valor del 1 al 9.

Los factores del impacto técnico son las variables de confidencialidad, disponibilidad e integridad del activo asociado al riesgo con mayor sensibilidad. El valor del impacto técnico está dado por [4]:

$$\boldsymbol{I_T} = \frac{C_O + D + I_N}{3} . \tag{5}$$

donde:

C_O= Confidencialidad del activo con la mayor sensibilidad de la lista de activos asociados al riesgo en un valor del 1 al 9.

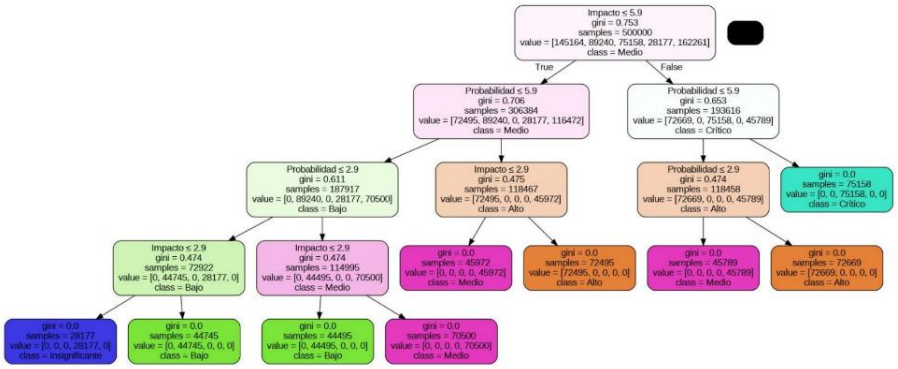

Fig. 1. Árbol de decisiones de la priorización del riesgo.

D= Disponibilidad del activo con la mayor sensibilidad de la lista de activos asociados al riesgo en un valor del 1 al 9.

I_N= Integridad del activo con la mayor sensibilidad de la lista de activos asociados al riesgo en un valor del 1 al 9.

Con dichos factores, el valor del impacto del riesgo está dado por [4]:

$$\textbf{Impacto} = \frac{I_E + I_T}{\mathbf{2}}. \tag{6}$$

4. Análisis del riesgo: Se calcula el riesgo para cada activo multiplicando el impacto y la probabilidad. Este cálculo proporciona una medida cuantitativa de la gravedad del riesgo.

5. Priorización de riesgos: Con la información sobre los riesgos calculados, se priorizan los activos en función de la magnitud de su riesgo. Los activos con riesgos más altos reciben una atención especial en términos de medidas del plan de seguridad. El diagrama de flujo de la priorización representa el árbol de decisiones que solo evalúa las variables de estimación de probabilidad e impacto del riesgo, como lo muestra la Figura 1.

La aplicación de técnicas de machine learning en la metodología de gestión de riesgos aporta una capacidad predictiva y adaptativa al prototipo, permitiendo anticipar posibles amenazas y tomar medidas preventivas de manera proactiva. Al analizar datos históricos y tendencias emergentes, el sistema puede identificar de forma automática escenarios de riesgo potenciales, brindando a los usuarios una herramienta avanzada para la toma de decisiones informadas en materia de seguridad de la información. Asimismo, la estructura de árboles de decisiones facilita la visualización y comprensión de las diferentes opciones de mitigación de riesgos, ofreciendo una guía clara y estructurada para la gestión integral de la seguridad en proyectos de software.

Al visualizar las diferentes herramientas de machine learning se observó que la que más se acomodaba a las necesidades del prototipo era el árbol de decisiones tipo CART

Tabla 1. Comparación de resultados de gestores de riesgo.

Herramienta utilizada	Riesgos detectados	Planes implementados	Riesgos reevaluados
Prototipo	57	81	44
Plantilla	38	63	26

ya que en la metodología de gestión de riesgos evalúa la importancia de diferentes variables y sus impactos en los resultados del proyecto. Esta evaluación precisa ayuda a priorizar los riesgos más críticos y tomar decisiones informadas sobre cómo abordarlos. Esto es una ayuda para gestores de riesgo que no tienen experiencia en el análisis y gestión de riesgos ya que mucha de la dificultad radica en que no han realizado estas actividades con de una forma estructurada. Además, que capacidad predictiva y la estructura jerárquica de decisiones de los árboles de decisiones de este tipo ofrecen anticipar posibles riesgos, esto es especialmente útil en la fase de planificación del proyecto, ya que se pueden tomar medidas preventivas desde ese punto del ciclo de vida del proyecto.

3. Resultados y trabajo a futuro

Se plantearon una serie de pruebas a lo largo del desarrollo e implementación de un proyecto de software, se contrastarían los riesgos detectados, los planes de mitigación implementados y los riesgos reevaluados de la Aplicación móvil para el apoyo a la lectura y comprensión lectora [1] con el prototipo propuesto y la herramienta al alcance del equipo de desarrollo que eran las plantillas de matriz y gestión de riegos en Excel [9] ya que esta aplicación forma parte de un proyecto de investigación y el utilizar otra herramienta saldría del presupuesto del mismo.

Se eligieron el número de riesgos detectados planes implementados y riesgos reevaluados ya que son una forma de medir la diferencia que hay entre una herramienta que ayuda a la gestión a otra que depende totalmente de la experiencia del gestor, por lo que al contabilizar estos parámetros vemos si la herramienta ayuda a gestores de un mismo proyecto con una experiencia similar.

Para la ejecución de las pruebas se tomaron dos integrantes del equipo de desarrollo con una experiencia similar a la gestión de riesgos, los cuales se encargarían de realizar la evaluación y gestión de riesgos con el prototipo o la plantilla y los resultados fueron los siguientes:

Durante las pruebas realizadas en la Aplicación móvil para el apoyo a la lectura y comprensión lectora [1] se resolvió que el prototipo ayudo a la gestión de riesgos de la información e infraestructura durante el desarrollo del proyecto, tomando en cuenta las fases de desarrollo en las cuales se encontraba el mismo, también ayudo a la gestión de

riesgos ya cuando la aplicación de encontraba en un ambiente productivo. Esto nos dice que el prototipo puede ser implementado para que esté al alcance de los compañeros que realizan un proyecto de software para fines escolares, incluyendo proyectos parciales o finales, haciendo énfasis en los trabajos terminales ya que es un requisito hacer un análisis de riesgos por proyecto.

La implementación de una herramienta como la que se propone siendo está el prototipo de gestión de riesgos apoyaría a la población estudiantil a general un análisis de riesgos con una metodología y una estructura para realizar la evaluación de riesgos, lo que nos ayuda a crear planes para mitigar y controlar los riesgos a lo largo del ciclo de vida del desarrollo del proyecto y en algunos casos este proyecto en un ambiente productivo.

Como trabajo a futuro, se plantea la exploración de la implementación de sistemas conscientes del contexto en el prototipo, con el fin de adaptar la gestión de riesgos de manera dinámica según las necesidades y la interacción del usuario con el entorno digital. Se considera que la fusión de datos de fuentes heterogéneas, en línea con la detección de inconsistencias y la adaptación contextual, podría ser un enfoque prometedor para fortalecer la gestión de riesgos en entornos digitales en constante evolución.

Para abordar estos desafíos futuros, se lanzará el prototipo para el uso del mismo dentro de tres unidades de aprendizaje en la Escuela Superior de Computo, siendo estas: Análisis y Diseño, Formulación de Proyectos de TI y Gobierno de TI, teniendo estas 3 relación con el análisis y gestión de riesgos, poniendo a disposición la herramienta para los alumnos y profesores, probando en diferentes escenarios y contextos la herramienta y poblando la base de datos.

También se pondrá a disposición de los alumnos que estén realizando su Trabajo Terminal tanto la primer y segunda parte, esto para observar cómo se realiza un análisis de riesgos en proyectos de software, para la primera parte de los Trabajos Terminales, y el seguimiento de los mismos a lo largo del desarrollo del proyecto.

También se propone la investigación y aplicación de métodos de fusión de datos de fuentes heterogéneas en el contexto de la gestión de riesgos en proyectos de software. Se busca mantener la consistencia de los datos y mejorar la capacidad predictiva del sistema mediante la integración de información contextual [8] diversa.

Se espera que esta línea de trabajo contribuya a fortalecer la seguridad de la información en proyectos de software y a proporcionar herramientas más avanzadas y adaptativas para la gestión de riesgos en un entorno digital cada vez más complejo y dinámico.

Agradecimientos. Los resultados de este trabajo se desarrollaron en el marco del proyecto de investigación: Aplicación móvil para el apoyo docente en los procesos enseñanza - aprendizaje de la lectura mediante técnicas de gamificación y machine learning con número de registro asignado por el SIP: 20242235, desarrollado en Instituto Politécnico Nacional.

Referencias

1. Rodríguez, O.E., Huerta, M., Reyes, E.D.: Trabajo Terminal: Aplicación móvil para el apoyo a la lectura y la comprensión lectora. Escuela Superior de Cómputo, Instituto Politécnico Nacional (2024)
2. Ieeeauthorcenter: Information security, cybersecurity and privacy protection, Information security management systems, Requirements. In: ISO/IEC 27001:2022. https://ieeeauthorcenter.ieee.org/wp-content/uploads/IEEE-Reference-Guide.pdf (2022)
3. Standards: Information technology, Security techniques – Information security management systems, In: Overview and vocabulary ISO/IEC 27000:2018. https://standards.iso.org/ittf/PubliclyAvailableStandards/index.html (2018)
4. OWASP: Risk Rating Methodology. https://owasp.org/www-community/OWASP_ Risk_ Rating_Methodology (2022)
5. Project Management Institute: Practices Standard for Project Risk Management. Project Management Institute, Inc., USA (2009)
6. Information security: Cybersecurity and privacy protection: Information security controls. ISO/IEC 27002:2022. International Organization for Standardization (2022)
7. Information Technology: Security techniques -Information security risk management. ISO/IEC 27005:2018. International Organization for Standardization (2018)
8. Julio, M., Guillermo, M.C., Edgard, B.G.: Método de fusión de datos de fuentes heterogéneas para mantener la consistencia de datos. Research in Computing Science Issue, 139, pp. 33–46 (2017)
9. Smartsheet: Plantillas de matriz y gestión de riegos en Excel. https://es.smartsheet.com/all-risk-assessment-matrix-templates-you-need (2023)
10. Pirani: https://www.piranirisk.com/es/ (2023)
11. Aperitisoft: https://info.gartnerdigitalmarkets.com/rpm3solutions-gdm-lp?category=integrated-risk-management&utm_source=Capterra (2023)
12. NVIDIA: Pandas Python. https://www.nvidia.com/en-us/glossary/pandas-python/ (2023)
13. Pandas Documentation: https://pandas.pydata.org/docs/user_guide/dsintro.html (2023)
14. Fine Tuning LLMs: DataCamp. https://www.datacamp.com/tutorial/fine-tuning-large-language-models (2024)
15. Vectorization techniques: https://neptune.ai/blog/vectorization-techniques-in-nlp-guide (2023)
16. Understanding TF-IDF: Geeks for Geeks. https://www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/ (2023)
17. TF-IDF: LearnDataSci. https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/ (2024)
18. Classification in Machine Learning: DataCamp. https://www.datacamp.com/blog/classification-machine-learning (2024)
19. Random Forest Classifier using Scikit-learn: Geeks for Geeks. https://www.geeksforgeeks.org/random-forest-classifier-using-scikit-learn/ (2024)

Evolution of Encryption Measures

ISSN 1870-4069

Manuel Alejandro Cardona-López[1], Julieta Mazzetti-Riveros[2], Laura Del-Rosario-González[2], Karla Denisse Taba-Díaz[2], Juan Carlos Chimal-Eguía[1]

[1] Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico

[2] Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica, Mexico

{mcardonal2022,chimal}@cic.ipn.mx, {jmazzettir2200,ldelrosariog1900,ktabad2300}@alumno.ipn.mx

Abstract. The article examines the evolution of encryption measures, focusing on how results have improved over time and emphasizing recent security achievements. It discusses various metrics used to evaluate the performance of encryption algorithms, including Number of Pixel Change Rate, Unified Average Changing Intensity, correlation coefficient, information entropy, Chi-square test, and key space analysis. These metrics are crucial for ensuring the integrity and confidentiality of images, and their improvements over time are commonly aimed. Also, it is analyzed the key advancements in encryption algorithms that have led to improve these encryption metrics, from older to more recent works. The analysis focuses on the critical differences that have contributed to enhanced security. In addition, the article underscores the importance of standardizing security measures in Cryptography to enhance the comparison of robustness between different encryption systems against advanced attacks to various types of data and applications.

Keywords: Correlation, encryption, entropy, image, security

1 Introduction

The massive transfer of data, including digital images, has become a common practice, presenting significant challenges in terms of security and privacy. To address these challenges, image encryption systems have been developed. These systems employ algorithms to convert readable images, which contain clear messages, into encrypted versions. This process involves the use of a specific key to decrypt the image and restore it to its original state, thereby ensuring the confidentiality, integrity, and authenticity of the image during its transmission, reception, and storage [12, 34]. Such systems are critically applied in environments that prioritize information security, such as medical images

with confidential diagnoses, personal identifiers, and sensitive data in military applications [32].

The effectiveness of encryption algorithms heavily relies on their ability to withstand unauthorized decryption attempts, prompting the development of security analyses to assess their robustness and reliability across various usage scenarios [38]. Consequently, this work explores various existing analyses to verify the quality of different encryption proposals. These include evaluation metrics like NPCR (Number of Pixels Changed Rate), UACI (Unified Average Changing Intensity), correlation coefficient analysis, and histogram analysis, among others [29]. Furthermore, it details the quantifiable evolution and improvements of these analyses over recent years, emphasizing advancements in encryption techniques and their adaptation to evolving security threats in digital environments.

The paper is structured as follows. Section 2 introduces the key metrics in the field of image encryption, which are essential for tracking the evolution of reported values. This section provides a detailed explanation to underscore the importance of these metrics in evaluating encryption effectiveness. Section 3 presents the collected data on the evolution of each metric, highlighting improvements over time, with corresponding references and details of the images used for the evaluation. Section 4 outlines the main differences between older and more recent studies that have contributed to these advancements in metrics. Finally, Section 5 concludes the paper.

2 Background of Encryption Metrics

In this section, we provide a brief overview of each metric used for evaluating image encryption security. This includes the purpose of each evaluation, the equations employed, the procedures for their application, and the target values required to indicate a high level of security. The measures included, in order of appearance, are NPCR, UACI, Correlation Coefficient, Information Entropy, Chi-Square Test, and Key Space. The reported values of these measures are then collected for analysing the improvements in encryption security.

2.1 NPCR

The Number of Pixel Change Rate (NPCR) is a metric used to calculate the percentage of differing pixels D between two encrypted images C and C' whose original images differ by exactly one pixel. This metric is crucial for evaluating an encryption algorithm's resistance to differential attacks, which test the algorithm's sensitivity to the smallest changes in the plain image [27]. An NPCR value of 100% is desirable, indicating that the encryption technique is highly resistant to such attacks, as it shows that a minute change in the original image produces a total change in the encrypted image [17]. Conversely, a value of 0% means no change in the resultant encrypted pixels. NPCR is calculated using the following Equation (1), and the difference of two image is shown in Equation (2):

$$\text{NPCR} = \frac{\Sigma_{i,j} D(i,j)}{M \times N} \times 100\%, \tag{1}$$

$$D(i,j) = \begin{cases} 0 \text{ if } C(i,j) = C'(i,j) \\ 1 \text{ if } C(i,j) \neq C'(i,j) \end{cases}. \tag{2}$$

2.2 UACI

The Unified Average Changing Intensity (UACI) measures the average of the absolute intensity differences between two encrypted images corresponding to original images that differ by only a single pixel. This metric is important for evaluating how changes are distributed within the encrypted image in response to slight variations in the original image [20]. A high UACI indicates that the encryption algorithm effectively diffuses changes throughout the encrypted image, which is essential for encryption security [21]. UACI is calculated using the formula in Equation (3), with a desirable value of 33%:

$$\text{UACI} = \frac{1}{M \times N} [\Sigma_{i,j} \frac{\mid C(i,j) - C'(i,j) \mid}{255}] \times 100\%. \tag{3}$$

2.3 Correlation Coefficient

The correlation coefficient (CC) in Equation (4) is a measure used to evaluate the degree of relationship between two variables x, y, in this case, the pixel values [11]. The coefficient indicates their level of similarity. The CC is defined using covariance (Cov) of Equation (5) and variance (Var) of pixel values that is the square of the deviation D in Equation (7), and the expected value E in Equation (6). All these values are for the $N \times N$ total number of pixels in an image. In an original image, the horizontal, diagonal, and vertical values of adjacent pixels are strongly correlated. An effective encryption method reduces this correlation in the encrypted image, bringing the correlation coefficient value closer to zero [28]. This reduction implies that the pixels in the encrypted image are less predictable and, therefore, more secure against statistical attacks. In contrast, a CC of 1 implies perfect correlation, meaning the pixels are linearly identical, indicating a failed encryption process. The correlation coefficient is calculated using equation, which evaluates the quality of the cryptographic system:

$$\text{CC}_{xy} = \frac{\text{Cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{4}$$

$$\text{Cov}(x,y) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))(y_i - E(y)), \tag{5}$$

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i, \tag{6}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2. \tag{7}$$

2.4 Information Entropy

Entropy H is a measure of the expected value or average amount of information that can be extracted from a message, in this case, an image [33]. It represents the quantity of information present and indicates the degree of uncertainty or randomness in the data. Introduced by Claude Shannon in 1949, entropy values range from 0 to 8 and should be close to 8 for an 8-bit image representation. A high entropy value of 8.0 indicates a uniform distribution of pixel values, which is desirable for an encrypted image because it reflects a high level of unpredictability and randomness [47]. This uniform distribution means each pixel has an equal probability of appearing, ensuring maximum entropy. Conversely, if the entropy of the encrypted image is significantly less than 8, it suggests the image has patterns or predictability, posing a security risk. Entropy can be calculated using Equation (8) and is based on the probabilities $p(x_i)$ of the occurrence of each pixel value x_i in an image with N pixels:

$$H(x) = -\sum_{i=0}^{N-1} P(x_i)\log_2 P(x_i). \tag{8}$$

2.5 Chi-square Test

The Chi-square χ^2 test is used to evaluate the uniformity of histograms, with the aim of determining the resistance to statistical attacks [7]. A lower Chi-square value indicates better uniformity, where the ideal value is equal to zero, suggesting the strongest resistance [48]. To confirm the uniform distribution of the resulting ciphered images more precisely, the Chi-square test is conducted, demonstrating that the encrypted image follows a uniform distribution. It is based on the expected frequency E_i of a uniform histogram and the observed o_i from the image, following the procedure of Equation (9):

$$\chi^2 = \sum_{i=1}^{255}\frac{(o_i - E_i)^2}{E_i}. \tag{9}$$

2.6 Key Space Analysis

Key Analysis (KA) is crucial in any encryption algorithm, as the algorithm's security heavily relies on the strength of the secret keys. Assessing the effectiveness of an encryption system involves analyzing the key space, where the key is a unique value used for encrypting data, ensuring only authorized parties can decrypt the information. Desirable properties of strong secret keys include a large key space. The key space depends on the size of the secret key;

Table 1. Numer of pixel change rate (NPCR).

Evolution of Encryption Measures

Proposal	Publication year	Reported value	Image size	Image name
Ref. [25]	2012	99.5850	256×256	Lena
Ref. [30]	2015	99.5789	200×200	Peppers
Ref. [50]	2018	99.5697	256×256	Cameraman
Ref. [42]	2020	99.6100	512×512	Lena
Ref. [49]	2023	99.6399	256×256	Baboon
Ref. [16]	2024	99.6500	256×256	Medicine

a larger size makes it more challenging for an attacker to guess the key. An excellent image encryption algorithm requires a robust key space, indicating resistance against brute-force attacks. With a large key space, exhaustive key searches become impractical, requiring an infeasible number of operations (e.g., 2^{128} for a 128-bit key). In other words, a larger key size significantly reduces the likelihood of a brute-force attack succeeding, given the vast number of potential combinations that must be tested to find the correct key.

3 Encryption Metrics Evolution

This section presents the evolution of encryption results across eight tables. The tables compile data from various works over the years, including the measures introduced in the previous section, along with the image name and size. Over time, these measures have progressively approached optimal values, illustrating how each new cryptosystem generally enhances security outcomes. The data in each table is organized chronologically, from the oldest to the most recent publications.

3.1 NPCR

In this context, the NPCR (Number of Pixel Change Rate) aims to achieve a 100% difference in the encrypted pixels between two encrypted images that differ by only one pixel in their plain representation. Recent advancements have brought this value closer to the ideal 100%. This improvement is illustrated in Table 1.

3.2 UACI

For this metric, it is important that not only the pixels are different, but also differ so far from in its intensity levels. A value closer to 33% is the optimal, where in the recent years has been kept.

Table 2. Unified average changing intensity (UACI).

Proposal	Publication year	Reported value	Image size	Image name
Ref. [30]	2012	32.9495	200×200	Baboon
Ref. [37]	2015	33.3430	256×256	Lena
Ref. [50]	2018	33.3618	256×256	Cameraman
Ref. [41]	2020	33.4700	512×512	Cameraman
Ref. [49]	2023	33.4199	256×256	Lena
Ref. [22]	2023	33.6500	256×256	Barbara

Table 3. Correlation Coefficient (Horizontal Direction).

Proposal	Publication year	Reported value	Image size	Image name
Ref. [3]	2007	0.0308	256×256	Lena
Ref. [45]	2010	0.0036	256×256	Lena
Ref. [25]	2011	0.0068	256×256	Lena
Ref. [13]	2015	0.0273	512×512	Boat
Ref. [44]	2016	−0.0060	1024×1024	Male
Ref. [18]	2017	−0.0163	256×256	Colour Flower
Ref. [46]	2018	0.0004	256×256	Clock
Ref. [50]	2018	−0.0017	256×256	Cameraman
Ref. [1]	2024	0.0008	256×256	Lena

3.3 Correlation Coefficient

The ideal correlation coefficient is 0.0, regardless of the direction in which the correlation is measured. The values should be around this ideal, indicating no correlation. Over time, approaches have increasingly approximated this ideal. Improvements in correlation can be seen in Tables 3, 4, and 5.

3.4 Information Entropy

For pixels with 256 intensity values, the optimal entropy value is 8.0. Recently, image cryptosystems have been getting closer to this value, as shown in Table 6.

3.5 Chi-square Test

The goal of the Chi-square test is to achieve a value of 0, indicating no difference between the histogram of the encrypted image and a uniform distribution. The differences have decreased over the years, as detailed in Table 7.

Table 4. Correlation Coefficient (Vertical Direction).

Proposal	Publication year	Reported value	Image size	Image name
Ref. [3]	2007	0.0304	256×256	Lena
Ref. [45]	2010	0.0023	256×256	Lena
Ref. [25]	2011	0.0091	256×256	Lena
Ref. [13]	2015	−0.0321	512×512	Boat
Ref. [44]	2016	−0.0103	1024×1024	Male
Ref. [18]	2017	0.0185	256×256	Colour Flower
Ref. [46]	2018	−0.0054	256×256	Clock
Ref. [50]	2018	−0.0279	256×256	Cameraman
Ref. [1]	2024	−0.0003	256×256	Lena

Table 5. Correlation Coefficient (Diagonal Direction).

Proposal	Publication year	Reported value	Image size	Image name
Ref. [3]	2007	0.0317	256×256	Lena
Ref. [45]	2010	0.0039	256×256	Lena
Ref. [25]	2011	0.0063	256×256	Lena
Ref. [13]	2015	-0.0060	512×512	Boat
Ref. [44]	2016	0.0031	1024×1024	Male
Ref. [18]	2017	-0.0129	256×256	Colour Flower
Ref. [46]	2018	0.0111	256×256	Clock
Ref. [50]	2018	0.0047	256×256	Cameraman
Ref. [1]	2024	0.0002	256×256	Lena

Table 6. Information Entropy.

Proposal	Publication year	Reported value	Image size	Image name
Ref. [39]	2009	7.9970	256×256	Lena
Ref. [6]	2012	7.9890	256×256	Lena
Ref. [24]	2013	7.9881	256×256	Pepper
Ref. [26]	2015	7.9637	256×256	Cameraman
Ref. [2]	2016	7.9979	256×256	Baboon
Ref. [10]	2019	7.9992	256×256	Lena
Ref. [17]	2021	7.9994	512×512	Lena
Ref. [43]	2023	7.9998	1024×1024	Barbara
Ref. [36]	2024	7.9998	1024×1024	Male

Table 7. Chi-square test.

Proposal	Publication year	Reported value	Image size	Image name
Ref. [8]	2009	290	256×256	Lena
Ref. [9]	2013	260	128×128	Lake
Ref. [19]	2014	243	256×256	Lena
Ref. [15]	2016	251.2	256×256	Lena
Ref. [17]	2021	233.729	512×512	Lena
Ref. [43]	2023	207.445	256×256	Mountain

3.6 Key Space Analysis

The key space should be as large as possible to prevent feasible brute force attacks. The size of the key space has increased over time, enhancing security by expanding the number of possible combinations that must be tested to decrypt the image. This trend is illustrated in Table 8.

Table 8. Key Space.

Proposal	Publication year	Reported value
Ref. [35]	2005	2^{128}
Ref. [31]	2006	2^{150}
Ref. [5]	2010	2^{256}
Ref. [23]	2015	2^{299}
Ref. [40]	2018	2^{312}
Ref. [14]	2021	2^{604}
Ref. [4]	2023	2^{1658}

4 Improvements in Algorithms

The following list summarizes the advancements in encryption algorithms that have led to improved encryption metrics. These insights are drawn from the collection of papers reporting on encryption metrics. A comparative analysis of older and more recent works highlights the key differences that have contributed to enhanced security. While the fundamental concepts and features discussed have persisted over the years, their application has evolved to provide stronger security measures.

1. Chaotic systems: Although encryption procedures based on chaotic systems have been in use since the early 2000s, their properties have significantly

improved. Modern chaotic systems now feature a broader range of chaotic parameters, leading to enhanced chaotic characteristics.

2. XOR operations: XOR operations have long been a fundamental technique in encryption. While they remain crucial, they are now complemented by additional methods such as SHA functions and substitution operations, applied before or after the XOR process to create a more chaotic bit distribution.
3. DNA application: DNA sequences have been combined with chaotic maps in encryption. Recent advancements have parallelized DNA coding, reducing execution time and allowing for more encryption rounds, thereby enhancing security.
4. Permutations: Previously, only one permutation was typically applied per round, often generated from chaotic maps. Now, permutations are applied in diverse ways and derived from various sources, such as random walks within the same encryption round, which effectively reduces correlation.
5. Encryption metrics: Measuring current results is the first step toward improvement. Over time, more encryption metrics have been introduced to evaluate the chaotic distribution of encrypted data, acknowledging that while each metric is necessary, none alone is sufficient.
6. Key space: The key space has expanded significantly. Earlier encryption schemes operated with smaller key sizes due to limited computational resources. However, increasing key sizes now allows for the inclusion of additional information within the keys, contributing to better securityoutcomes.
7. Symmetric cryptography: Symmetric cryptography, a staple in both older and newer proposals, remains favored due to its efficient execution time, even that new proposals have more encryption rounds, making it more suitable than asymmetric cryptography in many contexts.
8. Histogram: Permutations are essential for breaking the relationship between pixels without altering pixel values, thus maintaining the histogram. Over time, chaos has been increasingly applied to modify pixel values to achieve a uniform histogram, not just positions. Chaos has enhanced the substitution process too.

5 Conclusion

The imperative to enhance security levels in image encryption drives this article to review the results of various encryption approaches. Currently, past proposals may appear insecure due to their results, highlighting the ongoing need for improved security outcomes. On the other hand, while older and current encryption algorithms share fundamental concepts and features, their application has evolved over the years to deliver stronger security measures. In addition, to make image cryptography more consistent and promising, it is crucial to establish robust security frameworks with consistent evaluation metrics. We reviewed the most common security metrics, displaying standardized measures

that can be universally adopted. This standardization is essential for enabling reliable comparisons of different algorithms and their effectiveness over time. Such adaptability ensures that the benefits of the framework extend to specific application scenarios, guiding future research and development efforts toward improving evaluation metrics in image cryptography.

Acknowledgments. The authors would like to thank the Instituto Politécnico Nacional of México (Secretaría Académica, Comisión de Operación y Fomento de Actividades Acadé-micas COFAA, SIP, ESIME ZAC, and CIC), and the CONAHCyT for their support in the development of this work.

References

1. Abed, Q.K., Al-Jawher, W.A.M.: Optimized color image encryption using arnold transform, uruk chaotic map and gwo algorithm. Journal Port Science Research 7(3), 219–236 (2024)
2. Ahmad, J., Hwang, S.O.: A secure image encryption scheme based on chaotic maps and affine transformation. Multimedia Tools and Applications 75, 13951–13976 (2016)
3. Ahmed, H.E.d.H., Kalash, H.M., Allah, O.S.F.: An efficient chaos-based feedback stream cipher (ecbfsc) for image encryption and decryption. Informatica 31(1), 121–129 (2007)
4. Alexan, W., Alexan, N., Gabr, M.: Multiple-layer image encryption utilizing fractional-order chen hyperchaotic map and cryptographically secure prngs. Fractal and Fractional 7(4), 287 (2023)
5. Amin, M., Faragallah, O.S., Abd El-Latif, A.A.: A chaotic block cipher algorithm for image cryptosystems. Communications in Nonlinear Science and Numerical Simulation 15(11), 3484–3497 (2010)
6. Bahrami, S., Naderi, M.: Image encryption using a lightweight stream encryption algorithm. Advances in Multimedia 2012(1), 767364 (2012)
7. Budiman, F., Andono, P.N., Setiadi, M., et al.: Image encryption using double layer chaos with dynamic iteration and rotation pattern. International Journal of Intelligent Engineering & Systems 15(2) (2022)
8. Etemadi Borujeni, S., Eshghi, M.: Chaotic image encryption design using tompkins-paige algorithm. Mathematical problems in engineering 2009(1), 762652 (2009)
9. Etemadi Borujeni, S., Eshghi, M.: Chaotic image encryption system using phase-magnitude transformation and pixel substitution. Telecommunication Systems 52, 525–537 (2013)
10. Ge, R., Yang, G., Wu, J., Chen, Y., Coatrieux, G., Luo, L.: A novel chaos-based symmetric image encryption using bit-pair level process. IEEE Access 7, 99470–99480 (2019)
11. Guleria, V., Kumar, Y., Mishra, D.C.: Multiple colour image encryption using multiple parameter frdct, 3d arnold transform and rsa. Multimedia Tools and Applications 83(16), 48563–48584 (2024)
12. Haddad, S., Coatrieux, G., Moreau-Gaudry, A., Cozic, M.: Joint watermarking-encryption-jpeg-ls for medical image reliability control in encrypted and compressed domains. IEEE Transactions on Information Forensics and Security 15, 2556–2569 (2020)

13. Hua, Z., Zhou, Y., Pun, C.M., Chen, C.P.: 2d sine logistic modulation map for image encryption. Information Sciences 297, 80–94 (2015)
14. Iqbal, N., Naqvi, R.A., Atif, M., Khan, M.A., Hanif, M., Abbas, S., Hussain, D.: On the image encryption algorithm based on the chaotic system, dna encoding, and castle. IEEE Access 9, 118253–118270 (2021)
15. Jallouli, O., El Assad, S., Chetto, M.: Robust chaos-based stream-cipher for secure public communication channels. In: Procedings of the 11th International Conference for Internet Technology and Secured Transactions (ICITST). pp. 23–26. IEEE, Barcelone, Spain (2016)
16. Jiang, D., Tsafack, N., Boulila, W., Ahmad, J., Barba-Franco, J.: Asb-cs: Adaptive sparse basis compressive sensing model and its application to medical image encryption. Expert Systems with Applications 236, 15 (2024)
17. Khairullah, M.K., Alkahtani, A.A., Bin Baharuddin, M.Z., Al-Jubari, A.M.: Designing 1d chaotic maps for fast chaotic image encryption. Electronics 10(17), 2116 (2021)
18. Khan, J.S., Ahmad, J.: Chaos based efficient selective image encryption. Multidimensional Systems and Signal Processing 30, 943–961 (2019)
19. Khanzadi, H., Eshghi, M., Borujeni, S.E.: Image encryption using random bit sequence based on chaotic maps. Arabian Journal for Science and engineering 39, 1039–1047 (2014)
20. Kumar, A., Dua, M.: Novel pseudo random key & cosine transformed chaotic maps based satellite image encryption. Multimedia tools and applications 80(18), 27785–27805 (2021)
21. Kumar, A., Dua, M.: A gru and chaos-based novel image encryption approach for transport images. Multimedia Tools and Applications 82(12), 18381–18408 (2023)
22. Kumar, S., Sharma, D.: A chaotic based image encryption scheme using elliptic curve cryptography and genetic algorithm. Artificial Intelligence Review 57(4), 87 (2024)
23. Li, S., Chen, G., Mou, X.: On the dynamical degradation of digital piecewise linear chaotic maps. International journal of Bifurcation and Chaos 15(10), 3119–3151 (2005)
24. Liu, H., Wang, X., Kadir, A.: Color image encryption using choquet fuzzy integral and hyper chaotic system. Optik-International Journal for Light and Electron Optics 124(18), 3527–3533 (2013)
25. Loukhaoukha, K., Chouinard, J.Y., Berdai, A.: A secure image encryption algorithm based on rubik' s cube principle. Journal of Electrical and Computer Engineering 2012(1), 13 (2012)
26. Luo, Y., Du, M., Liu, J.: A symmetrical image encryption scheme in wavelet and time domain. Communications in Nonlinear Science and Numerical Simulation 20(2), 447–460 (2015)
27. Ma, Y.: Research and application of big data encryption technology based on quantum lightweight image encryption. Results in Physics 54, 107057 (2023)
28. Mali, K., Chakraborty, S., Roy, M.: A study on statistical analysis and security evaluation parameters in image encryption. entropy 34, 36 (2015)
29. Mohammad, O.F., Rahim, M.S.M., Zeebaree, S.R.M., Ahmed, F.: A survey and analysis of the image encryption methods. International Journal of Applied Engineering Research 12(23), 13265–13280 (2017)
30. Pareek, N.K.: Design and analysis of a novel digital image encryption scheme. International Journal of Network Security & Its Applications 4(2), 13–15 (2012)
31. Pareek, N.K., Patidar, V., Sud, K.K.: Image encryption using chaotic logistic map. Image and vision computing 24(9), 926–934 (2006)

32. Patel, K.D., Belani, S.: Image encryption using different techniques: A review. International Journal of Emerging Technology and Advanced Engineering 1(1), 30–34 (2011)
33. Shen, H., Shan, X., Xu, M., Tian, Z.: A new chaotic image encryption algorithm based on transversals in a latin square. Entropy 24(11), 1574 (2022)
34. Singh, K.N., Singh, A.K.: Towards integrating image encryption with compression: A survey. ACM Transactions on Multimedia Computing, Communications, and Applications 18(3), 1–21 (2022)
35. Socek, D., Li, S., Magliveras, S.S., Furht, B.: Short paper: Enhanced 1-d chaotic key-based algorithm for image encryption. In: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05). pp. 406–407. Athens, Greece (2005)
36. Song, W., Fu, C., Zheng, Y., Zhang, Y., Chen, J., Wang, P.: Batch image encryption using cross image permutation and diffusion. Journal of Information Security and Applications 80, 103686 (2024)
37. Stoyanov, B., Kordov, K.: Image encryption using chebyshev map and rotation equation. Entropy 17(4), 2117–2139 (2015)
38. Tiken, C., Samlı, R.: A comprehensive review about image encryption methods. Harran Üniversitesi Mühendislik Dergisi 7(1), 27–49 (2022)
39. Tong, X., Cui, M., Wang, Z.: A new feedback image encryption scheme based on perturbation with dynamical compound chaotic sequence cipher generator. Optics Communications 282(14), 2722–2728 (2009)
40. Ur Rehman, A., Liao, X., Ashraf, R., Ullah, S., Wang, H.: A color image encryption technique using exclusive-or with dna complementary rules based on chaos theory and sha-2. Optik 159, 348–367 (2018)
41. Xu, C., Sun, J., Wang, C.: An image encryption algorithm based on random walk and hyperchaotic systems. International Journal of Bifurcation and Chaos 30(04), 2129–2151 (2020)
42. Xue, X., Zhou, D., Zhou, C.: New insights into the existing image encryption algorithms based on dna coding. Plos one 15(10), 31 (2020)
43. Ye, G.D., Wu, H.S., Huang, X.L., Tan, S.Y.: Asymmetric image encryption algorithm based on a new three-dimensional improved logistic chaotic map. Chinese Physics B 32(3), 030504 (2023)
44. Ye, G., Zhao, H., Chai, H.: Chaotic image encryption algorithm using wave-line permutation and block diffusion. Nonlinear Dynamics 83, 2067–2077 (2016)
45. Zhang, Q., Guo, L., Wei, X.: Image encryption using dna addition combining with chaotic maps. Mathematical and Computer Modelling 52(11-12), 2028–2035 (2010)
46. Zhang, Y.: The unified image encryption algorithm based on chaos and cubic s-box. Information Sciences 450, 361–377 (2018)
47. Zhao, J., Wang, S., Zhang, L.: Block image encryption algorithm based on novel chaos and dna encoding. Information 14(3), 150 (2023)
48. Zhu, H., Dai, L., Liu, Y., Wu, L.: A three-dimensional bit-level image encryption algorithm with rubik's cube method. Mathematics and Computers in Simulation 185, 754–770 (2021)
49. Zhu, S., Deng, X., Zhang, W., Zhu, C.: Image encryption scheme based on newly designed chaotic map and parallel dna coding. Mathematics 11(1), 231 (2023)
50. Zhu, S., Zhu, C., Wang, W.: A new image encryption algorithm based on chaos and secure hash sha-256. Entropy 20(9), 716 (2018)

ISSN 1870-4069

Propuesta de automatización de aplicación web de gestión de riesgos de proyectos de software mediante inteligencia artificial

Axel Omar Alarcón Padilla, Gerardo Castro Rangel,
Jessie Paulina Guzmán Flores

Instituto Politécnico Nacional,
Escuela Superior de Cómputo,
México

{aalarconp1600, gcastror1600}@alumno.ipn.mx,
jguzmanf@ipn.mx

Resumen. El proceso de gestión de riesgos de software es una labor esencial y compleja que se lleva a cabo conforme todo el ciclo de vida de desarrollo. Tomando de base una aplicación web, y con el propósito de auxiliar a los gestores de riesgos a facilitar dicha labor, el presente trabajo propone mediante la implementación de un modelo de lenguaje preentrenado y ajustado mediante un Fine Tuning de dominio específico, y un modelo de clasificación basado en múltiples arboles de decisiones, la automatización de la identificación, valoración de riesgos y de planes de seguridad de prevención, mejorando la identificación temprana de los riesgos y a su gestión de forma proactiva.

Palabras clave: Gestión de riesgos, aplicación web, inteligencia artificial, modelo de lenguaje, modelo de clasificación, propuesta de automatización.

Proposal for Automation of Web Application for Software Project Risk Management Using artificial Intelligence

Abstract. The software risk management process is an essential and complex task that is carried out throughout the development life cycle. Based on a web application, and with the purpose of helping risk managers to facilitate this task, this paper proposes, through the implementation of a pre-trained language model adjusted by means of a domain-specific Fine Tuning and a classification model based on multiple decision trees, the automation of the identification, risk assessment and prevention security plans, improving the early identification of risks and their management in a proactive way.

Keywords: Risk management, web application, artificial intelligence, language model, classification model, automation proposal.

1. Introducción

En el caso específico de un proyecto de software, un riesgo de seguridad de la información es un efecto de incertidumbre en los objetivos de la seguridad de la información. La causa de los riesgos es un incidente, el cual perjudica al ciclo de vida del desarrollo software en cualquier etapa. Por lo tanto, un equipo de desarrollo necesita tener un plan de acción frente a incidentes, compuesto por valoración, identificación, administración y tratamiento de riesgos [1].

La gestión de riesgos de un proyecto pretende identificar y priorizar los riesgos antes de que se produzcan, y proporcionar información orientada a la acción a los gestores del proyecto. Esta orientación exige considerar sucesos que pueden ocurrir o no, por lo que se describen en términos de probabilidad de que ocurran, además de otras dimensiones como su impacto en los objetivos [2].

El desarrollo de una aplicación web de gestión de riesgos para proyectos de software tiene como objetivo ofrecer un enfoque específico con base en diferentes marcos de trabajo, diferenciándose de otras herramientas de gestión de riesgos hacia infraestructuras de TI y proyectos en general.

La actual implementación de la herramienta RiskProtego se basa en auxiliar al gestor de riesgos de un proyecto de software a la realización del proceso integral de gestión y de tratamiento de riesgos. El gestor de riesgos con su experiencia debe considerar amenazas y vulnerabilidades que tenga su proyecto para poder identificar dichos riesgos, y con base en la herramienta, dar un valor conforme a marcos de trabajo. Es decir, el proceso es manual dentro de una aplicación que auxilia al gestor a identificar, valorar y tratar los riesgos mediante un enfoque estructurado y eficiente [3].

Como se mencionó, tradicionalmente, los resultados de las valoraciones de riesgos dependen de la fiabilidad de los datos utilizados y de las habilidades y experiencia de la persona que realiza la valoración, dicho caso se aplica a los módulos proporcionados por RiskProtego, en el que el usuario gestor lleva a cabo este proceso. Sin embargo, es posible realizar valoraciones más precisas al recurrir a la inteligencia artificial, ya que una de sus competencias básicas es la agregación e interpretación de datos [4]. Asimismo, la inteligencia artificial en este ámbito reduce el trabajo manual, que es bastante intensivo, y que requiere un gran esfuerzo de los gestores. De misma forma puede detectar a escala patrones y prioridades [5].

Los algoritmos de inteligencia artificial pueden evaluar datos no estructurados, y patrones relacionados con incidentes pasados que pueden identificarse y convertirse en predictores de riesgos. A partir de estos patrones, pueden construirse escenarios plausibles de cara al futuro para predecir sucesos y proyectar riesgos [4].

Actualmente, existen enfoques específicos para la utilización de algoritmos de inteligencia artificial en la gestión de riesgos, como modelos de regresión de rangos, modelos inteligencia artificial explicable (XAI) [6] y de modelos grandes de lenguaje [4].

La intención de este artículo es dar una propuesta al desarrollo de RiskProtego para automatizar ciertas secciones de la gestión de riesgos mediante inteligencia artificial, específicamente, utilizando un modelo de lenguaje y un modelo de clasificación. Dependiendo del proyecto específico a registrar en la herramienta, la automatización

se basa en identificar riesgos y dar un valor de probabilidad e impacto como una primera valoración, y a su vez, dar una acción de prevención para dichos riesgos de forma automática.

Esta automatización busca mejorar la eficiencia del proceso de gestión de riesgos, permitiendo a los gestores concentrarse en aspectos más estratégicos y complejos, en vez de identificar riesgos que se repiten en muchos proyectos.

2. Estado actual de la aplicación

RiskProtego fue desarrollado utilizando el microframework de desarrollo web back-end Flask, basado en el lenguaje de programación Python. Para el frond-end, se utilizó el motor de plantillas Jinja junto al framework CSS Bootstrap, que complementa a Flask permitiendo la separación de la lógica de presentación con la lógica de negocio. Finalmente, el equipo de desarrollo utilizó la base de datos relacional MySQL utilizando la librería ORM SQLAlchemy para su conexión con Python [3].

2.1 Módulo de gestión de proyectos y usuarios

RiskProtego permite la sesión de dos tipos de usuarios: el gestor de riesgos y un participante. El gestor de riesgos utiliza la aplicación para registrar sus proyectos, registrar los activos informáticos de cada proyecto, e identificar, valorar, priorizar y tratar los riesgos identificados, es decir, tiene el acceso completo de la aplicación y sus módulos.

El usuario participante forma parte del proceso de tratamiento, pues es el encargado de dar reportes de los avances de las acciones de prevención al gestor. Estos pueden ser supervisores de área, jefes de área, técnicos de área, u otros roles responsables de mantener la integridad y el orden de los reportes de tareas dentro de la organización [3].

2.2 Módulo de inventario de activos

Los proyectos de software cuentan con activos informáticos, tales como documentos, dispositivos de redes, plataformas o bases de datos. Dichos activos informáticos son susceptibles a riesgos, por lo que se deben registrar para tener constancia de ellos para propósitos informativos y para la valorización de los riesgos identificados con estos.

La sección de activos consta de dos partes: inventario y evaluación. En el inventario de activos el gestor puede visualizar datos, los cuales se componen de una ficha técnica sin métricas de valoración para el análisis de riesgos. Mientras que, en la sección de evaluación, se les dan valores a las variables de confidencialidad, disponibilidad e integridad de los activos del inventario, dando un valor de sensibilidad del activo [3].

2.3 Módulo de gestión de riesgos

Tendrá desglosados cada riesgo identificado con sus activos asociados, y su amenaza, vulnerabilidad, probabilidad de ocurrencia e impacto. Además de la matriz de riesgos con un semáforo colorimétrico en el que se visualizará la valoración de cada riesgo. Para valorizar el riesgo, el gestor debe identificar amenazas y evaluar vulnerabilidades utilizando valores numéricos [3]. El marco de trabajo de valoración de riesgos del OWASP da fórmulas y valores específicos a cada factor [7], de los cuales derivan valores específicos para estimar la probabilidad y el impacto del riesgo. Se calcula el riesgo con respecto a dichos valores numéricos y con dicha información, se priorizan los activos en función de la magnitud de su riesgo. Los activos con riesgos más altos reciben una atención especial en términos de medidas del plan de seguridad [3].

2.4 Módulo de plan de seguridad

Permite al gestor de riesgos registrar acciones para el tratamiento de los riesgos identificados. Este módulo ayuda al usuario gestor a tener un registro de lo que el equipo ha realizado para evitar, aceptar, modificar o trasladar los riesgos, dando la responsabilidad de cada acción a un usuario participante, el cual deberá reportar periódicamente cada avance en la acción al gestor, incluyendo posibles cancelaciones [3].

3. Propuesta

Partiendo de las lecciones aprendidas al implementar RiskProtego en el proyecto de aplicación móvil para el apoyo docente en los procesos enseñanza - aprendizaje de la lectura y de la lectura, se requiere implementar técnicas de inteligencia artificial para la automatización. La automatización del proceso de gestión de riesgos se puede dar con la generación automática de riesgos y acciones para cada proyecto, utilizando de referencia una base de datos de la aplicación poblada con una cantidad de proyectos, su inventario de activos, riesgos identificados y planes de seguridad. Es importante destacar que estos datos deben ser registrados por los usuarios gestores de riesgos, tomando en cuenta que ellos realizan dicho proceso conforme a su experiencia y conocimiento en el área, por lo que la base de datos debe contar con registros verdaderos y revisados manualmente por dichos gestores. La base fundamental de esta propuesta es la base de datos debido que se requiere una contextualización específica de la información para el entrenamiento de los modelos y con ello, mantener la consistencia de los datos y mejorar la capacidad predictiva de la aplicación [8]. El modelo a utilizar para la generación de texto se basa en un modelo de lenguaje grande, mientras que para la generación de variables numéricas se utilizará un modelo de clasificación.

En la Figura 1 se aprecia el diagrama relacional de la base de datos de la aplicación, incluyendo las tablas de todos los módulos descritos anteriormente. Sin embargo, para

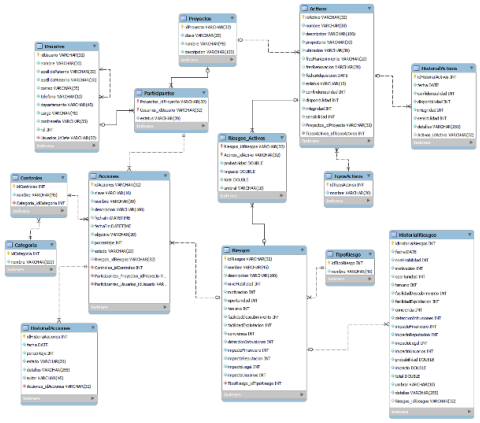

Fig. 1. Diagrama relacional de la base de datos de la aplicación.

Tabla 1. Atributos de un riesgo a tomar en consideración.

Tipo de dato	Nombre	Descripción
VARCHAR(45)	nombre	Descripción corta del riesgo, utilizada para identificación dentro de la interfaz junto a la clave.
VARCHAR(110)	descripcion	Descripción amplia del riesgo.
VARCHAR(45)	amenaza	Descripción en texto de la amenaza potencial que puede efectuar el riesgo.
VARCHAR(45)	vulnerabilidad	Descripción en texto de la vulnerabilidad que puede ser explotada al efectuarse el riesgo.
INTEGER	nivelHabilidad motivación oportunidad tamaño	Subfactores del factor numérico de amenaza, utilizado para el cálculo de probabilidad.
INTEGER	facilidadDescubrimiento facilidadExplotacion conciencia deteccionIntrusiones	Subfactores del factor numérico de vulnerabilidad, utilizado para el cálculo de probabilidad.
INTEGER	sensibilidad	Sensibilidad máxima de los activos asociados al riesgo, parte del factor de impacto técnico, utilizado para el cálculo del impacto del riesgo.
INTEGER	impactoFinanciero impactoReputacion impactoLegal impactoUsuarios	Subfactores del factor numérico de impacto empresarial, utilizado para el cálculo del impacto del riesgo.

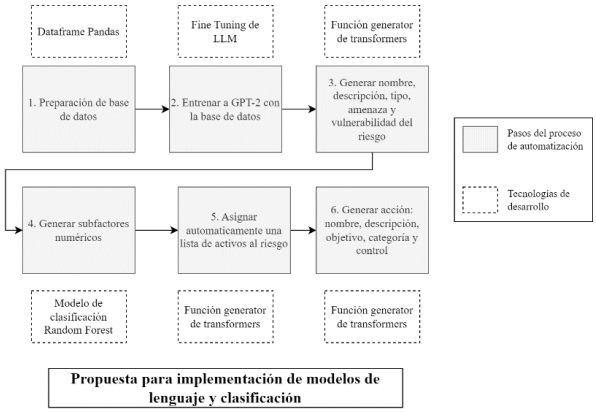

Fig. 2. Diagrama de bloques para el proceso de automatización.

Tabla 2. Atributos de una acción a tomar en consideración.

Tipo de dato	Nombre	Descripción
VARCHAR(45)	nombre	Descripción corta de la acción, utilizada para identificación dentro de la interfaz junto a la clave.
VARCHAR(110)	descripcion	Descripción amplia de la acción.
VARCHAR(20)	objetivo	Objetivo de la acción al realizarse: modificar, trasladar, evitar o aceptar el riesgo.
VARCHAR(110) VARCHAR(45)	categoria control	Categoría y control de la acción conforme a la ISO 27001. [9]

la generación de riesgos y acciones solo se toman en cuenta atributos específicos en texto y enteros de las tablas de riesgos y acciones.

La Tabla 1 contiene los valores de los riesgos a tomar en consideración [3].

En la Tabla 2 se muestran los valores a considerar para las acciones a generar [3].

3.1 Diagrama de bloques

Se contemplan seis procesos para la generación automática, empezando por la preparación de la base de datos hacia un dataframe y teminando con la generación de los VARCHAR en lenguaje natural de las acciones. (véase Fig. 2)

3.2 Preparación de base de datos

Para generar los riesgos y las acciones se necesita generar un texto a partir de otro, es decir, implementar un modelo de procesamiento de lenguaje natural.

Teniendo en cuenta la estructura de la base de datos poblada por usuarios gestores, en el paso de extracción de características, se necesita de una librería específica para análisis de datos, ya que todos los registros se utilizarán para hacer un ajuste del modelo de lenguaje, y con ello, tomar de referencia todos los proyectos y riesgos que

manualmente reconocieron todos los gestores de riesgos que utilicen la aplicación. Esta librería del lenguaje de programación Python llamada Pandas está diseñada específicamente para la manipulación y el análisis de datos. Pandas se utiliza ampliamente para la manipulación de datos. Este término engloba los métodos de transformación de datos no estructurados para hacerlos utilizables [10].

La base de datos relacional se necesita convertir en un objeto dataframe que utilice la librería. Un dataframe es una estructura de datos bidimensional etiquetada con columnas de tipos potencialmente diferentes. Se puede pensar en él como una hoja de cálculo o una tabla SQL, o como un dictado de objetos Series. Tienen funciones para manipulación de datos como filtrado, agrupamiento, agregación y transformación de datos, [11] haciendo más fácil de utilizar y potentes en comparación con SQL puro. Ya que el framework back-end utilizado es Flask, utilizar esta librería da ventajas al momento de la implementación con el código desarrollado.

El dataframe deberá tener las tablas y los atributos descritos anteriormente para la generación automática, es decir, no es necesario tener todas las tablas adicionales, como usuarios o activos.

3.3 Fine Tuning de un modelo grande de lenguaje

El procesamiento de lenguaje natural permite que una computadora reconozca, comprenda y genere texto, combinando lingüística computacional con modelado estadístico [12]. Con dicha tecnología, se han desarrollado inteligencias artificiales generativas, llegando a la generación de texto. La cual funciona utilizando algoritmos y modelos lingüísticos para procesar los datos de entrada y generar el texto de salida. Consiste en entrenar modelos de inteligencia artificial con grandes conjuntos de datos de texto para aprender patrones, gramática e información contextual.

El núcleo de la generación de texto son los modelos lingüísticos, como GPT (Generative Pre-trained Transformer). Estos modelos emplean técnicas de aprendizaje profundo, en concreto redes neuronales, para comprender la estructura de las frases y generar textos coherentes y contextualmente relevantes [13].

Un riesgo consiste en una descripción del efecto de incertidumbre que puede suceder al proyecto, y de los subfactores para el cálculo de sus umbrales de probabilidad e impacto, por lo que se consiste en texto y números. Para generar la descripción, la amenaza, la vulnerabilidad y el nombre del riesgo es necesario utilizar procesamiento de lenguaje natural, específicamente un LLM (Modelo grande de lenguaje) que tome de referencia el contexto de la aplicación y de la gestión de riesgos de software.

Por lo que es necesario implementar un LLM entrenado mediante Fine Tuning para el contexto específico de la gestión de riesgos.

Fine Tuning. El Fine Tuning es el proceso de tomar un modelo preentrenado (específicamente un LLM) y entrenarlo más en un conjunto de datos específico. Ofrece ventajas considerables, como la reducción de los gastos de cálculo y la posibilidad de aprovechar sin necesidad de construir uno desde cero [14].

Para ello, se utiliza la librería Transformers, parte de la plataforma HuggingFace. Transformers permite acceder a una amplia colección de modelos preentrenados para diversas tareas. Existen distintos tipos de acercamientos al Fine Tuning, el más

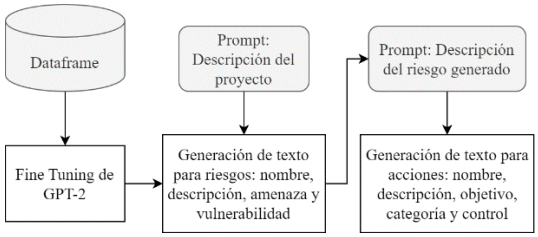

Fig. 3. Proceso de generación de texto para riesgos y acciones.

apropiado es el Fine Tuning de dominio específico, que trata de adaptar el modelo para que comprenda y genere texto específico de un sector concreto. El modelo se pone a punto en un conjunto de datos compuesto por texto del dominio de destino (gestión de riesgos) para mejorar su contexto y su conocimiento de las tareas específicas [14].

Se puede llevar a cabo mediante un Fine Tuning supervisado que requiere un conjunto de datos etiquetado [14].

Con la librería Transformers, se puede utilizar el LLM gpt2-large-bne, un modelo de GPT2 previamente entrenado con un corpus en español con la tarea específica de generación de texto [15].

Tras tener el dataframe y el conjunto de datos etiquetado con las descripciones de los riesgos, sus amenazas, vulnerabilidades y nombres, el paso siguiente es utilizar un tokenizer para preparar dichos datos para ser analizado por el modelo. Es separar las oraciones registradas en el dataframe en palabras, estas palabras son llamadas tokens, y todo LLM trabaja con ellas. Este proceso se conforma de limpiar las oraciones con las que se entrenará al modelo, como la eliminación de caracteres no deseados o de *stop words*, palabras que son muy comunes en el lenguaje natural como “el”, “para” o “y”, es decir, conectores o preposiciones [11, 16].

El Fine Tuning suele ser un proceso largo que debe repetirse con distintos parámetros: tasas de aprendizaje, los tamaños de lote y el número de épocas de entrenamiento para identificar la mejor configuración para conseguir salidas lo más cercanas posible a la redacción de un gestor de riesgos [14].

3.4 Generación de texto para riesgos y acciones

Tras el entrenamiento y contextualización del LLM, se deben introducir prompts específicos para cada texto. Este proceso se utiliza para la generación de textos tanto de riesgos como de acciones. En el caso de la generación de texto para riesgos, el prompt debe contener la descripción del proyecto introducida manualmente por el gestor de riesgos, y especificar que la salida tenga los datos necesarios: nombre, descripción, amenaza potencial y vulnerabilidad. Mientras que, para la generación de acciones, el prompt deberá depender de la descripción generada anteriormente de cada riesgo y describir los datos necesarios para la salida: nombre, descripción, objetivo,

categoría y control; por lo que la generación de texto será un proceso secuencial, como se muestra en la Figura 3.

Se puede utilizar de misma forma el LLM ajustado para elegir los activos asociados al riesgo recién generado mediante un prompt específico. La implementación de la lista de los activos depende enteramente del back-end al manejarse como objetos.

3.5 Valoración automática de riesgos generados

Utilizando el mismo dataframe que contiene los datos de todos los subfactores de probabilidad e impacto de la base de datos, lograr una valoración automática de los riesgos conlleva distintos pasos.

Para ello, se debe concatenar los datos de texto generados por el LLM para cada riesgo en el dataframe, y vectorizarlo. La vectorización de texto es una forma clásica de convertir los datos de entrada de su formato bruto (es decir, texto) en vectores de números reales, que es el formato que admiten los modelos de Machine Learning. La vectorización es un paso en la extracción de características. La idea es obtener algunas características distintivas del texto para que un modelo se entrene, convirtiendo el texto en vectores numéricos [17]. La técnica de vectorización a utilizar es TF-IDF (Term Frequency-Inverse Document Frequency).

TF-IDF. Se define como el cálculo de la relevancia de una palabra en una serie o corpus en relación con un texto. El significado aumenta proporcionalmente al número de veces que aparece una palabra en el texto, pero se compensa con la frecuencia de palabras en el corpus (conjunto de datos) [18]. Las palabras de un documento de texto se transforman en números de importancia.

Consta de dos partes: TF e IDF:

TF (Term Frequency). Una puntuación de frecuencia normalizada, el TF de un término o palabra es el número de veces que dicho término aparece en un documento en comparación al número total de palabras en el documento [19].

IDF (Inverse Document Frequency). Refleja la proporción de documentos en el corpus que contienen el término. Las palabras que tienen un pequeño porcentaje de documentos (como pueden ser términos técnicos) reciben valores más altos que palabras comunes [19].

El TF-IDF se consigue al multiplicar ambos valores, y significa que un término es importante cuando aparece bastante en un documento y poco en otros documentos del corpus. Por lo que la frecuencia medida por TF se equilibra con la rareza entre documentos que mide IDF, consiguiendo una puntuación numérica que refleja la importancia de un término [19].

Para la predicción de los valores numéricos de los subfactores, se utiliza un modelo de clasificación. La clasificación es un método de Machine Learning supervisado en el que el modelo intenta predecir la etiqueta correcta de unos datos de entrada.

En la clasificación, el modelo se entrena completamente con los datos de entrenamiento y, a continuación, se evalúa con los datos de prueba antes de utilizarlo

Tabla 3. Prompts utilizados para la generación de texto en GPT2.

Prompt	Dato
Dada la siguiente descripción y contexto del proyecto, describa el principal riesgo y sugiera una acción preventiva: Descripción del proyecto: proyecto.descripcion Contexto: El proyecto implica: proyecto.activos	Riesgo
Describe un riesgo potencial que pueda afectar a este proyecto de software, dicho riesgo debe contener un nombre, una descripción, una amenaza y una vulnerabilidad	Descripción del riesgo Amenaza Vulnerabilidad
Sugiere una acción preventiva para la mitigación de de este riesgo, dicha acción debe contener un nombre, descripción, objetivo y una categoría y control especificados por: controlesISO27001	Acción Descripción de acción Objetivo Categoría y control

para realizar predicciones con nuevos datos no vistos [20]. Por lo que el dataframe se necesita dividir en un dataset de entrenamiento y en un dataset de pruebas.

Clasificación Random Forest. En una clasificación de Random Forest, se crean múltiples árboles de decisión a partir de un subconjunto seleccionado aleatoriamente del dataset de entrenamiento. Es una técnica de aprendizaje por conjuntos diseñada para mejorar la precisión y robustez de la clasificación. Cada árbol de decisión del bosque aleatorio se construye utilizando un subconjunto de los datos de entrenamiento y un subconjunto aleatorio de características que introducen diversidad entre los árboles, lo que hace que el modelo sea más robusto [21].

Para utilizar dicho método de clasificación, se usa la librería scikit-learn, una librería de código abierto de Machine Learning para Python.

Se puede iniciar con una instancia del clasificador con parámetros por defecto, para después ajustar cada parámetro obteniendo más precisión con las predicciones.

Una forma de evaluar el modelo es con precisión, se revisa manualmente si los datos numéricos de los riesgos concuerdan con los datos del dataset de entrenamiento y se cuenta cuantos datos fueron correctamente predichos, es decir, cotejar los valores de cada subfactor de los riesgos de la base de datos con los que se generan, y ajustar conforme a los resultados, llevando una mejora continua del modelo.

Tabla 4. Comparación de riesgos manuales y automáticos.

Ejemplo	Descripción manual	Descripción LLM	Probabilidad manual	Probabilidad predicha	Impacto manual	Impacto predicho
E-commerce	Posible ataque DDoS que afecte la disponibilidad del servidor web.	El servidor de la plataforma de e-commerce podría verse comprometido por ataques DDoS, afectando su disponibilidad.	7.7	6.9	8.2	7.3
Aplicación financiera	Vulnerabilidad en la autenticación que podría permitir el acceso no autorizado a cuentas de usuarios.	Una debilidad en el sistema de autenticación podría exponer cuentas de usuario a accesos no autorizados.	5.2	4.4	9.2	8
Sistema de gestión documental	Pérdida de datos por falla en el sistema de respaldo.	Un fallo en el sistema de respaldo podría resultar en la pérdida de documentos críticos.	3.1	3.8	7.1	7.4

Tabla 5. Comparación de acciones manuales y automáticas.

Ejemplo	Descripción manual	Desccripción GPT2
E-commerce	Implementar un servicio de mitigación de DDoS	Configurar un servicio de protección contra DDoS y monitorear el tráfico de red constantemente.
Aplicación financiera	Implementar autenticación 2FA	Activar autenticación de dos factores y reforzar los controles de acceso.
Sistema de gestión documental	Implementar estrategia de copias de seguridad para redundancia (RAID)	Asegurar que el sistema de respaldo tenga redundancia y se verifique regularmente

4. Resultados

Con un dataset de entrenamiento mínimo que consta de tres proyectos de software, con nueve riesgos identificados, cinco activos y tres acciones de prevención cada uno, se entrenó a los modelos a forma de ensayo, sin implementación directa en la aplicación web. Dichos registros fueron completados manualmente. El modelo de clasificación fue entrenado con los parámetros por defecto de la librería scikit-learn, y a su vez el Fine Tuning fue realizado en una Notebook de Google Colab. Para la generación del riesgo con una acción preventiva, habiendo seleccionado los activos, se utilizaron los siguientes prompts, tal como se visualizan en la Tabla 3.

En la Tabla 4 se muestra la comparación entre riesgos valorados manualmente por el equipo de desarrollo y los riesgos predichos en combinación de ambos modelos.

En la Tabla 5 se muestra la comparación entre acciones preventivas descritas manualmente por el equipo de desarrollo y las acciones predichas por el LLM.

De forma manual, el proceso de valoración de riesgos, asignación de activos, e identificación de una acción por cada riesgo toma al gestor de riesgos aproximadamente dos horas por cada riesgo, considerando que la herramienta tenga en su base de datos todos los activos del proyecto de software [3]. La automatización derivada de los modelos, una vez que se dispone de una base de datos robusta y poblada con el aproximado de 130 usuarios gestores [3], permite generar automáticamente riesgos con sus activos asociados y una acción preventiva sugerida de calidad. Dicha cantidad se puede considerar un buen punto de partida, sin embargo, lo realmente fundamental es la calidad de los riesgos registrados por los gestores [22].

Con una cantidad significativa de riesgos generados, los proyectos de software pueden comenzar con un conjunto de riesgos predefinidos que son comunes y recurrentes, facilitando la identificación temprana de problemas potenciales. Esto permite que el gestor de riesgos se concentre en casos muy específicos que los modelos automáticos no logren predecir, maximizando su tiempo y experiencia en áreas donde su intervención es más crítica.

5. Conclusiones

La automatización de la gestión de riesgos mediante modelos de inteligencia artificial no solo mejora la eficiencia y la efectividad operativa, sino que también capacita a los gestores de riesgos y departamentos dentro de las organizaciones a tomar decisiones estratégicas más informadas. Utilizar modelos de inteligencia artificial en la gestión de riesgos conlleva ventajas al poder analizar grandes volúmenes de datos y optimizar recursos de las organizaciones como el tiempo y el propio personal necesario para la labor de la gestión de riesgos en sus proyectos de software. Además, dicha implementación de los modelos también auxilia para proyectos educativos, para que alumnos puedan visualizar con dicha experiencia de los gestores previos y con el entrenamiento propuesto, las acciones para la prevención de los riesgos, parte fundamental en cualquier etapa del ciclo de vida del desarrollo incluyendo distintas categorías de proyectos, dando un mejor entendimiento y optimizando el aprendizaje.

En la parte técnica de los modelos existen áreas de mejora, al utilizar un LLM más reciente conllevando un precio de licencia como puede ser GPT-3. Además, el aprendizaje supervisado es una tarea que depende de los desarrolladores que cotejan los resultados de las predicciones, con un enfoque diferente de aprendizaje no supervisado dicha situación se superaría.

Se contempla como trabajo a futuro la implementación de los modelos y validarlos con la opinión de al menos un gestor de riesgos experimentado con incidencias reales en distintos proyectos de software, quien puede dar una retroalimentación empírica sobre los resultados generados.

Finalmente, también es importante considerar la protección de los datos de entrenamiento, abordando cuestiones éticas y de privacidad hacia los usuarios de la aplicación web derivadas de las nuevas funcionalidades propuestas por este documento.

Agradecimientos. Los resultados de este trabajo se desarrollaron en el marco del proyecto de investigación: Aplicación móvil para el apoyo docente en los procesos enseñanza - aprendizaje de la lectura mediante técnicas de gamificación y Machine Learning con número de registro asignado por el SIP: 20242235. Desarrollado en Instituto Politécnico Nacional.

Referencias

1. Information Technology: Security techniques, Information security management systems, Overview and vocabulary ISO/IEC 27000:2018. https://standards.iso.org/ittf/PubliclyAvailable Standards/index.html (2018)
2. Project Management Institute: Practices Standard for Project Risk Management. Project Management Institute (2009)
3. Alarcón, A., Castro, G., Franco, L.E.: Trabajo Terminal: Prototipo de Sistema de Gestión de Riesgos para Proyectos de Software. Escuela Superior de Cómputo, Instituto Politécnico Nacional (2024)
4. Chan, A.: Can AI be used for Risk assessments? https://www.isaca.org/resources/news-and-trends/industry-news/2023/can-ai-be-used-for-risk-assessments (2023)
5. Boultwood, B.: How Artificial Intelligence will change qualitative Risk Management. https://www.garp.org/risk-intelligence/technology/how-artificial-intelligence-will-change-qualitative-risk-management (2020)
6. Giudici, P., Raffinetti, E.: Explainable AI methods in cyber risk management. Qual Reliab Eng Int., 38(3), pp. 1318–1326 (2022)
7. OWASP: Risk Rating Methodology. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology (2022)
8. Muñoz, J., Molero-Castillo, G., Benitez-Guerrero, E.: Método de fusión de datos de fuentes heterogéneas para mantener la consistencia de datos. Research in Computing Science Issue, 139 pp. 33–46 (2017)
9. Information security, cybersecurity and privacy protection. Information security management systems – Requirement ISO/IEC 27001:2022. https://standards.iso.org/ittf/PubliclyAvailable Standards/index.html (2022)
10. NVIDIA: Pandas Python, https://www.nvidia.com/en-us/glossary/pandas-python/ (2024)
11. Pandas Documentation: https://pandas.pydata.org/docs/user_guide/dsintro.html (2024)
12. Natural Language Processing, https://www.ibm.com/topics/natural-language-processing (2024)

13. IBM: Text Generation, DataCamp. https://www.datacamp.com/blog/what-is-text-generation (2023)
14. DataCamp: Fine Tuning LLMs. https://www.datacamp.com/tutorial/fine-tuning-large-language-models (2024)
15. HuggingFace: https://huggingface.co/PlanTL-GOB-ES/gpt2-large-bne (2023)
16. Soto, D., et al.: Clasificación bi-clase de canciones infantiles aplicando inteligencia artificial y procesamiento de lenguaje natural. Research in Computing Science Issue, 161(6), pp. 6 (2022)
17. Vectorization techniques: https://neptune.ai/blog/vectorization-techniques-in-nlp-guide (2023)
18. Understanding TF-IDF: Geeks for Geeks. https://www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/ (2023)
19. TF-IDF: LearnDataSci. https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/ (2024)
20. DataCamp: Classification in Machine Learning https://www.datacamp.com/blog/classification-machine-learning (2024)
21. Geeks for Geeks: Random Forest Classifier using Scikit-learn. https://www.geeksforgeeks.org/random-forest-classifier-using-scikit-learn/ (2024)
22. OpenAI: Fine-tuning. https://platform.openai.com/docs/guides/fine-tuning/analyzing-your-fine-tuned-model (2024)

Electronic edition
Available online: http://www.rcs.cic.ipn.mx

http://rcs.cic.ipn.mx

Centro de Investigación
en Computación