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

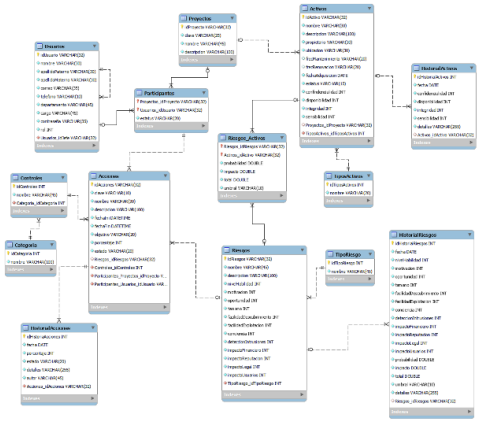

Fig. 1. Diagrama relacional de la base de datos de la aplicación.

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.

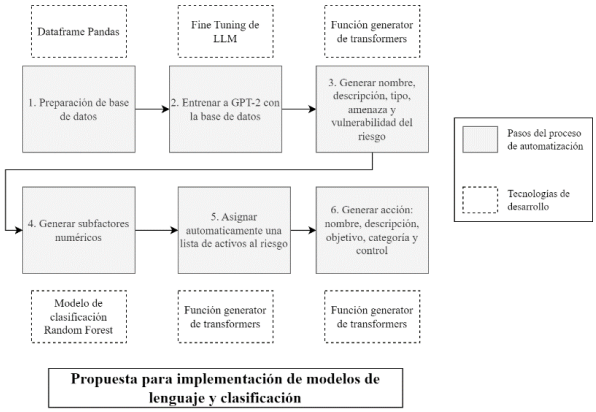

Fig. 2. Diagrama de bloques para el proceso de automatización.

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

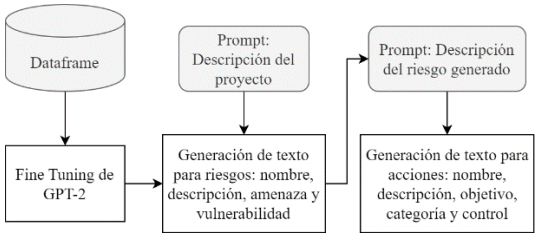

Fig. 3. Proceso de generación de texto para riesgos y acciones.

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)