UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

Aplicación de modelos de aprendizaje automático para la identificación de un indicador de riesgo temprano en los procesos de Subasta Inversa Electrónica en Ecuador

Valeria Melisa Guerrero Cisneros

Ingeniería en Ciencias de la Computación

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
Ingeniera en Ciencias de la Computación

Quito, 12 de mayo de 2025

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA

Aplicación de modelos de aprendizaje automático para la identificación de un indicador de riesgo temprano en los procesos de Subasta Inversa Electrónica en Ecuador

Valeria Melisa Guerrero Cisneros

Daniel Riofrío, PhD

Quito, 12 de mayo de 2025

© DERECHOS DE AUTOR

Nombres y apellidos: Valeria Melisa Guerrero Cisneros

Código: 00322205

Cédula de identidad: 1725889859

Lugar y fecha: Quito, 12 de mayo de 2025

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

RESUMEN

Esta investigación aborda la detección temprana de riesgos de corrupción en la contratación pública ecuatoriana, mediante el análisis de datos estructurados y documentos extraídos de los procesos de Subasta Inversa Electrónica (SIE) correspondientes a los años 2020, 2022 y 2023. Se implementó un pipeline de procesamiento que incluyó la selección de características numéricas y categóricas, la extracción y limpieza de texto desde archivos en formato `.ushay`, y la representación semántica de los contratos mediante embeddings de Doc2Vec. Inicialmente, se planteó un experimento de clasificación multiclase basado en la segmentación del indicador `sie_ic_promedio` en cuartiles; sin embargo, ante los bajos desempeños obtenidos, se reformuló el problema como una tarea de clasificación binaria, estableciendo un umbral óptimo de 0.56. Los experimentos de clasificación combinaron variables estructuradas y representaciones textuales, aplicando también estrategias de selección de fragmentos relevantes basadas en similitud de coseno. Los resultados muestran que es posible identificar patrones de riesgo en etapas tempranas de los procesos de contratación. Como líneas futuras, se propone incorporar variables adicionales provenientes de los archivos *proceso.xml* y explorar mecanismos formales para identificar procesos dirigidos a partir de las especificaciones técnicas y condiciones de licitación.

ABSTRACT

This research addresses the early detection of corruption risks in Ecuadorian public procurement by analyzing structured data and extracted documents from Reverse Electronic Auctions (SIE) conducted in 2020, 2022, and 2023. A processing pipeline was implemented, including feature selection for numerical and categorical variables, text extraction and cleaning from `.ushay` data files, and semantic representation of contracts using Doc2Vec embeddings. Initially, a multiclass classification experiment was carried out by segmenting the `sie_ic_promedio` indicator into quartiles; however, due to low performance, the problem was reformulated as a binary classification task, establishing an optimal threshold of 0.56. Classification experiments were implemented. These experiments combined structured features and textual vectors. Fragment selection strategies based on cosine similarity were also implemented. The results demonstrate that it is possible to detect risk patterns at early stages of procurement processes. Future research directions include incorporating additional variables from *proceso.xml* files and developing formal mechanisms to identify directed procurement processes based on technical specifications and bidding conditions.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

INTRODUCCIÓN

La corrupción en Ecuador ha tenido un gran impacto en la administración pública, afectando la estabilidad económica y social del país. Según un estudio de la Revista Latinoamericana de Ciencias Sociales y Humanidades, la corrupción afecta al PIB, la inversión extranjera directa y sectores importantes como la educación, la salud y el empleo (Santos Alava et al., 2024). Entre 180 países, Ecuador está en el puesto 105 del Índice de Percepción de la Corrupción (IPC) en 2024, con un puntaje de 32 sobre 100. En esta escala, 0 representa el nivel más alto de corrupción y 100 el nivel más bajo. Desde 2023, Ecuador ha bajado 2 puntos, lo que demuestra que el problema no ha mejorado en estos años. Desde 2012, el desempeño del país en este índice ha tenido avances y retrocesos, pero el puntaje actual muestra que es urgente implementar medidas para mejorar la transparencia y fortalecer las instituciones públicas (Transparency International, 2024). En este contexto, surge una interrogante fundamental: ¿existen patrones en la información de los procesos de contratación pública que permitan predecir el riesgo de corrupción?

Una iniciativa que aborda esta problemática es Kapak, una herramienta creada por la Universidad San Francisco de Quito y con el apoyo de la Cooperación Técnica Alemana (GIZ) en 2020, con el objetivo de combatir la corrupción en la contratación pública. Kapak utiliza datos extraídos automáticamente del Sistema Oficial de Contratación Pública del Ecuador (SOCE) y genera indicadores clave basados en modalidades como la Subasta Inversa Electrónica (SIE) y el Giro Específico del Negocio (GIE) (Universidad San Francisco de Quito, 2023). Sin embargo, una de las áreas menos exploradas es la identificación temprana de patrones de riesgo durante la fase inicial de los procesos de contratación. Esto resulta crucial para la creación de un indicador de riesgo temprano, ya que permite abordar las irregularidades antes de que escalen y causen mayores impactos. Para abordar esta problemática, esta

investigación aplica modelos de aprendizaje automático para analizar los procesos de contratación pública a partir de los datos de la Subasta Inversa Electrónica (SIE) correspondientes a los años 2020, 2022 y 2023. El análisis se centra en los documentos generados en cada proceso de contratación, con el objetivo de identificar patrones asociados a riesgos de corrupción. Para ello, se utilizan técnicas de representación de texto como Doc2Vec, que convierten los documentos en vectores numéricos preservando relaciones semánticas relevantes. Posteriormente, estos vectores sirven como insumo para la construcción de modelos de clasificación. Esta aproximación busca contribuir al fortalecimiento de mecanismos de control preventivo en la contratación pública.

Una línea de investigación relevante en este campo es el trabajo de García Rodríguez et al. (2022), quienes analizan la aplicación de algoritmos de aprendizaje automático para detectar prácticas colusorias en procesos de contratación pública. En su estudio, se utilizan seis conjuntos de datos provenientes de Brasil, Italia, Japón, Suiza y Estados Unidos, enfocados en licitaciones con colusión comprobada. Para mejorar la capacidad predictiva, los autores incorporan variables de cribado calculadas a partir de los valores de las ofertas. Estas variables permiten capturar patrones anómalos sin necesidad de información privada de los licitadores, lo que representa una ventaja importante frente a los métodos tradicionales.

Los resultados obtenidos muestran que los métodos de ensamble, como Random Forest, Extra Trees, AdaBoost y Gradient Boosting, ofrecen el mejor desempeño en la detección de colusión. Asimismo, la inclusión de variables de cribado, entendidas como indicadores estadísticos derivados de las ofertas que permiten identificar patrones anómalos sin necesidad de información confidencial, contribuye a mejorar la precisión de los modelos y a reducir las tasas de falsos positivos y negativos, incluso en escenarios de información limitada. Este estudio demuestra que los algoritmos de aprendizaje automático representan una herramienta

prometedora para fortalecer los mecanismos de control en la contratación pública, especialmente en contextos donde el acceso a información detallada es restringido.

Otro ejemplo relevante en el contexto latinoamericano es el caso de México, donde se aplicaron técnicas de aprendizaje automático para detectar posibles casos de corrupción en la contratación pública. En este estudio, se analizaron datos del sistema CompraNet, examinando contratos adjudicados entre 2013 y 2020. Se usaron listas oficiales de empresas sancionadas por el gobierno mexicano, para etiquetar los contratos como "corruptos" o "no corruptos". Esto permitió construir un conjunto de datos sólido para el análisis. Random Forest es un algoritmo de aprendizaje automático basado en un conjunto de árboles de decisión, los cuales se construyen de forma jerárquica mediante divisiones sucesivas de los datos. Cada árbol en el modelo se entrena con un subconjunto aleatorio de los datos, permitiendo que el conjunto de árboles tome decisiones más robustas. Este enfoque ayuda a mejorar la precisión del modelo y a evitar el sobreajuste (Aldana, Falcón-Cortés, & Larralde, 2022).

El proceso incluyó la limpieza y preparación de los datos, eliminando registros incompletos y duplicados. Luego, se seleccionaron 19 variables clave, como el número de participantes, el monto adjudicado y relaciones entre compradores y proveedores. Para manejar el desequilibrio de los datos, se dividieron en subconjuntos balanceados y se entrenaron clasificadores independientes, combinando sus resultados mediante un sistema de votación. Este enfoque permitió identificar patrones asociados a la corrupción en contratos públicos. El modelo mostró buenos resultados y se optimizó con una curva ROC, lo que permitió minimizar falsos positivos. Además, se demostró que las variables seleccionadas eran efectivas para detectar riesgos de corrupción (Aldana, Falcón-Cortés, & Larralde, 2022).

PROCESAMIENTO DE LENGUAJE NATURAL

El Procesamiento de Lenguaje Natural (PLN) ha transformado la manera en que los sistemas computacionales interactúan con el lenguaje humano, permitiendo tareas como la clasificación automática de textos, la detección de spam o la recuperación de información relevante. Una característica común en la mayoría de los algoritmos de aprendizaje automático aplicados a textos, como K-Means o regresión logística, es la necesidad de que las entradas sean representadas como vectores de características de longitud fija.

Para ello, han surgido diversas técnicas de representación textual que varían en complejidad y capacidad de capturar las propiedades semánticas y sintácticas del lenguaje. En un primer nivel de representación se encuentran enfoques simples como Bag-of-Words (BOW), donde un texto se representa únicamente a partir del conteo de la frecuencia de las palabras que contiene. Posteriormente, métodos como TF-IDF (Term Frequency–Inverse Document Frequency) incorporan ponderaciones que permiten distinguir términos relevantes dentro de un corpus. A medida que el PLN ha evolucionado, se han desarrollado representaciones distribuidas, basadas en la hipótesis distribucional, la cual sostiene que palabras que aparecen en contextos similares tienden a compartir significados (Jurafsky & Martin, 2025).

Técnicas como Word2Vec han permitido mapear palabras a vectores densos que preservan relaciones semánticas en el espacio vectorial. Finalmente, para superar las limitaciones de las representaciones a nivel de palabra, surgieron métodos como Doc2Vec, propuesto por Le y Mikolov (2014), los cuales permiten obtener vectores de longitud fija para fragmentos de texto más largos, tales como frases, párrafos o documentos completos, incorporando tanto la semántica como el orden de las palabras. En esta sección se presentarán

en detalle los fundamentos teóricos de estas técnicas, sus principales características y su importancia en la representación de textos en el ámbito del PLN.

TF-IDF

En el contexto del procesamiento de lenguaje natural y la recuperación de información, la Frecuencia Inversa de Documentos (IDF, por sus siglas en inglés) es una medida que evalúa qué tan importante es un término dentro de un corpus de documentos. Formalmente, el IDF de un término t_i se define como:

$$IDF\ (t_i) = \log\ (\frac{N}{n_i})$$

donde N representa el número total de documentos en la colección y n_i indica el número de documentos que contienen el término t_i. El modelo TF-IDF (Term Frequency–Inverse Document Frequency) combina esta medida con la frecuencia de aparición de un término dentro de un documento específico, ponderando así la importancia de un término no solo por su presencia en un documento, sino también por su rareza en el corpus general. La fórmula general de TF-IDF es:

$$TF - IDF\ (t_i, d) = TF\ (t_i, d)\ \times\ IDF\ (t_i)$$

donde $TF\ (t_i, d)$ es la frecuencia del término t_i en el documento d. Esta combinación permite destacar términos que son frecuentes en un documento pero poco comunes en el conjunto de documentos, lo cual resulta útil para tareas de clasificación, búsqueda y análisis de textos (Robertson, 2004).

Word2Vec

Word2Vec aprende representaciones vectoriales de palabras mediante un enfoque predictivo basado en contexto. En el modelo Skip-gram, se busca maximizar la

probabilidad logarítmica promedio de observar una palabra w_o dado una palabra w_I, utilizando como objetivo la siguiente función softmax:

$$P(w_O|w_I) = \frac{exp\left(v'_{w_O}{}^T v_{w_I}\right)}{\sum_{w=1}^{W} exp\ (v'_{w_O}{}^T v_{w_I})}$$

donde v'_{w_O} y v_{w_I} son los vectores de representación de entrada y salida para las palabras W_O y W_I respectivamente, La suma del denominador se realiza sobre todas las palabras del vocabulario, cuyo tamaño total se denota como W. Si bien esta formulación permite modelar de manera efectiva las relaciones semánticas entre palabras, presenta una limitación significativa desde el punto de vista computacional. Para cada par de palabras observado durante el entrenamiento, se requiere calcular el valor de la función softmax, lo cual implica evaluar una suma sobre todo el vocabulario, hace que la implementación directa de la función softmax sea inviable en la práctica para grandes corpus. Debido al alto costo computacional de aplicar softmax sobre todo el vocabulario, se emplean técnicas de aproximación como Negative Sampling o Hierarchical Softmax para optimizar el proceso. En particular, el Skip-gram con muestreo negativo (SGNS) reformula el problema como una clasificación binaria, en la que se busca distinguir entre palabras de contexto reales y muestras negativas generadas aleatoriamente. Este enfoque permite que palabras que comparten contextos similares se agrupen en el espacio vectorial, facilitando la identificación de relaciones semánticas implícitas (Mikolov et al., 2013).

La transición de modelos basados en frecuencias a métodos de aprendizaje profundo, como Word2Vec, sentó las bases para el desarrollo de embeddings contextuales más avanzados. Un ejemplo de esta evolución es BERT (Bidirectional Encoder Representations from Transformers), que introduce un enfoque bidireccional para

comprender el significado de una palabra tomando en cuenta tanto el contexto anterior como el posterior (Jurafsky & Martin, 2025).

Doc2Vec

Doc2Vec es una extensión de Word2Vec que permite aprender representaciones vectoriales para documentos de cualquier longitud, conservando tanto la información semántica como el contexto. A diferencia de enfoques como Bag of Words (BoW), que ignoran el orden de las palabras, Doc2Vec genera un vector denso que representa el contenido de un documento en su totalidad. Se basa en dos enfoques principales: Distributed Memory (PV-DM), que actúa como una memoria global del documento, y Distributed Bag of Words (PV-DBOW), que predice palabras dentro del documento sin considerar su orden (Le & Mikolov, 2014).

Además, la arquitectura Transformer ha revolucionado el PLN con un modelo basado en autoatención (self-attention), lo que permite procesar palabras en paralelo y capturar relaciones semánticas a larga distancia. A diferencia de modelos como Word2Vec, que generan representaciones estáticas de las palabras, los Transformers crean embeddings contextuales que varían según la oración. Dentro de esta arquitectura, BERT representa un gran avance al usar este enfoque bidireccional, lo que mejora la capacidad de los modelos para comprender y generar texto con mayor precisión (Jurafsky & Martin, 2025).

Distributed Memory (PV-DM)

El modelo Distributed Memory (PV-DM) extiende la arquitectura de Word2Vec para aprender representaciones vectoriales de párrafos o documentos completos. En este enfoque, cada documento es asignado a un vector único, el cual actúa como una memoria que captura

información semántica global ausente en las palabras locales del contexto. Durante el entrenamiento, el vector del párrafo se concatena o promedia con los vectores de palabras en un contexto de ventana fija para predecir la siguiente palabra en la secuencia. De este modo, tanto los vectores de palabras como los vectores de párrafos son optimizados conjuntamente a través de retropropagación, utilizando métodos como descenso de gradiente estocástico. Formalmente, la arquitectura modifica la función tradicional de Word2Vec incorporando un vector adicional proveniente de la matriz de párrafos, D; el cual participa en la construcción de la representación, h, utilizada para la predicción. Esta integración permite que el vector del documento sirva como un "tema latente" que complementa la información local aportada por las palabras adyacentes. Una de las principales ventajas de PV-DM es que conserva parcialmente el orden de las palabras a través de la secuencia de contextos y, al mismo tiempo, incorpora conocimiento semántico de nivel más alto, haciendo que las representaciones resultantes sean más robustas para tareas de clasificación, recuperación de información y análisis de sentimientos (Le & Mikolov, 2014).

Distributed Bag of Words (PV-DBOW)

El modelo Distributed Bag of Words (PV-DBOW) representa una variante más sencilla y eficiente del esquema Paragraph Vector, enfocándose en reducir la complejidad del entrenamiento. A diferencia de PV-DM, en PV-DBOW se omite el uso del contexto de palabras como entrada, y se entrena directamente el vector del documento para predecir palabras seleccionadas aleatoriamente dentro del mismo texto. En cada iteración, se escoge una ventana y se plantea una tarea de clasificación donde el vector del párrafo debe predecir correctamente las palabras correspondientes. Esta arquitectura es conceptualmente similar al modelo Skip-gram de Word2Vec, en el sentido de que prioriza la predicción de múltiples objetivos a partir de una sola entrada.

Además, PV-DBOW requiere almacenar menos información, ya que solo conserva los vectores de los párrafos y los pesos de la capa softmax, sin necesidad de mantener representaciones individuales de cada palabra. Aunque de forma aislada PV-DBOW puede ofrecer un rendimiento inferior al de PV-DM en ciertas tareas, su combinación con este último suele mejorar de manera consistente el desempeño general en tareas de clasificación y recuperación de texto. De hecho, se ha demostrado que integrar las representaciones aprendidas por ambos modelos tiende a proporcionar resultados más robustos y generalizables en una amplia variedad de tareas de procesamiento de lenguaje natural (Le & Mikolov, 2014).

DESCRIPCIÓN DEL CONJUNTO DE DATOS

Análisis exploratorio de datos

La base de datos contiene 5710 registros, cada uno representa un proceso de contratación pública llevado a cabo en los años 2020, 2022 y 2023. Cada entrada del dataset contiene información detallada sobre los montos presupuestados, adjudicados y contratados, junto con una variedad de metadatos relacionados con el proceso de contratación, la entidad contratante y la comisión evaluadora. Los registros corresponden a procesos en diferentes etapas del procedimiento contractual, incluyendo estados como adjudicado, cancelado, desierto, en recepción, finalizado por mutuo acuerdo, suspendido, por adjudicar, entre otros.

Para el análisis exploratorio del dataset, se realizó una depuración inicial que incluyó la eliminación de columnas duplicadas, obteniendo un conjunto final de 67 variables (features). Estas variables abarcan tanto atributos numéricos como categóricos vinculados al desarrollo de la contratación pública, tales como el presupuesto referencial, el número de oferentes, los tiempos de ejecución, las características de los proveedores, y el objeto contractual, entendido como la descripción del bien o servicio que la entidad pública busca adquirir. Asimismo, se

incluyen una serie de indicadores de riesgo construidos para evaluar condiciones que podrían representar señales de alerta en los procesos analizados.

Dentro de este conjunto, se identificaron 14 indicadores de riesgo de corrupción diseñados para evaluar distintos aspectos críticos del proceso de Subasta Inversa Electrónica. Trece de estos indicadores se formulan a partir de reglas lógicas, umbrales normativos y combinaciones de variables relevantes, y se expresan como variables binarias: toman el valor de 1 cuando se detecta una condición asociada a un posible riesgo de corrupción o irregularidad, y 0 cuando dicha condición no está presente. Estos indicadores consideran aspectos como la falta de concurrencia de oferentes, plazos de entrega inusualmente cortos, ausencia de verificación de inhabilidades, presencia de contratos complementarios, posibles sobreprecios, entre otros.

El indicador restante es un índice compuesto de riesgo, calculado a partir de los 13 indicadores binarios. Este índice resume el comportamiento general de cada procedimiento en términos de cumplimiento con criterios de transparencia y competencia, mediante una escala continua que permite cuantificar y comparar el nivel de riesgo entre distintos procesos. En esta investigación, el índice compuesto se utiliza como variable objetivo y se encuentra representado en la base de datos bajo el nombre `sie_ic_promedio`. Su valor, expresado en una escala continua de 0 a 1, refleja el grado de cumplimiento con prácticas de transparencia y competencia: a mayor puntaje, menor riesgo estimado de corrupción. A continuación, se describen los indicadores seleccionados por su alta correlación positiva con el índice compuesto de riesgo de corrupción (superior a 0.68), con el objetivo de destacar aquellos que más contribuyen a su comportamiento. Los valores de correlación se presentan en la *Tabla 1*, donde se listan los 10 indicadores con mayor influencia sobre el índice compuesto.

Indicador	Correlación con el índice compuesto
sie_ic_indicador_22	0.7842
sie_ic_indicador_11	0.7830
sie_ic_indicador_25	0.6984
sie_ic_indicador_19	0.5378
sie_ic_indicador_04	0.4479
sie_ic_indicador_09	0.3808
sie_ic_indicador_05	0.2603
sie_ic_indicador_06	0.2306
sie_ic_indicador_15	0.1892
sie_ic_indicador_27	0.1762

Tabla 1: Top 10 de indicadores con mayor correlación con el índice compuesto de riesgo de corrupción

Indicadores relevantes para el cálculo del índice compuesto

Las definiciones y fórmulas que se presentan a continuación han sido extraídas del portal *Kapak: Transparencia en compras públicas* (Kapak, 2023).

Indicador 11: Porcentaje de procedimientos adjudicados sin concurrencia de oferentes

Este indicador identifica los procedimientos en los que únicamente participó un oferente, evidenciando una posible falta de competencia. La variable binaria asociada toma el valor 1 si el proceso tuvo un solo oferente y 0 en caso contrario. Su fórmula es:

$$Y_i = X1$$

donde Y_i representa el resultado para el proceso i, y $X1$ es una variable binaria basada en la condición $\mathrm{OF} = 1$ (Ofertas realizadas = 1).

Indicador 22: Porcentaje de procedimientos adjudicados a un único oferente habilitado para la negociación

Este indicador evalúa los procesos en los que solo un proveedor fue habilitado para negociar, limitando la competencia. Se asigna un valor de 1 cuando el procedimiento cumple esta condición y 0 cuando no. Su cálculo se basa en:

$$Y_i = X1$$

donde Y_i representa el resultado para el proceso i, y $X1$ se obtiene si el procedimiento cuenta con un único oferente habilitado.

Indicador 25: Número de procedimientos declarados desiertos posiblemente sin motivación

Este indicador contabiliza los procedimientos declarados desiertos sin contar con una resolución que justifique tal decisión. Se considera una señal de alerta sobre posibles anulaciones discrecionales. La fórmula es:

$$Y_i = X1$$

donde Y_i toma el valor de 1 si el procedimiento i contiene la resolución de declaratoria de desierto, y 0 si no la contiene. Se identifica esta resolución mediante búsqueda textual de la frase “Resolución declaratoria de desierto” en los archivos del proceso.

Feature selection

Es importante realizar un proceso de feature selection, ya que no todos los modelos de machine learning mejoran su rendimiento al utilizar una gran cantidad de variables. De hecho, algunas características podrían no aportar valor al modelo e incluso introducir ruido, afectando negativamente la precisión de las predicciones. Por ello, en este estudio se experimentó con distintas técnicas de selección de características, con el objetivo de identificar aquellas variables que guardan una mayor relación con la variable objetivo `sie_ic_promedio`, fundamental para el desarrollo del indicador de riesgo temprano.

Correlación de Pearson

El coeficiente de correlación de Pearson es una medida estadística que evalúa el grado de asociación lineal entre dos variables cuantitativas. Su propósito principal es determinar en

qué medida los cambios en una variable están relacionados con los cambios en otra, bajo el supuesto de que la relación entre ellas es de naturaleza lineal. Esta medida es ampliamente utilizada en diversas disciplinas como la estadística, la economía, la psicología y las ciencias sociales para analizar relaciones entre variables y construir modelos predictivos. Desde una perspectiva matemática, el coeficiente de correlación de Pearson está definido dentro del rango de valores [−1, 1] Un valor de r_{xy}= 1 indica una correlación positiva perfecta, es decir, cuando una variable aumenta, la otra también lo hace en proporción exacta. En contraste, un valor de $r_{xy} = -1$ señala una correlación negativa perfecta, lo que significa que cuando una variable aumenta, la otra disminuye en la misma proporción. Finalmente, si $r_{xy} = 0$, esto implica que no existe una relación lineal entre las dos variables analizadas (Camacho Martínez Vara de Rey, s.f.).

El coeficiente de correlación de Pearson se calcula mediante la siguiente ecuación:

$$r_{xy} = \frac{\sum(X - \overline{X})(Y - \overline{Y})}{\sqrt{\sum(X - \overline{X})^2 \sum(Y - \overline{Y})^2}}$$

Esta ecuación refleja la covariabilidad estandarizada entre las dos variables, eliminando los efectos de escala y permitiendo comparaciones directas en distintos contextos. Su uso es recomendable cuando se ha verificado que la relación entre las variables es lineal, ya que, en caso contrario, la interpretación del coeficiente puede ser errónea o poco representativa (Camacho Martínez Vara de Rey, s.f.).

Variance Threshold

El método Variance Threshold es una técnica de selección de características utilizada en aprendizaje automático para eliminar características con baja varianza, ya que estas no aportan información relevante para la tarea de modelado. Los resultados del análisis muestran que reducir el número de características mediante Variance Threshold permite disminuir el

ruido en los datos sin comprometer significativamente la precisión del modelo (Muzaffar et al., 2023).

$$Var(X_i) = \frac{1}{N}\sum_{j=1}^{N}\left(X_{ij} - \bar{X}_i\right)^2$$

Donde el criterio de selección es:

$$Var(X_i) < t$$

Hist Gradient Boosting

Es una versión optimizada de Gradient Boosting que mejora la eficiencia computacional al utilizar histogramas en lugar de evaluar cada punto de datos individualmente. Este método permite manejar grandes volúmenes de datos sin comprometer el rendimiento del modelo. El proceso sigue una estructura iterativa en la que se construyen secuencialmente árboles de decisión, con el objetivo de minimizar la función de pérdida. Inicialmente, se establece una predicción base optimizada mediante:

$$F_0(x) = arg\,\min_{\beta}\sum_{i=0}^{N-1} L(t_i, \beta)$$

donde β es un parámetro inicial y t_i la variable objetivo. En cada iteración, se calcula la dirección del gradiente para determinar la diferencia entre la predicción actual y los valores reales:

$$y_i^m = -\left[\frac{\partial L\left(y_i, F(x_i)\right)}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$$

Con estos residuos, se ajusta un nuevo árbol de decisión minimizando el error:

$$a_m = \arg\min_{\beta}\sum_{i=0}^{N-1}[y_i^m - \beta \cdot h(x_i, a_m)]$$

Finalmente, el modelo se actualiza incorporando el nuevo árbol:

$$F_m(x) = F_{m-1}(x) + \beta_m \cdot h(x_i, a_m)$$

donde β_m es el peso que controla la contribución del nuevo árbol.

La optimización clave de Hist Gradient Boosting es el uso de bins para agrupar valores de características antes de calcular las divisiones en los árboles de decisión. Este enfoque reduce la carga computacional al evitar evaluar cada punto de datos de manera individual. La ganancia de información en cada división se mide mediante la ecuación:

$$\text{Ganancia} = \frac{\left(\sum G_{\text{izq}}\right)^2}{H_{\text{izq}} + \lambda} + \frac{(\sum G_{\text{der}})^2}{H_{\text{der}} + \lambda} - \frac{(\sum G_{\text{total}})^2}{H_{\text{total}} + \lambda} - \gamma$$

donde G representa la suma de gradientes, H la suma de las segundas derivadas (Hessiana), λ un parámetro de regularización y γ una penalización para evitar divisiones innecesarias. Este método mejora el rendimiento en términos de tiempo de entrenamiento y uso de memoria, haciéndolo especialmente útil en problemas con grandes conjuntos de datos y numerosas características (Nhat-Duc & Van-Duc, 2023).

Codificación de variables categóricas

One-Hot encoding

Es una técnica de codificación de variables categóricas agnóstica al objetivo, que transforma cada categoría en un vector disperso donde solo una posición contiene un '1' y el resto son '0'. Si una característica tiene N niveles únicos, su representación será un vector de tamaño N. Esta estrategia evita introducir relaciones de orden artificiales entre categorías, a diferencia de la codificación ordinal. Sin embargo, su principal limitación es que incrementa considerablemente el número de columnas, lo cual puede generar problemas de "maldición de la dimensionalidad" en conjuntos de datos pequeños o con alta cardinalidad. En conjuntos grandes, este efecto se mitiga, aunque aumenta el tiempo de entrenamiento y el consumo de memoria (Poslavskaya & Korolev, 2023).

Codificación Ordinal

La codificación ordinal, también conocida como integer encoding, es considerada una de las estrategias más simples para transformar características categóricas en valores numéricos. Este método asigna a cada nivel observado un número entero, generalmente entre 1 y L, donde L es el número de categorías únicas. Aunque en teoría los nuevos niveles podrían codificarse como L+1 o 0, esto llevaría a predicciones arbitrarias, ya que el orden de los enteros no refleja relaciones informativas. Para evitarlo, los niveles nuevos se tratan como valores faltantes y se imputan con la moda del conjunto de entrenamiento. Esta técnica es aceptable principalmente para modelos basados en árboles, que pueden separar los niveles mediante divisiones sucesivas (Pargent et al., 2022).

METODOLOGÍA

Selección de características sobre datos estructurados

El análisis de feature selection se llevó a cabo con el objetivo de identificar relaciones entre las variables del conjunto de datos. Para ello, se seleccionaron aquellas variables de tipo numérico y booleano, ya que estas permiten calcular coeficientes de correlación de manera adecuada. A continuación, se presentan los hallazgos más relevantes:

Hist Gradient Boosting	Variance Threshold	Correlación
sie_ic_monto_adjudicacion	sd_presupuesto_referencial_total_sin_iva	sie_ic_estado_proc_desierta
sie_ic_month	sie_ic_monto_presupuestado	sie_ic_monto_adjudicacion
sd_anticipo_pct	sie_ic_monto_contrato	sie_ic_monto_presupuestado
sie_ic_estado_proc_finalizada	sie_ic_monto_adjudicacion	sd_presupuesto_referencial_total_sin_iva
sie_ic_estado_proc_adjudicado_-_registro_de_co...	sd_plazo_de_entrega	sie_ic_monto_contrato

Tabla 2: Resultados de los modelos de feature selection top 5 variables

La *Tabla 2* presenta los cinco principales atributos seleccionados por distintos métodos de selección de características. En el caso de Variance Threshold, se observa que todas las variables seleccionadas corresponden a montos de dinero, a diferencia de lo que ocurre con los métodos Hist Gradient Boosting y Correlación. Esto se debe a que Variance Threshold prioriza las variables con mayor dispersión y descarta aquellas cuyos valores son más homogéneos, como los indicadores binarios que solo oscilan entre 0 y 1. Finalmente, la similitud en los resultados de Hist Gradient Boosting y Correlación es particularmente relevante, ya que el objetivo del análisis es clasificar los procesos según su nivel de riesgo de corrupción. La coincidencia entre ambos métodos refuerza la importancia no solo de los montos involucrados, sino también de los estados del proceso y de características clave asociadas a la contratación. Considerando que los indicadores de riesgo forman parte de la variable objetivo (`sie_ic_promedio`), se decidió eliminar del conjunto de datos estos indicadores, quedando un dataset final de 5710 filas y 55 columnas, correspondiente a procesos de contratación realizados en los años 2020, 2022 y 2023.

Extracción de texto desde documentos

Para esta sección, se realizó la extracción del contenido de los archivos con extensión .ushay, los cuales forman parte de los documentos disponibles para cada proceso de contratación pública. El formato .ushay es una compresión utilizada por Sistema Oficial de Contratación Pública del Ecuador (SOCE), funcionalmente similar a un archivo .zip, pero que incorpora un separador interno denominado "sercop" para estructurar su contenido. Algunos archivos .ushay están fragmentados en varias partes, las cuales deben ser reconstruidas para poder ser tratadas como un archivo comprimido estándar.

El procedimiento implementado consta de tres etapas principales. En primer lugar, se carga un archivo CSV que contiene la información de los documentos y se filtran únicamente aquellos con extensión .ushay. Luego, los archivos se agrupan según el `sl_contract_id`, clasificándolos en dos casos: archivos únicos, que se almacenan directamente en formato .zip, y archivos fragmentados, los cuales se reconstruyen verificando su integridad mediante el algoritmo SHA-1 antes de consolidarlos en un solo .zip. Finalmente, se asegura la integridad de los archivos reconstruidos y se manejan posibles errores para evitar el procesamiento de archivos corruptos. Para esto se implementaron funciones específicas como `calcular_hash_sha1` para validar fragmentos y `verificar_y_unir_partes` para unir los archivos.

Es importante destacar que aquellos archivos que no pudieron asociarse con ningún identificador dentro de la base fueron descartados del análisis para garantizar la coherencia y la integridad del conjunto de datos utilizado. Para realizar la extracción de texto desde los archivos, se utilizó la librería python-magic, la cual identifica el tipo de archivo examinando sus cabeceras según una lista predefinida de firmas, en lugar de basarse únicamente en su extensión. Esta funcionalidad, comúnmente utilizada en sistemas Unix mediante el comando file, permitió filtrar archivos no compatibles o irrelevantes (como temas, archivos vacíos o en formato XML) antes de intentar su procesamiento. Posteriormente, se empleó la librería textract para extraer el contenido textual de los documentos válidos de manera automatizada.

El conjunto de archivos a analizar estaba definido en un DataFrame, donde se especificaban las rutas y extensiones correspondientes. Para mejorar la eficiencia del procesamiento, se implementó ProcessPoolExecutor, lo que permitió ejecutar la extracción de texto en paralelo aprovechando múltiples núcleos del procesador. Cada archivo fue validado para asegurar su existencia y compatibilidad, y los resultados obtenidos se almacenaron en una nueva columna del DataFrame. Esta metodología permitió construir de forma eficiente una

columna denominada `extracted_text`. Para preparar el corpus textual (`extracted_text`) para su análisis se implementó un proceso de preprocesamiento lingüístico utilizando herramientas de procesamiento de lenguaje natural (PLN) en español. En primer lugar, se cargaron los recursos necesarios de las librerías nltk y spaCy, incluyendo el modelo grande de spaCy en español (es_core_news_lg) y las listas de palabras vacías (stopwords). El objetivo de esta etapa fue limpiar y normalizar el texto extraído previamente de los documentos .ushay.

El preprocesamiento consistió en dos fases. La primera aplicó una limpieza básica que convirtió el texto a minúsculas, eliminó caracteres especiales y redujo los espacios innecesarios. En la segunda fase, se aplicó lematización mediante spaCy, lo que permitió reducir las palabras a su forma base o canónica. Además, se eliminaron las stopwords en español, signos de puntuación y espacios vacíos, generando así una versión del texto más limpia y enfocada en su contenido semántico. Este procedimiento fue encapsulado en funciones reutilizables que facilitaron la transformación sistemática del corpus para su posterior análisis o modelado. Como resultado del proceso de extracción y preprocesamiento textual, se generó un archivo denominado `df_contracts_clean.pkl`. Este archivo en formato `.pkl` contiene una base de datos con 23342 entradas, cada una correspondiente a un archivo procesado. En esta base se almacena información clave de cada documento, incluyendo el identificador del proceso (`sl_contract_id`), el nombre del archivo (`file_name`), su ruta (`file_path`), la extensión (`file_extension`), el texto extraído (`extracted_text`) y el texto limpio y lematizado (`clean_text`).

Doc2Vec

Con el objetivo de capturar la información semántica contenida en los textos de los contratos públicos, se entrenó un modelo Doc2Vec a partir de los documentos preprocesados. Para ello, se utilizó la biblioteca gensim, agrupando primero los textos por contrato a través del

identificador sl_contract_id. Cada contrato fue representado como un único documento concatenado, y posteriormente transformado en un objeto TaggedDocument, necesario para el entrenamiento. El modelo fue entrenado con los siguientes parámetros: vector_size=300, window=5, min_count=2, epochs=40 y utilizando el esquema de entrenamiento distribuido (dm=1). Una vez entrenado, se generó un vector de 300 dimensiones para cada contrato, el cual captura patrones léxicos y contextuales representativos del contenido contractual.

Los vectores fueron almacenados inicialmente en una sola columna (doc2vec_vector) dentro de un archivo pkl, y luego expandidos en columnas individuales (emb_0 a emb_299). Estos vectores fueron utilizados posteriormente en conjunto con los datos estructurados como parte del conjunto de características para los modelos de clasificación.

Selección de fragmentos

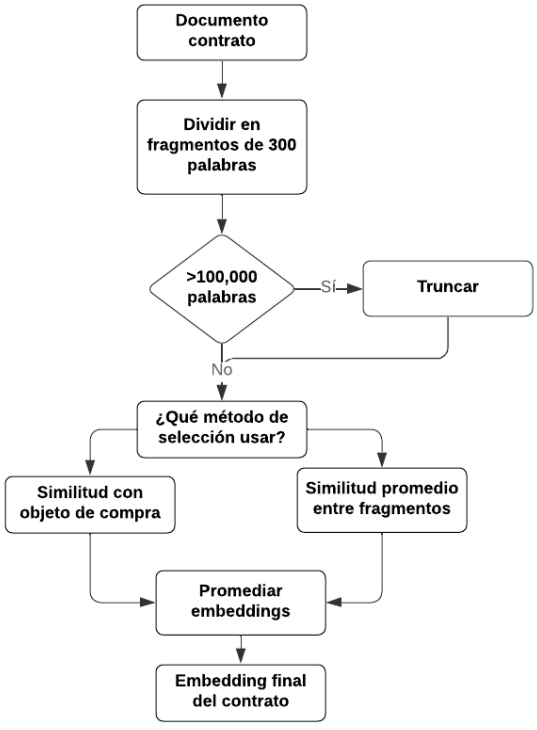

Figura 1: Pipeline selección de fragmentos

Como se muestra en la Figura 1, se implementaron dos estrategias para la selección de fragmentos en los archivos de procesos de compra pública. En la primera estrategia, los

fragmentos se seleccionan evaluando su similitud promedio con otros fragmentos del mismo contrato. Se calculan las similitudes de coseno entre todos los fragmentos y se retienen aquellos que no son excesivamente repetitivos (es decir, que no son demasiado parecidos al resto). De esta forma, se busca representar el contenido del contrato de manera más variada y evitar redundancias, sin depender del texto del objeto de compra. En la segunda metodología, la selección de fragmentos se guía por su similitud directa con el objeto de compra asociado a cada contrato. Se calcula la similitud de coseno entre el embedding de cada fragmento y el embedding del objeto de compra. Posteriormente, se priorizan los fragmentos más relevantes, acumulando tantos como sea posible hasta alcanzar un máximo de tokens permitidos por documento. Esta estrategia permite enfocar la representación textual en fragmentos más relevantes para el objetivo de la contratación.

Para la construcción de fragmentos, cada documento se divide en bloques de 300 palabras. Se establece un límite máximo de 100,000 palabras por contrato, truncándose aquellos documentos que superen este umbral. En la etapa de selección, cuando se emplea la similitud promedio entre fragmentos, se utiliza un umbral de 0.85 para descartar fragmentos redundantes. Por otro lado, en la metodología basada en el objeto de compra, se prioriza la selección de fragmentos más relevantes, acumulando hasta un máximo de 8192 tokens por contrato. Además, se implementa un mecanismo de guardado automático cada 50 contratos procesados, con el fin de preservar el avance y evitar pérdidas en caso de interrupciones. Finalmente, los fragmentos seleccionados se utilizan para generar un embedding final por contrato, ya sea mediante la concatenación de los fragmentos y la generación de un nuevo embedding, o mediante el promedio de los embeddings individuales.

Limpieza base de datos

Se cargó el archivo `cleaned_dataset-fe.csv` y se evaluaron sus dimensiones y la cantidad de valores nulos por columna. Se clasificaron las columnas en tres grupos: categóricas (`object` o `category`), booleanas (`bool`) y numéricas (`int64` y `float64`). Para garantizar la calidad del dataset utilizado en los experimentos, se aplicaron las siguientes etapas de limpieza y depuración:

- **Eliminación de columnas con valores nulos excesivos:** se eliminan todas las columnas cuyo porcentaje de valores nulos superaba el 50%.
- **Tipo booleano:** las columnas de tipo booleano son transformadas al formato numérico binario (0 y 1).
- **Eliminación de columnas duplicadas:** se detecta y elimina columnas que tienen nombres repetidos o contenido idéntico a otras columnas, manteniendo solo una copia de cada grupo duplicado.
- **Imputación de valores faltantes:** se utiliza `SimpleImputer` con estrategia de mediana para completar los datos nulos.
- **Aplicación de codificación ordinal:** a las columnas categóricas, como los nombres de entidades.
- **Codificación de la variable objetivo (`sie_ic_promedio`):** para el primer experimento, se utiliza una clasificación multiclase en cuatro categorías, mientras que para el segundo experimento se emplea una clasificación binaria (1/0).
- **Exportación del dataset limpio:** El dataset resultante, fue guardado como `1_clean_data_base.csv` para ser utilizado en las siguientes etapas.

Se imputaron los valores faltantes. Para las variables numéricas, se empleó la mediana como estrategia de imputación, ya que es robusta frente a valores atípicos. En el caso de las

variables categóricas, se utilizó la moda (valor más frecuente), con el fin de preservar la distribución más representativa de cada atributo.

Una vez completada la imputación, se generaron dos versiones del dataset, adaptadas a las necesidades de diferentes tipos de modelos. La primera versión utilizó la técnica de One-Hot Encoding, la cual transforma cada categoría en una nueva columna binaria. Este enfoque es particularmente adecuado para modelos lineales o redes neuronales, ya que evita introducir una relación ordinal artificial entre las categorías. Dado que la versión del dataset codificada mediante One-Hot Encoding contenía un total de 7.770 columnas, resultaba necesario aplicar una técnica de reducción de dimensionalidad con el fin de optimizar el rendimiento computacional y evitar el sobreajuste en los modelos de clasificación. Para ello, se utilizó el método de selección univariada de características basado en análisis de varianza (ANOVA), mediante la función SelectKBest con el estadístico f_classif. Se seleccionaron las 100 características más relevantes con mayor capacidad discriminativa respecto a la variable de riesgo, y se construyó un nuevo DataFrame resultante con estas columnas.

La segunda versión se construyó mediante codificación ordinal, donde cada categoría fue asignada a un valor numérico entero. Este tipo de codificación es comúnmente utilizado por modelos basados en árboles, como Random Forests o Gradient Boosting, ya que estos algoritmos no se ven afectados por el orden artificial impuesto. Esta versión del conjunto de datos se exporta como archivos separados: `2_1_encoded_onehot.csv` y `2_2_encoded_ordinal.csv`. Estas versiones constituyen la base para los experimentos de clasificación y selección de características que se realizan en fases posteriores.

Se comparan los resultados de modelos entrenados con datasets codificados mediante one-hot encoding y codificación ordinal (ver *Tabla 3*). Aunque one-hot encoding suele recomendarse para modelos sensibles al espacio vectorial, en el caso del SVM evaluado, la codificación ordinal ofrece mejores resultados. Para el MLP, en cambio, el uso de one-hot

encoding resulta más adecuado. Considerando que las variables codificadas no son relevantes para los modelos de clasificación, se decide descartar `2_1_encoded_onehot.csv` en las fases posteriores del análisis.

Tipo de Datos	Model	Best Params	Accuracy	F1 Score	Precision	Recall	AUC-ROC
ONE HOT ENCODED	**SVM**	{'C': 100, 'gamma': 'auto', 'kernel': 'rbf'}	0.4459	0.4410	0.4546	0.4459	0.7226
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100, 50), 'solver': 'adam'}	0.4630	0.4613	0.4617	0.4630	0.7065
ORDINAL ENCODED	**SVM**	{'C': 0.1, 'gamma': 'auto', 'kernel': 'rbf'}	0.4649	0.4578	0.4848	0.4649	0.7262
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (50, 50), 'solver': 'adam'}	0.4364	0.4355	0.4353	0.4364	0.6825

Tabla 3: Comparación de codificaciones One-Hot y Ordinal en SVM y MLP

Base de datos desbalanceada

Para este primer experimento se utilizó exclusivamente la base de datos del año 2022, la cual contiene un total de 2106 registros y 67 columnas, incluyendo los indicadores individuales denominados *sie_ic_indicador{número}*. Inicialmente, la distribución de archivos para la clasificación multiclase del riesgo asociado a un proceso en las categorías bajo, medio, medio alto y alto se realiza dividiendo el rango total del indicador compuesto en cuatro intervalos iguales de 0,25 (ver *Tabla 4*). De esta manera, los valores comprendidos entre 0 y 0,25 se categorizan como de alto riesgo, y así sucesivamente para los demás niveles.

Bajo	849
Medio	1179
Medio-alto	77
Alto	1

Tabla 4: Distribución de categorías de riesgo según el rango del indicador compuesto

Posteriormente, se adopta un enfoque basado en cuartiles para lograr una segmentación más representativa de los niveles de riesgo. Utilizando la variable `sie_ic_promedio`, se calculan los percentiles 25 (Q1), 50 (Q2) y 75 (Q3), los cuales sirven de referencia para establecer los siguientes rangos:

- Q1: valores entre 0.25 y 0.71
- Q2: valores entre 0.73 y 0.75
- Q3: valores entre 0.80 y 0.86
- Q4: valores entre 0.88 y 1.00

Cabe señalar que estos rangos no son perfectamente contiguos, debido a la distribución específica de los datos y a la existencia de valores concentrados en ciertos tramos. A continuación, se presenta la *Tabla 5* con el número de casos por categoría, así como el valor mínimo, máximo y promedio del índice dentro de cada grupo:

Nivel de riesgo	Cantidad	Rango	Promedio
Q1	622	0.25 – 0.71	0.61
Q2	494	0.73 – 0.75	0.75
Q3	490	0.80 – 0.86	0.84
Q4	272	0.88 – 1.00	0.89

Tabla 5: Distribución de procesos según nivel de riesgo (cuartiles)

Como resultado, se obtiene un dataset:

7_1_dataset_doc2vec_ordinal.csv: combinación de atributos estructurados (codificados mediante Ordinal Encoding) y vectores generados mediante Doc2Vec.

Base de datos balanceada

En el marco del segundo experimento de este estudio, se redefine el problema como una tarea de clasificación binaria supervisada. Esta decisión metodológica responde a la necesidad de simplificar la interpretación del riesgo en los procesos de contratación pública, estableciendo una distinción clara entre contratos de alto y bajo riesgo, en lugar de trabajar con múltiples clases. Para la construcción del conjunto de datos, se utiliza como base la información previamente procesada correspondiente al año 2022. Con el objetivo de mejorar la robustez del modelo y su capacidad de generalización, se integran también registros provenientes de los años 2020 y 2023, utilizando la base completa consolidada de 5710 registros y 67 columnas.

La variable de salida, originalmente continua (`sie_ic_promedio`), se transforma en una variable categórica binaria. Para determinar el umbral de decisión, se analiza la relación entre distintos valores de `sie_ic_promedio` y la estadística F promedio, evaluando así la separación entre las clases. Como se observa en la *Figura 2* de F-statistic promedio versus Threshold, el valor de 0.56 presenta un pico significativo en la capacidad de discriminación. Por esta razón, se establece 0.56 como umbral óptimo. En consecuencia, se etiquetan como clase 1 (riesgo alto) todos los contratos cuyo valor de `sie_ic_promedio` es inferior a 0.56, y como clase 0 (riesgo bajo) aquellos contratos con un valor igual o superior. Esta discretización permite reconfigurar el problema como una clasificación binaria, facilitando tanto la interpretación como la evaluación del desempeño de los modelos utilizados.

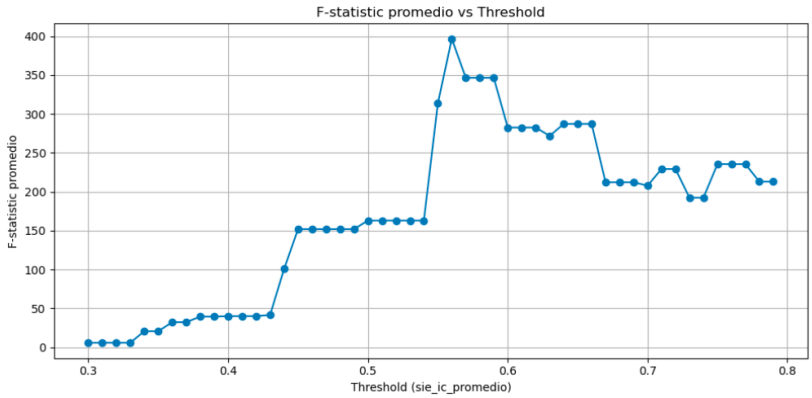

Figura 2: Evolución del valor promedio de la F-statistic en función del umbral aplicado sobre `sie_ic_promedio`*. El umbral de 0.56 se selecciona como óptimo al observarse un pico en la capacidad de separación entre clases.*

Adicionalmente, se observa un desbalance significativo entre ambas clases, con una clara predominancia de contratos etiquetados como riesgo bajo. Para mitigar este sesgo y asegurar un entrenamiento más equitativo de los clasificadores, se aplica una técnica de submuestreo aleatorio (undersampling) a la clase mayoritaria. En concreto, se extrae una muestra aleatoria de contratos de bajo riesgo igual en tamaño al conjunto de contratos de alto riesgo, con el fin de equilibrar el conjunto de entrenamiento. Esta estrategia busca reducir el sesgo hacia la clase dominante, mejorar la sensibilidad del modelo ante casos de riesgo y contribuir a una evaluación más justa de su capacidad predictiva. Como resultado de este proceso, se obtiene una base de datos balanceada compuesta por 930 contratos (filas) y 67 atributos (columnas), distribuidos equitativamente entre ambas clases: 465 contratos de alto riesgo y 465 contratos de bajo riesgo.

Cabe destacar que, para garantizar la coherencia metodológica entre ambos experimentos, se mantuvieron las mismas técnicas de limpieza de datos estructurados y preprocesamiento de textos utilizadas en el experimento anterior. Esto incluyó el tratamiento de valores nulos, la normalización de campos numéricos y categóricos, así como la limpieza, lematización y vectorización de los documentos textuales asociados a cada contrato. De esta

forma, se aseguró la comparabilidad entre los modelos desarrollados en ambos escenarios y se preservó la integridad del pipeline de procesamiento.

Evaluación comparativa de modelos de clasificación

Con el objetivo de determinar la efectividad de diferentes modelos de clasificación en la predicción del nivel de riesgo de los contratos públicos, se evaluaron cuatro algoritmos: Random Forest, SVM, HistGradient Boosting y MLP. Para cada modelo se realizó una búsqueda de hiperparámetros mediante validación cruzada estratificada de 5 pliegues, utilizando la métrica F1 ponderado como criterio de optimización. Esta búsqueda se implementó a través de GridSearchCV.

Las métricas obtenidas (precision, recall, F1-score, accuracy y AUC ROC) corresponden exclusivamente al desempeño del modelo final sobre el conjunto de prueba independiente. Los resultados que se presentan en la *Tabla 6* y la *Tabla 7* reflejan el rendimiento real de cada modelo una vez optimizado y evaluado sobre datos no vistos.

RESULTADOS Y ANÁLISIS

Clasificación multiclase y base de datos desbalanceada

Tipo de datos	Model	Best Params	Accuracy	F1 Score	Precision	Recall	AUC-ROC
Dataset completo	Random Forest	{'max_depth': None, 'max_features': 'sqrt', 'min_samples_split': 2, 'n_estimators': 200}	0.4410	0.4149	0.4475	0.4410	0.6831
	SVM	{'C': 0.01, 'gamma': 'scale', 'kernel': 'linear'}	0.3842	0.3880	0.4044	0.3842	0.6541
	Gradient Boosting	{'l2_regularization': 0.5, 'learning_rate': 0.1, 'max_depth': 10, 'max_iter': 200}	0.4061	0.3967	0.4013	0.4061	0.6868
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'lbfgs'}	0.3842	0.3833	0.3871	0.3842	0.6395
Sin Embeddings	Random Forest	{'max_depth': None, 'max_features': None, 'min_samples_split': 2, 'n_estimators': 100}	0.4759	0.4699	0.4800	0.4759	0.7309
	SVM	{'C': 10, 'gamma': 'auto', 'kernel': 'poly'}	0.4061	0.4086	0.4145	0.4061	0.6804
	Gradient Boosting	{'l2_regularization': 0.1, 'learning_rate': 0.05, 'max_depth': 10, 'max_iter': 200}	0.4672	0.4640	0.4708	0.4672	0.7214
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'adam'}	0.4235	0.4240	0.4266	0.4235	0.6495
Solo embeddings	Random Forest	{'max_depth': 10, 'max_features': None, 'min_samples_split': 5, 'n_estimators': 200}	0.3056	0.2661	0.3351	0.3056	0.5119
	SVM	{'C': 0.01, 'gamma': 'scale', 'kernel': 'linear'}	0.2925	0.2811	0.2832	0.2925	0.5069
	Gradient Boosting	{'l2_regularization': 0.1, 'learning_rate': 0.05, 'max_depth': None, 'max_iter': 300}	0.2663	0.2562	0.2543	0.2663	0.4967
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'adam'}	0.2707	0.2683	0.2672	0.2707	0.5046

Tabla 6: Resultados de clasificación multiclase con datos desbalanceados.

En la *Tabla 6* se presentan los resultados de la clasificación multiclase utilizando una base de datos desbalanceada. Los modelos evaluados, tanto sobre el dataset completo como sobre variantes con y sin embeddings, muestran desempeños limitados en términos de precisión, recall y AUC-ROC. Aunque el uso de datos estructurados sin embeddings logra resultados ligeramente superiores (por ejemplo, Random Forest alcanza un AUC-ROC de 0.7309), en general, los clasificadores tienen dificultades para discriminar adecuadamente entre las clases definidas a partir de los cuartiles estadísticos. Esto puede atribuirse, en parte, a la forma en que se agruparon los datos: los rangos de los cuartiles son estrechos y se superponen considerablemente (Q1: 0.25–0.71; Q2: 0.73–0.75; Q3: 0.80–0.86; Q4: 0.88–1.00), lo que limita la variabilidad interna y dificulta que los modelos identifiquen patrones distintivos entre categorías. Además, los experimentos basados únicamente en embeddings muestran un descenso significativo en el rendimiento, evidenciando que la representación textual aislada no es suficiente para capturar las diferencias necesarias entre las clases. Estos resultados sugieren que, en el escenario planteado, la segmentación de los datos no genera clases bien separables y que es necesario considerar estrategias adicionales de ingeniería de atributos o redefinición de las categorías para mejorar la capacidad predictiva de los modelos.

Clasificación binaria y base de datos balanceada

Tipo de datos	Model	Best Params	Accuracy	F1 Score	Precision	Recall	AUC-ROC
Dataset completo	Random Forest	{'max_depth': None, 'max_features': 'sqrt', 'min_samples_split': 2, 'n_estimators': 200}	0.6738	0.6698	0.6832	0.6738	0.7465
	SVM	{'C': 0.01, 'gamma': 'scale', 'kernel': 'linear'}	0.7339	0.7338	0.7339	0.7339	0.7381
	Gradient Boosting	{'l2_regularization': 0.5, 'learning_rate': 0.1, 'max_depth': 10, 'max_iter': 200}	0.6695	0.6682	0.6725	0.6695	0.7448
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'lbfgs'}	0.7081	0.7081	0.7081	0.7081	0.7816
Sin Embeddings	Random Forest	{'max_depth': None, 'max_features': None, 'min_samples_split': 2, 'n_estimators': 100}	0.7210	0.7368	0.6947	0.7845	0.7784
	SVM	{'C': 10, 'gamma': 'auto', 'kernel': 'poly'}	0.7339	0.7281	0.7411	0.7155	0.7960
	Gradient Boosting	{'l2_regularization': 0.1, 'learning_rate': 0.05, 'max_depth': 10, 'max_iter': 200}	0.6695	0.6778	0.6585	0.6983	0.7492
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'adam'}	0.6738	0.6637	0.6818	0.6466	0.7384
Solo embeddings	Random Forest	{'max_depth': 10, 'max_features': None, 'min_samples_split': 5, 'n_estimators': 200}	0.5923	0.5909	0.5932	0.5923	0.6213
	SVM	{'C': 0.01, 'gamma': 'scale', 'kernel': 'linear'}	0.5579	0.5572	0.5581	0.5579	0.5672
	Gradient Boosting	{'l2_regularization': 0.1, 'learning_rate': 0.05, 'max_depth': None, 'max_iter': 300}	0.5622	0.5621	0.5622	0.5622	0.6236
	MLP	{'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'adam'}	0.6094	0.6090	0.6097	0.6094	0.6370

Tabla 7: Resultados de clasificación binaria con datos balanceados.

En la *Tabla 7* se presentan los resultados de la clasificación binaria sobre una base de datos balanceada construida con información de contratación pública correspondiente a los años 2020, 2022 y 2023. La variable de salida, originalmente continua (`sie_ic_promedio`), se discretizó utilizando un umbral de 0.56, determinado mediante el análisis de la relación entre distintos thresholds y la estadística F promedio. Esta discretización permitió redefinir el problema como una tarea de clasificación binaria, diferenciando contratos de alto y bajo riesgo de corrupción.

Los resultados muestran un desempeño generalizado superior al observado en el escenario multiclase previo. Dentro del grupo que utiliza el dataset completo (información estructurada y textual), el clasificador SVM obtiene el mayor accuracy (0.7339) y un AUC-ROC competitivo (0.7381), mientras que el MLP alcanza el mejor AUC-ROC (0.7816), destacándose en la capacidad de distinguir entre contratos de riesgo alto y bajo de corrupción. Al evaluar los modelos que excluyen embeddings, se observa que SVM sigue obteniendo el mayor AUC-ROC (0.7960), superando incluso su desempeño anterior, lo cual sugiere que las variables estructuradas contienen información relevante para la detección temprana del riesgo. En contraste, los modelos entrenados únicamente con embeddings presentan una disminución significativa en su rendimiento, siendo Random Forest el que logra el mejor resultado en este escenario (AUC-ROC = 0.6213), aunque inferior al alcanzado en las otras configuraciones.

Estos resultados confirman que la redefinición del problema mediante un umbral estadísticamente fundamentado y la inclusión de variables estructuradas favorecen considerablemente el desempeño de los modelos. Asimismo, evidencian la importancia de integrar distintas fuentes de información en el proceso de detección temprana de riesgos en la contratación pública.

CONCLUSIONES

Los resultados de la clasificación binaria muestran que la redefinición del problema utilizando un umbral estadísticamente fundamentado (0.56 en `sie_ic_promedio`) mejora notablemente la capacidad de los modelos para discriminar entre contratos de alto y bajo riesgo. Los modelos basados en datos estructurados y documentos completos (sin excluir embeddings) logran desempeños sólidos, destacando el SVM y el MLP como los mejores clasificadores en términos de precisión y capacidad de discriminación (AUC-ROC superiores a 0.73).

Sin embargo, al analizar los experimentos utilizando únicamente representaciones de texto (embeddings), se observa una disminución considerable en el rendimiento, lo cual sugiere que los documentos de contratación por sí solos, sin variables adicionales que actúen como indicadores de riesgo, no contienen suficiente información explícita para diferenciar de manera efectiva entre procesos de alto y bajo riesgo de corrupción. Esta limitación puede deberse a diversos factores, como la falta de relevancia de los fragmentos textuales seleccionados para representar los documentos, o la posibilidad de que el indicador de riesgo no esté directamente correlacionado con las especificaciones técnicas presentes en los textos. Además, es posible que los modelos de embeddings utilizados no capten adecuadamente las sutilezas del lenguaje técnico-administrativo de los documentos, lo cual dificulta la extracción de señales semánticas significativas. Este hallazgo evidencia la importancia de complementar las representaciones textuales con variables estructuradas o diseñar indicadores específicos que capturen dimensiones relevantes no explícitas en el lenguaje natural.

En conclusión, integrar variables estructuradas como indicadores de riesgo, resulta fundamental para mejorar la capacidad predictiva de los modelos de aprendizaje automático en tareas de detección temprana de corrupción. Además, estos resultados resaltan la necesidad de estrategias de ingeniería de atributos más sofisticadas que permitan extraer, sintetizar y

representar información crítica que no está directamente disponible en los textos de contratación pública.

TRABAJO FUTURO

La detección temprana de riesgos de corrupción en la contratación pública requiere aprovechar de manera sistemática los grandes volúmenes de datos generados por los procesos de compra. Aunque los registros electrónicos ofrecen una fuente rica de información, su análisis masivo todavía representa un desafío en términos metodológicos y de interpretación (Jorquera, 2019).

En esta dirección, un tema prioritario es la detección de procesos de contratación pública dirigidos. Según Salazar Morales y Angles Arenas (2018), un proceso dirigido ocurre cuando las reglas o instrumentos de contratación, aunque formalmente cumplen con la legalidad, son diseñados de forma que favorecen intencionadamente a un proveedor específico, restringiendo la competencia legítima. Este direccionamiento puede manifestarse mediante especificaciones técnicas excesivamente restrictivas, estudios de mercado sesgados. A partir de esta conceptualización, sería posible analizar los objetos de compra y las características técnicas de los procesos de licitación, extrayendo métricas como la cantidad y especificidad de las especificaciones técnicas. Con esta información, se podría construir un modelo que estime la probabilidad de que un proceso haya sido direccionado, incorporando variables adicionales como el número de oferentes, la duración de los plazos y la rigidez de los requisitos establecidos.

En este mismo sentido, fortalecer los modelos predictivos de riesgo de corrupción mediante la integración sistemática de indicadores de "banderas rojas" constituye otra vía complementaria de investigación. Diversos estudios han demostrado que características observables, como la falta de transparencia en los pliegos, la existencia de oferentes únicos o anomalías en los procedimientos de adjudicación, están asociadas a mayores niveles de riesgo (Adam, Fazekas, Regös, & Tóth, 2020). Incorporar estos indicadores permitiría enriquecer las

variables analizadas y capturar patrones de riesgo que no siempre son evidentes en los datos textuales o estructurados tradicionales.

Finalmente, se propone explorar el uso de los archivos *proceso.xml*, contenidos dentro de los paquetes `.ushay`, que incluyen más de diez tablas con información detallada de cada proceso de contratación. El análisis de estos datos permitiría incorporar nuevas variables no consideradas en el presente estudio, contribuyendo al diseño de nuevos indicadores de riesgo y al enriquecimiento del dataset disponible. Asimismo, queda abierta una pregunta relevante para investigaciones futuras: ¿Cómo identificar formalmente cuándo un proceso de contratación pública ha sido dirigido, a partir de la evidencia técnica contenida en las especificaciones y condiciones de licitación?

REFERENCIAS BIBLIOGRÁFICAS

Santos Alava, J. E., Albarez Navarro, G. H., Mata Anchundia, D. D., & Chang Rizo, F. S. (2024). *El impacto de la corrupción en la sociedad y la economía de la administración pública en Ecuador*. *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades*, *5*(6), 2975–2989. https://doi.org/10.56712/latam.v5i6.3219

Transparency International. (2024). Corruption Perceptions Index 2024: Ecuador. [Sitio web]. Recuperado el 27 de enero de 2025, de https://www.transparency.org/en/cpi/2021/index/ecu

Universidad San Francisco de Quito. (2023). Kapak: Transparencia en Compras Públicas. [Sitio web]. Recuperado el 27 de enero de 2025, de https://noticias.usfq.edu.ec/2023/11/kapak-transparencia-en-compras-publicas.html

Aldana, A., Falcón-Cortés, A., & Larralde, H. (2022). *A machine learning model to identify corruption in México's public procurement contracts* (arXiv:2211.01478). arXiv. https://doi.org/10.48550/arXiv.2211.01478

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *arXiv preprint*, arXiv:1301.3781. https://arxiv.org/abs/1301.3781

Le, Q., & Mikolov, T. (2014). *Distributed Representations of Sentences and Documents.* Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed Representations of Words and Phrases and their Compositionality*. http://arxiv.org/abs/1310.4546

Jurafsky, D., & Martin, J. H. (2025). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition with language models* (3rd ed.). https://web.stanford.edu/~jurafsky/slp3

Camacho Martínez Vara de Rey, C. (s.f.). *Coeficiente de correlación lineal de Pearson*. Recuperado de https://personal.us.es/vararey/adatos2/correlacion.pdf

Muzaffar, A., Hassen, H. R., Zantout, H., & Lones, M. A. (2023). *A comprehensive investigation of feature and model importance in Android malware detection*. arXiv. Recuperado de https://arxiv.org/abs/2301.12778

Nhat-Duc, H., & Van-Duc, T. (2023). *Comparison of histogram-based gradient boosting classification machine, random forest, and deep convolutional neural network for pavement raveling severity classification*. Automation in Construction, 148, 104767. https://doi.org/10.1016/j.autcon.2023.104767

Robertson, S. (2004). *Understanding inverse document frequency: On theoretical arguments for IDF*. Journal of Documentation, 60(5), 503–520. https://doi.org/10.1108/00220410410560582

Poslavskaya, E., & Korolev, A. (2023). *Encoding categorical data: Is there yet anything 'hotter' than one-hot encoding?* arXiv. https://arxiv.org/abs/2312.16930

Pargent, F., Pfisterer, F., Thomas, J., & Bischl, B. (2022). *Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features*. Computational Statistics. https://doi.org/10.1007/s00180-022-01207-6

García Rodríguez, M. J., Rodríguez-Montequín, V., Ballesteros-Pérez, P., Love, P. E. D., & Signor, R. (2022). Collusion detection in public procurement auctions with machine learning algorithms. *Automation in Construction, 133*, 104047. https://doi.org/10.1016/j.autcon.2021.104047

Jorquera, M. (2019). *Compras públicas y big data: Investigación en Chile sobre índice de riesgo de corrupción*. Banco Interamericano de Desarrollo.

Adam, I., Fazekas, M., Regös, N., & Tóth, B. (2020). *Más allá de las fugas: Cuantificando los efectos de la corrupción en el sector de agua y saneamiento de América Latina y el Caribe* (Nota Técnica BID-TN-2055). Banco Interamericano de Desarrollo. https://doi.org/10.18235/0002600

Salazar Morales, D., & Angles Arenas, A. (2018). *Mapeando la corrupción en los procesos de contratación pública: conceptos y metodología*. Contraloría General de la República del Perú.

Kapak. (2023). *Transparencia en compras públicas*. Universidad San Francisco de Quito. https://kapak.usfq.edu.ec/#/sie