MODELO ANALÍTICO PARA CLASIFICACIÓN DE MENSAJES LABORALES USANDO NLP

Camilo Orbes Cabrera

Ingeniero Electrónico

Semestre de Industria

Asesor interno
Daniel Escobar Grisales, M. Sc. en Ingeniería Electrónica y de Telecomunicaciones

Universidad de Antioquia
Facultad de Ingeniería
Ingeniería Electrónica
Medellín
2025

Cita	(Orbes Cabrera, 2025)
Referencia **Estilo APA 7 (2020)**	Orbes Cabrera, C. (2025). *Modelo analítico para clasificación de mensajes laborales usando NLP* [Informe de práctica]. Universidad de Antioquia, Medellín, Colombia.

Dedicatoria

A mi mamá, quien ha sido mi mayor fuente de amor, inspiración y fortaleza. Este trabajo es el reflejo de todo lo que me has enseñado: de tu esfuerzo, sacrificio y tu capacidad de soñar siempre grande. Gracias por tu paciencia, por tu apoyo incondicional, y por ser mi mayor motivación en cada paso de mi vida. Sin ti, este logro no habría sido posible. Te dedico todo mi cariño y gratitud, hoy y siempre.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mi mamá, por su amor, comprensión y apoyo incondicional durante todo el proceso de mis estudios. Por su constante respaldo emocional y motivación para seguir adelante en cada paso de mi carrera. Sin su apoyo constante, este logro no habría sido posible.

Agradezco también a mi profesor Daniel Escobar Grisales, por su dedicación y orientación durante el desarrollo de este trabajo. Sus valiosos consejos y su crítica constructiva me ayudaron a mejorar este proyecto, brindándome la confianza necesaria para llevarlo a cabo.

Finalmente, quiero expresar mi más profundo agradecimiento a mi asesor de práctica, David Esteban Betancur por su guía profesional, paciencia y por poner a mi disposición sus conocimientos y experiencia en el área. Su apoyo a lo largo de este proceso fue crucial para alcanzar los objetivos de la práctica y el desarrollo de este trabajo.

Tabla de contenido

Lista de tablas

Lista de figuras

Siglas, acrónimos y abreviaturas

RF	Random Forest
LSTM	Long Short-Term Memory
TF-IDF	Term Frequency-Inverse Document Frequency
SVM	Support Vector Machine
BoW	Bag of Words
NLP	Natural Language Processing
LSA	Latent Semantic Analysis

Resumen

La comunicación mediante plataformas digitales es una práctica común en las empresas donde se tiene una gran cantidad de personal y donde todos los colaboradores podrían estar distanciados geográficamente. Plataformas como Microsoft Teams ofrecen servicios para la comunicación interna en una empresa, pero estos servicios tienen un costo asociado. En la empresa Bancolombia se ha evidenciado un sobrecosto respecto al intercambio de mensajes fuera del límite contratado. Dentro de los diferentes análisis realizados internamente en el banco, se ha evidenciado que muchos de los mensajes que se intercambian no tienen un contenido laboral.

En este trabajo se proponen y comparan diversas metodologías para identificar aquellos mensajes que no tienen un contenido laboral. Los resultados indican que los enfoques basados en boosting de gradiente extremo (XGBoost, del inglés Xtreme Gradient Boosting), y bosques aleatorios (RF del inglés, Random Forest) logran obtener desempeños de hasta 99%, especialmente cuando la representación del texto es obtenida mediante caracterizaciones basadas en la frecuencia de términos, como la técnica de frecuencia de término – frecuencia inversa de documento (TF-IDF, del inglés Term Frequency-Inverse Document Frequency). También se consideraron estrategias más recientes, como Word2Vec, pero su desempeño fue menor, aunque su eficiencia computacional fue mayor. Finalmente, estos análisis fueron integrados los resultados en un tablero en Power Bi, con el fin de visualizar los resultados, facilitando el análisis de los flujos de mensajes en las diferentes áreas de la organización y las métricas de clasificación de los modelos.

Palabras clave: clasificación, algoritmo, procesamiento de lenguaje natural.

Abstract

Communication through digital platforms is a common practice in companies that have many employees and where all collaborators could be geographically distant. Platforms like Microsoft Teams provide services for internal communication within a company, but these services come with an associated cost. At Bancolombia, an additional cost has been observed regarding the exchange of messages beyond the contracted limit. In the different analyses conducted internally at the bank, it has been found that many of the exchanged messages do not have a work-related content.

This paper proposes and compares several methodologies to identify those messages that do not have a work-related content. The results indicate that approaches based on extreme gradient boosting (XGBoost) and random forests (RF) achieve performances of up to 99%, especially when the text representation is obtained through term frequency-based characterizations, such as the Term Frequency-Inverse Document Frequency (TF-IDF) technique. More recent strategies, such as Word2Vec, were also considered, but their performance was lower, although their computational efficiency was higher. Finally, the analysis results were integrated into a Power BI dashboard to visualize the outcomes, facilitating the analysis of message flows across different areas of the organization and the classification metrics of the models.

*Keywords***:** classification, algorithm, natural language processing.

1. Introducción

En los últimos años, el aumento en el uso de plataformas digitales para la comunicación empresarial ha generado nuevos desafíos, tanto operativos como financieros. En el caso de Bancolombia, una de las instituciones financieras más grandes de Colombia y con presencia en varios países de Latinoamérica, el uso de Microsoft Teams como canal principal de comunicación interna ha sido esencial para la interacción entre empleados (Alfonso et al., 2024). Sin embargo, el envío masivo de mensajes, muchos de los cuales no están relacionados con tareas laborales (C Fernández - La Vanguardia, 2021), ha generado sobrecostos significativos para la empresa. En noviembre de 2023, se estimó un gasto adicional de 104.274 USD ya que fueron generados 160 millones de mensajes por fuera del límite contratado (116 millones de mensajes/mes).

A pesar de la creciente importancia de este tema, en el estudio del estado del arte no se encontraron trabajos que analizaran específicamente la clasificación de mensajes en laborales y no laborales provenientes de Microsoft Teams en una compañía bancaria. Sin embargo, se identificaron estudios donde se consideran problemáticas similares usando un enfoque basado en algoritmos de procesamiento de lenguaje natural (NLP, del inglés Natural Language Processing). En (Toba et al., 2023), los autores analizaron sentimientos en un foro de discusión en línea, con el objetivo de mejorar el aprendizaje colaborativo. En este estudio, los autores demostraron que es posible extraer características relevantes de las interacciones textuales utilizando embeddings a partir de la técnica de GloVe (con vectores de 500 dimensiones), lo que facilita la categorización de los textos en tres grupos: negativo, neutral y positivo. Para la clasificación de los textos, emplearon el algoritmo de RF y la arquitectura de aprendizaje profundo empleada fue la memoria larga a corto plazo (LSTM, del inglés Long Short-Term Memory). Los resultados mostraron que el modelo de basado en RF es especialmente adecuado para el análisis y clasificación de texto ya que obtuvo una buena precisión.

Además, en la búsqueda de enfoques similares, se encontró un trabajo que se centraba en predecir la satisfacción del usuario en chats de atención al cliente de una Fintech (Romanisio & Gravano, 2024). Este estudio utilizó técnicas de preprocesamiento que incluyeron manejo de información faltante y tokenización de los mensajes de texto. El método TF-IDF fue usado para transformar el texto en representaciones numéricas. Esto les permitió a los autores extraer características

relevantes para el análisis y poder predecir actitudes a partir de los chats de atención al cliente. Los autores emplearon clasificadores como RF y XGBoost, con ajuste de hiperparámetros mediante búsqueda aleatoria y validación cruzada, con el fin de reducir el sobre ajuste.

Finalmente, se identificó un trabajo donde los autores buscaban analizar interacciones de servicio al cliente para predecir el incumplimiento de pagos en una entidad bancaria usando técnicas de NLP (Javier, 2023). Los autores consideraron extraer características que permiten captar las actitudes y preocupaciones de los clientes con el fin de encontrar patrones de comportamiento en cuanto a los pagos de sus tarjetas. Utilizaron representaciones vectoriales generadas mediante Word2Vec y otros embeddings para transformar el texto en datos estructurados, permitiendo el entrenamiento de un modelo de Máquina de Soporte Vectorial (SVM, del inglés Support Vector Machine). Los resultados de clasificación de este modelo mostraron un rendimiento alto considerando métricas de desempeño como: precisión, recall y f1-score, permitiendo la identificación de clientes en riesgo de incumplimiento a partir de ciertas actitudes encontradas. En este trabajo los autores destacan la efectividad de las representaciones semánticas en el análisis de patrones de comportamiento.

En este proyecto presentamos una estrategia para clasificar los mensajes laborales y no laborales intercambiados mediante Microsoft Teams. Para el preprocesamiento de los textos, se llevó a cabo una limpieza y normalización de los datos, seguida de la tokenización de los mensajes. En cuanto a las técnicas de NLP, se utilizaron enfoques como TF-IDF con análisis semántico latente (LSA, del inglés Latent Semantic Analisis), bolsa de palabras (BoW, del inglés Bag of Words) con LSA, TF-IDF sin LSA y Word2Vec. Estos métodos permiten encontrar una representación numérica con el objetivo de ajustar diferentes algoritmos de clasificación. Para la clasificación, se aplicaron varios algoritmos clásicos de machine learning, entre los que se incluyen RF, SVM, K- Vecinos más cercanos (KNN, del inglés K-Nearest Neighbors) y XGBoost. El resultado de los clasificadores se integró con un entorno gráfico que facilita el análisis de los mensajes con el fin de priorizar las áreas. Esta priorización consiste en que las áreas con un mayor número de mensajes laborales serán las que se prioricen para la descarga de dichos mensajes a la base de datos interna. El resultado de este proyecto busca facilitar he implementar el desarrollo de políticas que permitan dar un mejor manejo al uso de los chats, evitando excedentes de sobrecostos al no priorizar las áreas con un mayor número de mensajes no laborales.

2. Objetivos

2.1 Objetivo general

Diseño e implementación de un modelo basado en procesamiento de lenguaje natural y algoritmos de aprendizaje clásicos, para la clasificación automática de chats laborales y no laborales.

2.2 Objetivos específicos

- Etiquetar la base de datos usando dos categorías (laboral y no laboral), para el uso de diversos algoritmos de aprendizaje supervisado tales como: RF, SVM, XGboost y KNN.
- Seleccionar diversas técnicas de procesamiento del lenguaje natural como: TF IDF, LSA, Word2vec y GloVe, para caracterizar los chats laborales y no laborales.
- Evaluar diferentes algoritmos de clasificación para distinguir entre chats laborales y no laborales.
- Implementar un entorno de visualización interactiva, mediante power BI, con el fin de analizar un conjunto de chats.

3. Marco teórico

El principal problema al momento de analizar documentos o mensajes de manera automática es la naturaleza no estructurada de estos datos. Enfoques basados en NLP, buscan encontrar una representación númerica a los textos de manera que puedan ser analizados mediante diferentes estrategias de aprendizaje de máquina. Una de las primeras etapas consiste en el preprocesamiento, donde se busca principalmente mitigar el ruido que pueden tener estos datos y mantener la información relevante para el problema que se busca analizar.

3.1 Preprocesamiento

Aunque existen diferentes técnicas de preprocesamiento, las más relevantes en diferentes problemas de NLP son: la lematización, la eliminación de palabras de parada (del inglés, stop words), la eliminación de caracteres especiales y convertir todas las letras a minúsculas.
Una de las tareas más importantes en el preprocesamiento de texto es la tokenización, proceso mediante el cual el texto se divide en unidades más pequeñas, generalmente palabras, llamadas tokens. La tokenización permite separar cada mensaje en sus componentes básicos, lo que facilita la representación y análisis de las palabras de forma individual. Posteriormente, estos tokens serán representados numéricamente a través de diferentes técnicas de caracterización consideradas.

La lematización consiste en transformar los verbos conjugados a su forma como infinitivo, de esa manera se reduce el tamaño del vocabulario y se simplifica la representación. Por ejemplo, "corriendo" se lematiza a "correr" (Obando et al., 2020). Por otra parte, la eliminación de palabras de parada elimina términos de bajo contenido semántico que aparecen de manera frecuente pero que no aportan información distintiva, tales como “de”, “la”, “con”, entre otros. Estos términos suelen ser descartados ya que, al ser demasiado comunes generan ruido y afectan la precisión en modelos (Angélica, 2024). Otros métodos muy utilizados son la eliminación de caracteres especiales, este proceso consiste en quitar símbolos alfanuméricos esto incluye signos de puntuación, caracteres especiales (como @, #, $, /, etc.) (Ángel, 2024).

3.2 Caracterización

Luego del preprocesamiento del texto, la caracterización permite transformar los documentos en representaciones numéricas. Para este fin, se utilizan técnicas como BoW, TF-IDF, LSA y Word2Vec. BoW representa el texto según la frecuencia de las palabras, sin tener en cuenta el orden, convirtiendo cada documento en un vector en el que cada posición corresponde a una palabra del vocabulario y su frecuencia de aparición en un documento (Hamid et al., 2020). En este contexto, el vocabulario se refiere al conjunto de todas las palabras únicas que aparecen en todos los documentos. BoW ha demostrado ser efectiva en numerosas aplicaciones de clasificación de texto, como la clasificación de correos electrónicos o el análisis de sentimientos, debido a su simplicidad y su capacidad para identificar temas recurrentes (Hasan et al., 2019). Sin embargo, la técnica BoW puede generar un sesgo hacia palabras que aparecen con alta frecuencia en los documentos, pero que realmente no aportan información significativa para la tarea de clasificación. Para mitigar esta limitación, surge la técnica de TF-IDF, que genera representaciones vectoriales de documentos de un corpus. Esta técnica se divide en dos componentes: TF (frecuencia de término) mide la frecuencia de una palabra en un documento, mientras que IDF (frecuencia inversa de documento) mide su rareza en el corpus (Miguel et al., 2022). Esta ponderación en la frecuencia de aparición permite que palabras frecuentes en un documento y poco comunes en todos los documentos tengan un mayor valor comparado con las palabras que son frecuentes en un documento pero que también lo son en el resto de los documentos. TF-IDF resulta especialmente útil cuando se necesita diferenciar documentos con vocabularios similares, ya que prioriza los términos que aportan una mayor cantidad de información (Soufyane et al., 2021).

Otra de las caracterizaciones de texto más relevantes es LSA, que permite reducir la dimensionalidad del espacio de características al identificar patrones semánticos cercanos. Mediante la factorización de matrices usando la descomposición de valores singulares (SVD, del inglés Singular Value Decomposition), LSA permite el agrupamiento de palabras que comparten un significado similar, generando una representación conceptual del texto que es menos sensible al ruido y más robusta ante las variaciones en el lenguaje natural (Rodriguez-Bazan et al., 2020), en comparación con métodos más simples como BoW. Esta técnica ha sido ampliamente utilizada en tareas de clasificación de texto y recuperación de información, ya que permite encontrar relaciones latentes entre palabras y temas, lo cual es crucial en el análisis de textos complejos donde la sinonimia es frecuente (Chiru et al., 2014).

Aunque los enfoques previos han mostrado un buen desempeño en el análisis de documentos, existen enfoques recientes que permiten obtener representaciones vectoriales preservando las relaciones semánticas de las palabras, entre ellos uno de los enfoques más simples son los denominados word embeddings como Word2Vec. Esta técnica transforma palabras en embeddings, preservando tanto las relaciones sintácticas como las semánticas. A diferencia de representaciones como BoW, donde las representaciones finales se tratan de manera independiente, Word2Vec posiciona palabras con contextos similares en puntos cercanos del espacio vectorial, por esta razón se dice que es una representación vectorial que mantiene la relación semantica de las palabras.

Estos word embeddings pueden ser obtenidos mediante dos estrategias: el modelo de bolsa de palabras continua (CBOW, del inglés continous bag of words) y el modelo Skip-Gram. CBOW toma el contexto de cada palabra como entrada para predecir la palabra objetivo. En contraste, Skip-Gram utiliza una palabra objetivo para predecir su contexto, siendo eficaz en la representación de palabras raras y en conjuntos de datos más pequeños (Karani, 2022). Estas representaciones vectoriales mejoran significativamente el análisis semántico y resultan valiosas en tareas de clasificación de texto, ya que permiten descubrir patrones y conexiones profundas entre palabras y conceptos dentro del corpus.

3.3 Clasificación

La etapa de clasificación es una fase fundamental en el análisis de textos, y para ello se emplean diferentes algoritmos que pueden ser de aprendizaje supervisado o no supervisado, en este trabajo solo serán considerados métodos de aprendizaje supervisado. SVM, KNN, RF y XGBoost son algunos de los algoritmos seleccionados para esta tarea.

3.3.1 SVM

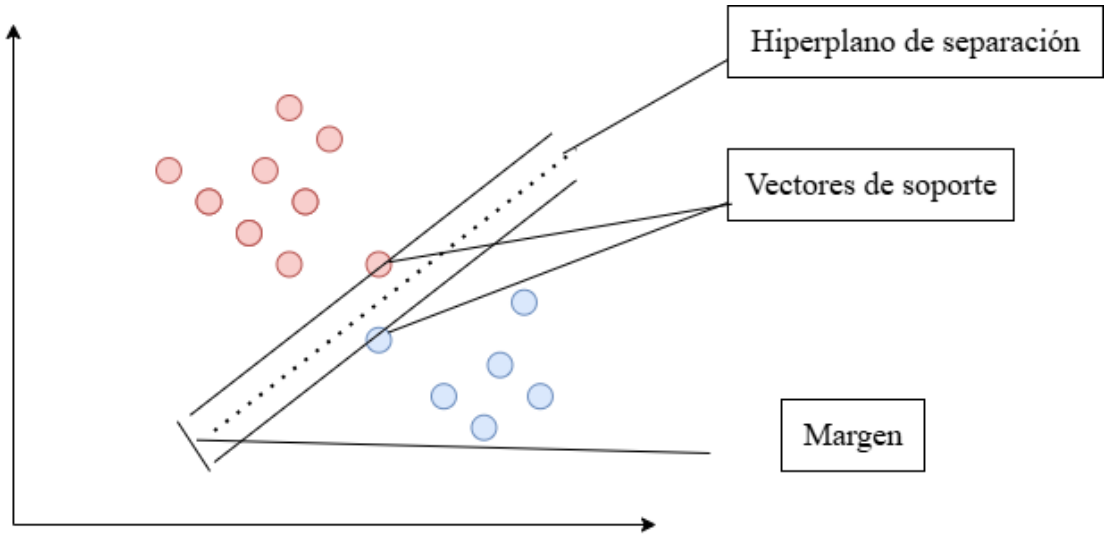
SVM es un algoritmo de clasificación supervisado que busca encontrar un hiperplano de separación lineal que divida las diferentes clases en el espacio de características. Este algoritmo tiene como objetivo maximizar el margen de separación entre clases, como se muestra en la Figura 1. El

hiperplano de separación es una línea o un plano que divide los datos en el espacio de características de tal manera que las diferentes clases queden separadas en lados opuestos. El margen es la distancia entre el hiperplano y los vectores de soporte. Maximizar este margen permite mejorar la capacidad de generalización del modelo. Este algoritmo permite introducir un margen de error controlado mediante el parámetro de regularización C. En el caso de un kernel Gaussiano, se añade un hiperparámetro llamado Gamma, el cual controla la influencia de cada muestra en el espacio de características. En el caso del kernel lineal, no se utiliza este parámetro.

Las SVM son particularmente eficaces en problemas de clasificación binaria y han mostrado un excelente rendimiento en problemas de clasificación de texto, especialmente en problemas donde el espacio de representación tiene una alta dimensionalidad (Gupta et al., 2021).

Figura 1.

Representación gráfica SVM con kernel lineal.



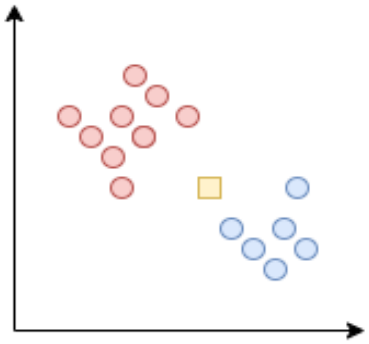
3.3.2 KNN

El algoritmo KNN es un método basado en instancias que clasifica un dato en función de las clases de sus vecinos más cercanos en el espacio de características. La Figura 2 ilustra este algoritmo. En este algoritmo, la predicción de una nueva muestra se determina a partir de la clase predominante en los k vecinos más cercanos, los cuales son definidos mediante una métrica de distancia como la euclidiana o la de manhattan. Además, el algoritmo puede utilizar diferentes criterios para ponderar

la influencia de los vecinos, como el uso de pesos uniformes o ponderados por distancia, donde los vecinos más cercanos tienen mayor peso en la predicción. Aunque KNN es fácil de implementar y entender, puede ser menos eficiente en grandes volúmenes de datos, ya que su rendimiento tiende a disminuir a medida que aumenta el número de puntos de datos en el espacio de características (Riza et al., 2023).

Figura 2.

Representación gráfica KNN



3.3.3 RF

El algoritmo de bosques aleatorios se basa en la construcción de múltiples árboles de decisión, donde cada árbol se entrena utilizando una muestra aleatoria del conjunto de datos original (García, 2024). Además, en cada nodo del árbol, se selecciona aleatoriamente un subconjunto de características para tomar decisiones sobre la división, lo que introduce mayor variabilidad y diversidad entre los árboles. Esto permite que cada árbol aprenda diferentes patrones de los datos, lo que aumenta la capacidad de generalización del modelo Al final, el algoritmo promedia las predicciones de todos los árboles para obtener la predicción final (Vijay et al., 2020). Los parámetros clave que afectan el rendimiento del modelo incluyen el número de árboles a construir, la profundidad máxima de cada árbol, el número máximo de características a considerar en cada división y el número mínimo de muestras por hoja, los cuales son fundamentales para controlar la complejidad y la capacidad del modelo.

3.3.4 XGBoost

XGBoost es un algoritmo de boosting que ajusta iterativamente modelos para mejorar las predicciones de clasificación. Comienza con un modelo base, típicamente un árbol de decisión y luego construye árboles adicionales para corregir los errores cometidos por los árboles anteriores (Petropoulos & Siakoulis, 2021). Cada árbol posterior se enfoca en las instancias mal clasificadas, ajustando el modelo para minimizar el error residual utilizando técnicas como el gradiente descendente (Van Rousselt, 2020). Su eficacia y escalabilidad en tareas de clasificación y regresión lo convierten en una elección popular en competiciones de machine learning, especialmente cuando se trabaja con una gran cantidad de datos (Razzak et al., 2023).

En comparación XGBoost y RF suelen tener un mayor costo computacional que modelos basados en SVM o KNN, pero en algunas tareas pueden lograr resultados comparables con metodologías más modernas como los modelos basados en aprendizaje profundo. Por otro lado, las SVMs suelen obtener mejores desempeños en problemas donde se tiene una alta dimensionalidad.

3.4 Validación

Una de las etapas más importantes en el reconocimiento de patrones es la estrategia de validación. En esta etapa se busca evaluar el desempeño del modelo frente a nuevos datos. La validación cruzada es una técnica utilizada para medir la precisión y estabilidad de un modelo, dividiendo el conjunto de datos en varios pliegues. En cada iteración, el conjunto de datos se separa en dos partes: una para entrenamiento y otra para prueba (Zhang et al., 2018). El modelo se entrena utilizando el conjunto de entrenamiento y se evalúa con el conjunto de prueba, que no se ha utilizado durante el entrenamiento. Este proceso se repite múltiples veces, asegurando que todos los datos, en alguna iteración, pertenezcan al conjunto de prueba, mientras que los datos restantes se usan para el entrenamiento. De este modo, cada subconjunto de datos participa tanto en el entrenamiento como en la evaluación en diferentes iteraciones.

Una de las estrategias de validación más rigurosas es la validación cruzada anidada. En este proceso, el primer nivel de la validación cruzada anidada, conocido como el bucle externo, se utiliza para evaluar el rendimiento del modelo en un conjunto de datos de prueba. Este bucle

externo garantiza que los resultados obtenidos sean una representación confiable del rendimiento del modelo frente a datos no vistos (Joel et al., 2022). El conjunto de datos se divide en múltiples particiones. Dentro de cada división del bucle externo, se realiza una validación cruzada adicional, conocida como el bucle interno, que optimiza los hiperparámetros del modelo utilizando el conjunto de entrenamiento de esa partición. El bucle interno en este estudio también utiliza validación cruzada estratificada, lo que significa que cada partición mantiene la misma proporción de muestras de cada clase que el conjunto de datos original. Esto asegura que la distribución de las clases sea representativa en los conjuntos de entrenamiento y prueba. Este enfoque es clave para evitar el sobreajuste y asegurar que los hiperparámetros seleccionados sean los más adecuados para el modelo, sin introducir sesgo hacia los datos de entrenamiento (Funes et al., 2022).

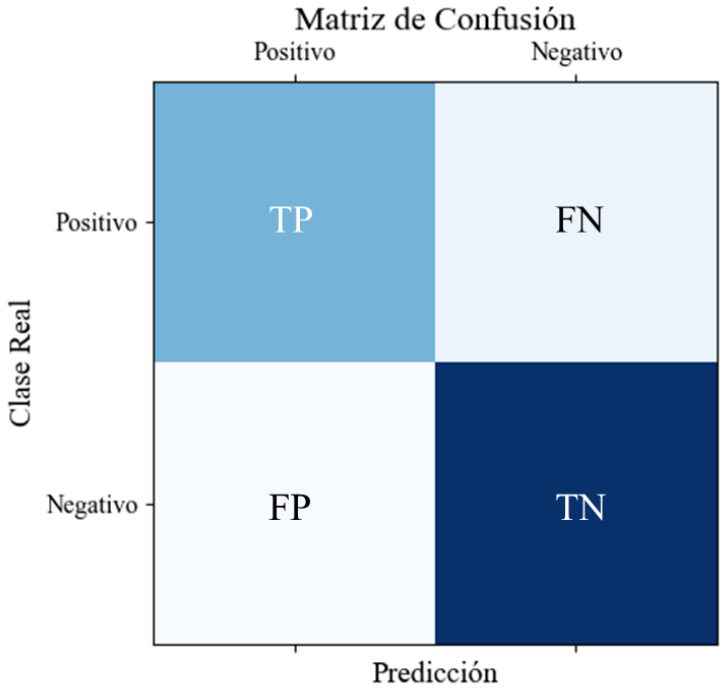
3.5 Métricas de desempeño

La evaluación del rendimiento de los modelos de clasificación se realiza utilizando varias métricas de desempeño, las cuales permiten una comprensión detallada de la capacidad del modelo para diferenciar entre las clases de mensajes laborales y no laborales. Una de las herramientas más usada para evaluar el desempeño de un modelo de clasificación es la matriz de confusión, la cual es un método de visualización para mostrar el desempeño del modelo en las diferentes clases (Santos et al., 2019). En el caso de un problema de clasificación binaria, la matriz de confusión se compone de:

- Verdaderos Positivos (TP, del inglés True Postives): Son los casos en los que el clasificador predice correctamente las muestras de la clase considerada como positiva.
- Falsos Positivos (FP, del inglés False Positives): Son los casos en los que el clasificador predice que una muestra es positiva cuando en realidad pertenecía a la clase considerada como negativa.
- Verdaderos Negativos (TN, del inglés True Negatives): Son los casos en los que el clasificador predice correctamente las muestras de la clase considerada como negativa.
- Falsos Negativos (FN, del inglés False Negatives): Son los casos en los que el clasificador predice que una muestra pertenece a la clase negativa cuando en realidad pertenece a la clase positiva.

Figura 3.

Entendimiento de los componentes de la matriz de confusión



Como se observa en la Figura 3, se tiene un ejemplo de una matriz de confusión, a partir de estos valores, se pueden calcular varias métricas importantes para evaluar el rendimiento del modelo.

Precisión: También conocido como Accuracy, es una de las métricas más comunes y mide el porcentaje de predicciones correctas realizadas por el modelo (Amin et al., 2023). Es la relación entre el número total de predicciones correctas y el número total de predicciones realizadas. Se define como:

$$Precisión = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (1)$$

Sensibilidad: También se conoce como recall y mide la habilidad del modelo para predecir las muestras de la clase positiva. Se define como:

$$Sensibilidad = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad (2)$$

Especificidad: También conocido como Specificity, mide la capacidad del modelo para identificar correctamente los mensajes de la clase negativa. Se define como:

$$Especifidad = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad (3)$$

ROC: También conocida como la curva característica operativa del receptor (del inglés, Receiver Operating Characteristic), es una herramienta gráfica que representa la relación entre la tasa de verdaderos positivos (sensibilidad) y la tasa de falsos positivos (1 - especificidad) para diferentes umbrales de clasificación (Hung et al., 2017). El área bajo la curva (AUC, del inglés Area Under Curve) proporciona una medida de la capacidad del modelo para discriminar entre las clases positiva y negativa. Un valor de AUC cercano a 1 indica un excelente rendimiento del modelo.

UAR: También conocida como el promedio de sensibilidad sin ponderar (del inglés, Unweighted Average Recall) es útil cuando las clases están desbalanceadas. Este promedio calcula la sensibilidad de cada clase individualmente, sin considerar el tamaño de cada clase, y luego toma el promedio. Se define como:

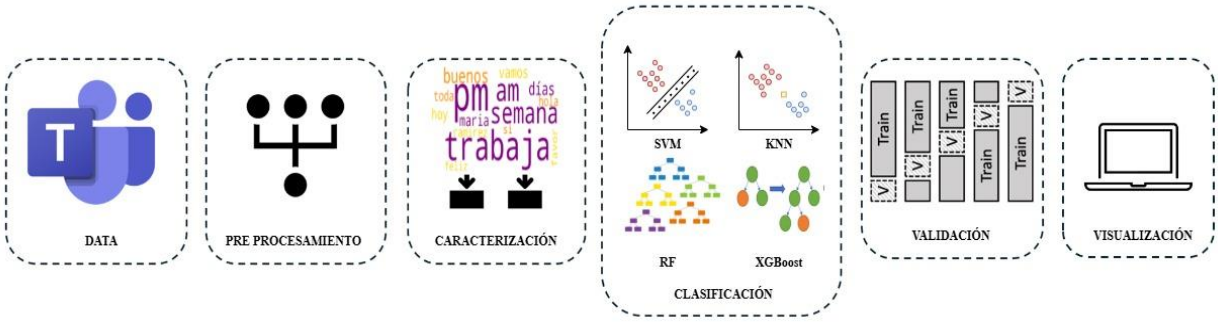
$$UAR = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}\right) (4)$$

4. Metodología

La Figura 4 resume la metodología abordada en este trabajo. Primero se construye la base de datos compuesta por mensajes laborales y no laborales. Luego se realiza una etapa de preprocesamiento. Mas adelante se exploran diferentes estrategias de caracterización para obtener una representación matemática de los diferentes documentos. Una vez se tiene las representaciones vectoriales de los documentos, se exploran diferentes algoritmos de clasificación y para medir el desempeño del modelo se usa la estrategia de validación k-fold. Finalmente, estos resultados se integran a una plataforma de visualización.

Figura 4.

Metodología de trabajo

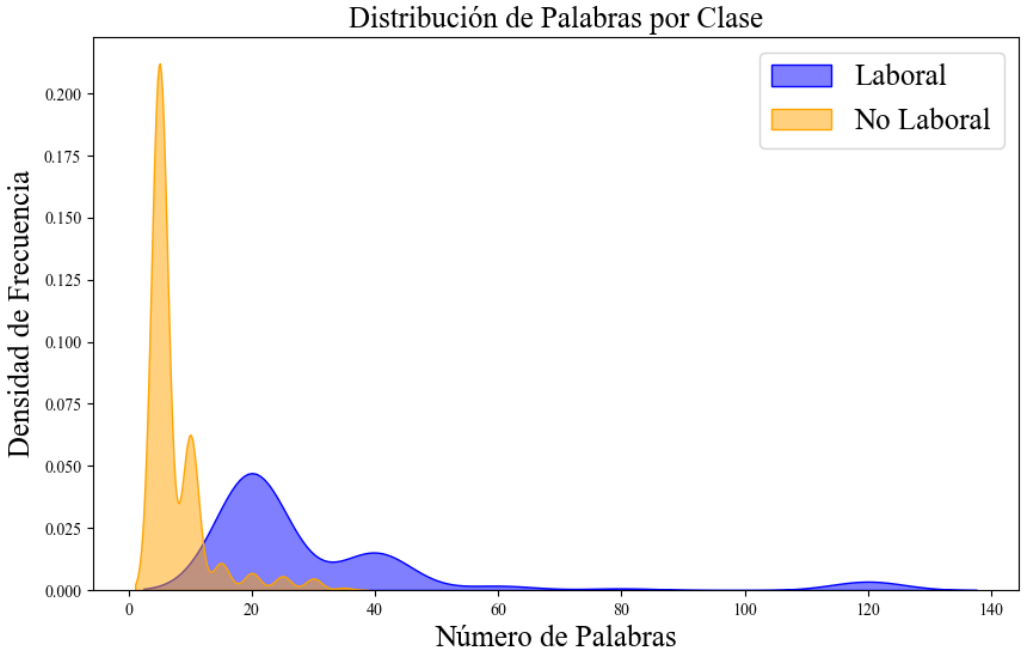


4.1 Data

La base de datos está compuesta por 2000 mensajes reales provenientes de interacciones entre los empleados mediante la plataforma Microsoft Teams. Estos mensajes fueron etiquetados manualmente siguiendo dos etiquetas: laboral y no laboral. Un mensaje es etiquetado como laboral si su contenido está relacionado con temas como: la empresa, cumplimiento de metas, reuniones, proyectos, entre otras y es etiquetado como no laboral si el contenido del mensaje tiene tópicos como: temas personales, películas, música, deportes y otros tópicos relacionados. En total se tienen 1000 mensajes por cada clase. La Figura 5 muestra la distribución de palabras por clase, los mensajes de la clase laboral se componen de 43 palabras en promedio, mientras que en promedio la clase no laboral está compuesta de 12 palabras.

Figura 5.

Distribución de palabras por clase (laboral vs no laboral)



La nube de palabras presentada a continuación en la Figura 6 muestra las palabras más utilizadas en los mensajes, destacando aquellas con mayor frecuencia. En esta visualización, el tamaño de cada palabra está directamente relacionado con su frecuencia, de modo que las palabras más repetidas aparecen en un tamaño más grande. Por ejemplo, la palabra “pm” se repite 1290 veces, mientras que “feliz”, aunque también aparece con cierta frecuencia, se presenta en un tamaño más pequeño debido a su menor número de repeticiones, con solo 231 apariciones en los mensajes.

Figura 6.

Nube de palabras más frecuentes para la clase laboral y no laboral

4.2 Preprocesamiento

Para el preprocesamiento de cada mensaje se utilizó la librería nltk (del inglés, Natural Language Toolkit), una de las herramientas más usadas para el procesamiento de lenguaje natural en Python. En primer lugar, se utilizó la función de lematización de nltk para reducir cada palabra a su forma raíz (Wang & Hu, 2021). Adicionalmente, se emplearon las funciones de nltk para eliminar las palabras bloqueantes del idioma español. Otra tarea realizada en esta etapa fue la eliminación de caracteres especiales. Esto incluye la eliminación de puntuación, números y símbolos innecesarios, los cuales podrían interferir en el análisis del texto.

4.3 Caracterización

En este proyecto, se emplearon varias técnicas para caracterizar los mensajes, comenzando con métodos clásicos como BoW y TF-IDF, que asignan pesos a las palabras basados en su frecuencia en el corpus. Para mejorar la captura de relaciones semánticas y reducir la dimensionalidad de estas representaciones, se aplicó LSA sobre las matrices generadas por BoW y TF-IDF. Todas estas representaciones fueron generadas usando la librería sklearn disponible abiertamente en Python.

Con el objetivo de utilizar representaciones más recientes se exploró el uso de embeddings basados en Word2Vec. En este trabajo se usó un modelo ya pre entrenado de Word2Vec (Hugging Face,

2001). Este modelo fue entrenado con Wikicorpus, el cual este compuesto por documentos provenientes de Wikipedia en español. Este modelo tiene las siguientes características: la dimensión de los embeddings es de un tamaño de 300, una ventana de tamaño 5 y 10 iteraciones. Este modelo permite obtener una representación vectorial para cada palabra. Luego, la representación vectorial de cada mensaje es calculada como el vector promedio de las representaciones de las palabras que componen el mensaje.

4.4 Clasificación

En la fase de clasificación del proyecto, se emplearon cuatro algoritmos de aprendizaje supervisado ampliamente utilizados para tareas de clasificación de texto: KNN, SVM, RF y XGBoost. Cada uno de estos algoritmos fue implementado y evaluado para determinar cuál ofrece el mejor rendimiento para la clasificación de los mensajes en las categorías de laboral y no laboral.

Para cada uno de estos algoritmos debe definirse un conjunto de hiperparametros, los cuales, típicamente, son seleccionados de manera experimental. En este trabajo se usó una malla de búsqueda para cada uno de los algoritmos. El objetivo es encontrar la combinación de hiperparámetros que maximice el rendimiento de cada modelo, evitando problemas como el sobreajuste o el subajuste. Los hiperparámetros optimizados en este trabajo para cada uno de los modelos son:

- **KNN**: El número de vecinos k (3, 5, 7, 9), los pesos (uniforme y por distancia) y la métrica de distancia utilizada (en este caso, Euclidiana o Manhattan).
- **SVM**: El parámetro de regularización C (0.1, 1, 10, 100) y el valor de Gamma (0.001, 0.01, 0.1, 1) y el kernel utilizado, que fue el RBF.
- **RF**: El número de árboles (100, 200, 300), la profundidad máxima de los árboles (10, 20, 30, None), el parámetro del máximo de características a considerar (auto y sqrt) y el número mínimo de muestras por hoja (2, 5, 10).
- **XGBoost**: El número de árboles (100, 200, 300), la profundidad máxima de los árboles (10, 20, 30, None), la tasa de aprendizaje (0.001, 0.1, 0.2, 0.3) y el peso mínimo de la muestra (1, 3, 5).

4.5 Validación

En esta etapa, se utilizó una validación cruzada anidada con 4 pliegues externos y 3 pliegues internos. El bucle externo se repitió 2 veces utilizando el método StratifiedKFold. La repetición del bucle externo se realizó con el fin de obtener una evaluación más robusta del modelo.
Las métricas de desempeño se estiman a partir de los pliegues externos. En cada repetición, la precisión se calcula como el promedio de los resultados obtenidos en cada uno de los pliegues del bucle externo, con el fin de describir el comportamiento del modelo más ajustado a cómo será su comportamiento con un nuevo conjunto de datos.

4.6 Visualización

Para la visualización de los resultados, se construyó un tablero interactivo utilizando Power BI, una herramienta versátil y poderosa para la creación de gráficos y consolidación de datos. Además de los resultados obtenidos a partir de las métricas de desempeño de los modelos, se incluyeron otros datos relevantes para la empresa, que fueron extraídos mediante consultas SQL. Estas consultas proporcionaron información detallada sobre proveedores, bots, empleados y equipos de trabajo en diferentes países. El tablero permite visualizar información adicional que proporciona contexto al análisis, como la distribución de los mensajes por área interna. También se incluyó un desglose de los mensajes según los días laborales, permitiendo observar el volumen de mensajes en diferentes períodos y facilitar la identificación de patrones en la comunicación laboral.

5. Análisis de resultados

5.1 Clasificación entre mensajes laborales y no laborales

La Tabla 1 muestra los resultados obtenidos para cada algoritmo de clasificación usando las diferentes estrategias de caracterización consideradas, donde se puede observar que los algoritmos de clasificación muestran precisiones superiores a 90%.

Tabla 1.

Resultados de cada clasificador por técnica de caracterización

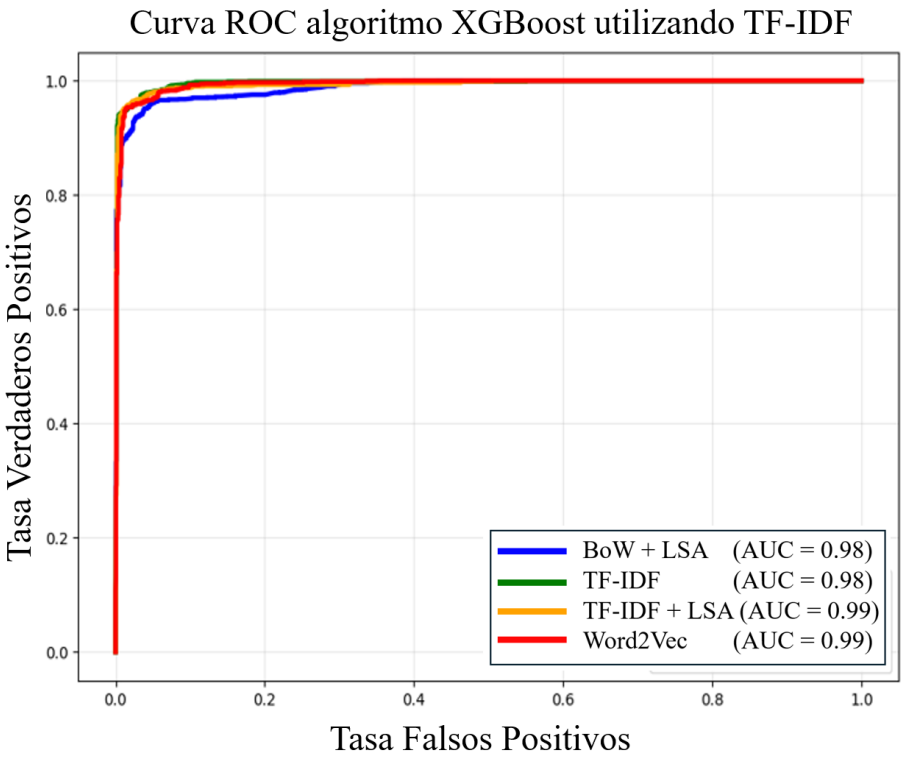
SVM					
Técnica de caracterización	Precisión	Sensibilidad	Especificidad	Promedio de sensibilidad sin ponderar	Área bajo la curva
BoW-LSA	0,86	0,76	0,86	0,86	0,93
TF IDF - LSA	0,93	0,95	0,93	0,93	0,98
TF IDF	**0,96**	**0,98**	**0,96**	**0,96**	**0,98**
Word2Vec	0,87	0,86	0,87	0,87	0,94
RF					
Técnica de caracterización	Precisión	Sensibilidad	Especificidad	Promedio de sensibilidad sin ponderar	Área bajo la curva
BoW-LSA	0,95	0,94	0,95	0,95	0,98
TF IDF - LSA	0,96	0,95	0,96	0,96	0,99
TF IDF	**0,97**	**0,96**	**0,97**	**0,97**	**0,99**
Word2Vec	0,96	0,96	0,96	0,96	0,99
KNN					
Técnica de caracterización	Precisión	Sensibilidad	Especificidad	Promedio de sensibilidad sin ponderar	Área bajo la curva
BoW-LSA	0,93	0,92	0,93	0,93	0,99
TF IDF - LSA	0,94	0,95	0,94	0,94	0,99
TF IDF	**0,96**	**0,95**	**0,96**	**0,96**	**0,99**
Word2Vec	0,95	0,97	0,95	0,95	0,99
XGBoost					
Técnica de caracterización	Precisión	Sensibilidad	Especificidad	Promedio de sensibilidad sin ponderar	Área bajo la curva
BoW-LSA	0,97	0,96	0,97	0,97	0,99
TF IDF - LSA	0,96	0,96	0,96	0,96	0,99
TF IDF	**0,97**	**0,96**	**0,97**	**0,97**	**0,99**
Word2Vec	0,96	0,95	0,96	0,96	0,99

5.2 Comparación de los algoritmos

En general, los modelos de XGBoost y RF alcanzaron los mejores resultados, especialmente cuando se utilizó la técnica de TF-IDF, logrando un AUC-ROC de 0.99, lo que indica la capacidad de discriminación entre las clases. Además, los modelos TF-IDF - LSA y TF-IDF se destacaron por obtener los mejores resultados en las métricas de sensibilidad y especificidad. En cuanto a la sensibilidad, el algoritmo XGBoost fue el que mostró la mejor capacidad de detección, especialmente al combinarlo con TF-IDF. TF-IDF mostró un excelente balance entre especificidad y sensibilidad, destacándose en los modelos de XGBoost y RF, donde alcanzaron especificidades superiores al 95% y sensibilidades cercanas al 98%. Aunque Word2Vec también mostró buenos resultados, no alcanzó el mismo nivel de precisión en comparación con TF-IDF. Por otro lado, XGBoost y RF lograron una alta especificidad (0.97-0.98), lo que indica una sólida capacidad para identificar los mensajes no laborales.

Figura 7.

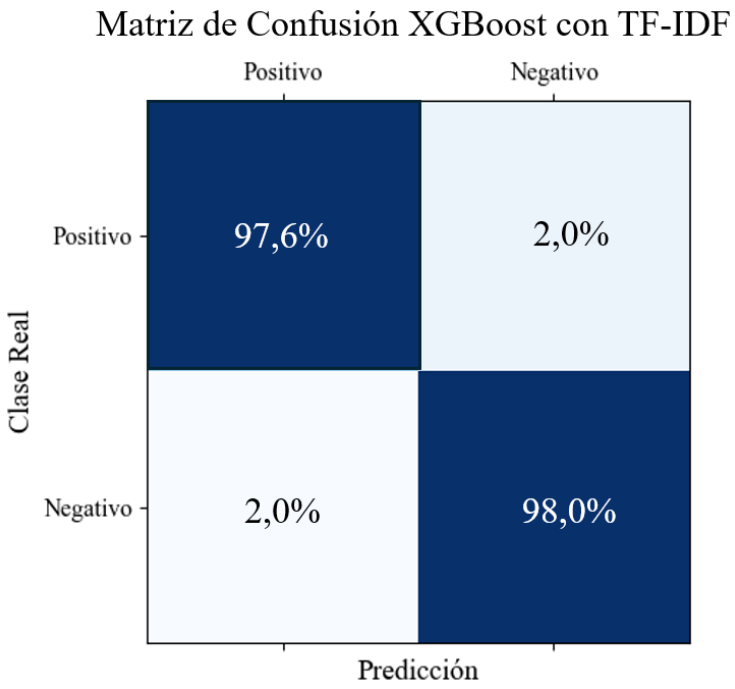
Curva ROC algoritmo XGBoost utilizando TF-IDF



La Figura 7 muestra la curva ROC obtenida con los modelos basados en XGBoost considerando diferentes caracterizaciones. Los resultados muestran un AUC de 0.98 para BoW con LSA y 0.99 para las demás técnicas.

Figura 8.

Matriz de confusión algoritmo XGBoost usando caracterización TF-IDF



En la figura 8, muestra un alto porcentaje de verdaderos positivos y verdaderos negativos. Los falsos positivos y los falsos negativos son bajos, lo que indica que el clasificador tiene un alto desempeño en la tarea de distinguir entre los mensajes laborales y no laborales.

Por otro lado, la técnica Word2Vec resultó ser menos eficiente en comparación con TF-IDF y BoW con LSA en los modelos de SVM y RF, aunque mostró un rendimiento competitivo en modelos como KNN y XGBoost, pero Word2vec es la caracterización que en los 4 algoritmos de clasificación presentaba el menor tiempo de ejecución como se observa en la Tabla 2. Es importante aclarar que XGBoost y RF representaron un costo computacional más alto. Estos algoritmos configuran un aumento en el tiempo de ejecución de entre 56.88% y 95.93% en comparación con KNN y SVM, dependiendo de la técnica de caracterización utilizada.

Tabla 2.

Tiempos de ejecución por clasificador

Técnica de caracterización	SVM	RF	K-NN	XGBoost
BoW-LSA	1286 s	2165 s	1380 s	2520 s
TF IDF - LSA	2806 s	4621 s	3780 s	8281 s
TF IDF	4562 s	9015 s	6360 s	9724 s
Word2Vec	574 s	4140 s	603 s	5340 s

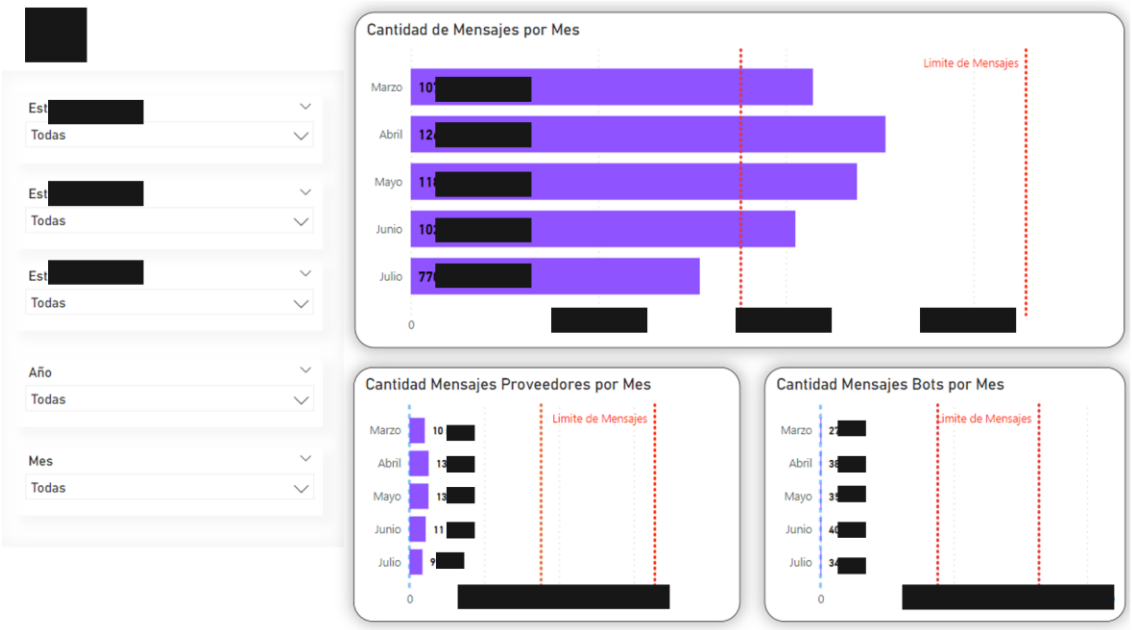
5.3 Visualización

A continuación, se presentan las imágenes del tablero de Power BI implementado. Por motivos de confidencialidad, algunas cifras y nombres han sido censurados. El diseño del tablero incluye filtros interactivos que permiten buscar y visualizar los flujos de mensajes según diferentes niveles de la estructura organizacional de la compañía. Esto facilita el análisis detallado del comportamiento de los mensajes a través de las distintas áreas.

Las gráficas mostradas no solo reflejan la cantidad de mensajes, sino también incluyen ayudas visuales, como límites de mensajes que pueden presentarse por mes. En caso de que este límite sea sobrepasado, las gráficas cambian de color como una alerta para señalar posibles riesgos. Además, se incluyen gráficos que muestran los resultados de los modelos de clasificación, reflejando las métricas de precisión y la cantidad de palabras por clase clasificada, lo que permite una comprensión más profunda de los mensajes que intercambian los equipos y cuales tienen un contenido laboral o no laboral.

Figura 9.

Vista tablero Power BI resumen de mensajes por mes



En la Figura 9, se puede observar que la mayor cantidad de mensajes se envía durante el horario laboral y en días laborales, lo que podría reflejar un patrón de comunicación típico dentro de la organización.

Figura 10.

Vista tablero Power BI mensajes a partir del horario laboral

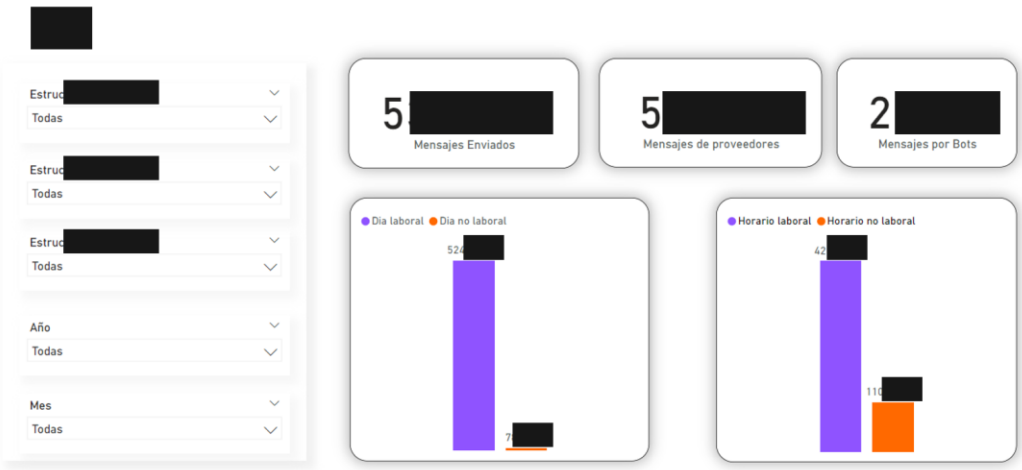
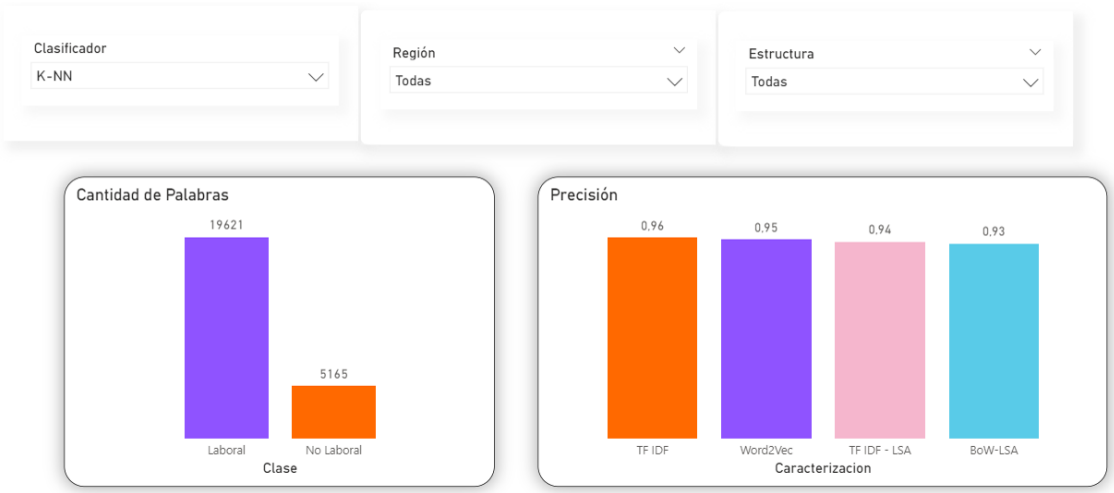


Figura 11.

Vista tablero Power BI resultados clasificadores



En la Figura 11, se puede ver cómo los filtros interactivos permiten cambiar los resultados de las caracterizaciones según el clasificador seleccionado, facilitando una visualización personalizada de los resultados. Agregado a esto con el fin de analizar el comportamiento del flujo de mensaje se incluyen filtros para hacer búsquedas por un área del banco o por una región, estos filtros son útiles para el análisis de las figuras 12 y 13 que se muestran a continuación:

Figura 12.

Vista tablero Power BI flujo de mensajes de la clase laboral por región y área

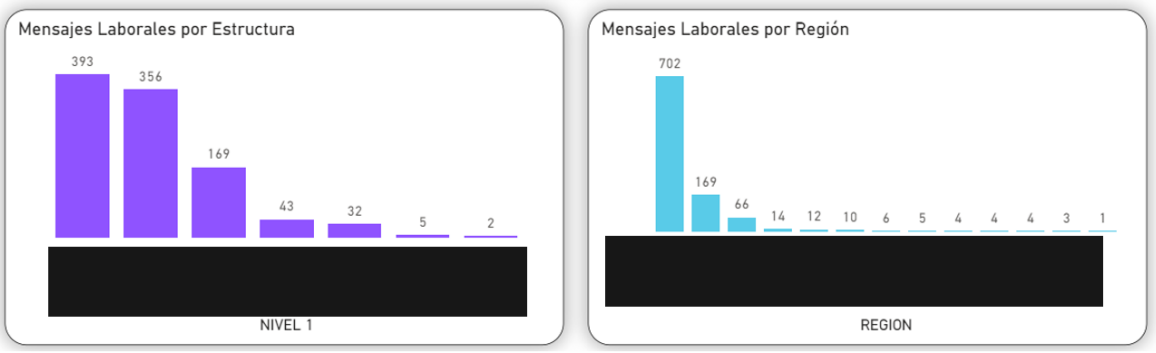
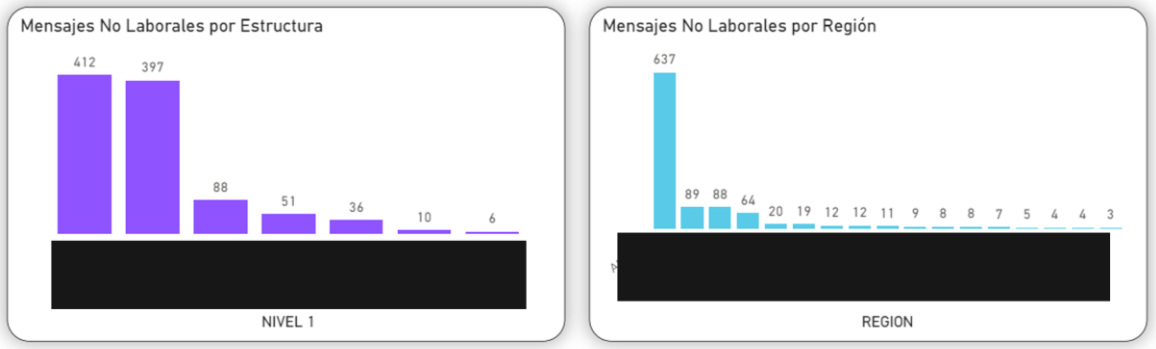


Figura 13.

Vista tablero Power BI flujo de mensajes de la clase no laboral por región y área



Las Figuras 12 y 13, permiten visualizar las áreas del banco que más envían mensajes de tipo laborales y no laborales, esto es importante conocerlo para poder realizar la priorización de áreas, donde aquellas que tengan una mayor cantidad de mensajes de tipo laboral tendrían una mayor prioridad versus aquellas áreas que tengan más mensajes no laborales.

De igual manera se incluye la distribución de mensajes por regiones, la mayor cantidad de mensajes tanto de la clase laboral o no laboral se presentan en Antioquia, este es un comportamiento normal, ya que la mayoría de los empleados se encuentran en esta región del país.

6. Conclusiones y recomendaciones

El presente trabajo ha demostrado la eficacia de diversas técnicas de procesamiento de lenguaje natural y algoritmos de clasificación en la tarea de distinguir entre mensajes laborales y no laborales. Los resultados obtenidos permiten extraer las siguientes conclusiones:

1. Los modelos de clasificación XGBoost y RF sobresalieron como los mejores rendimientos, alcanzando métricas de precisión, sensibilidad y especificidad superiores al 96% cuando se utilizaron técnicas de caracterización basadas en TF-IDF.
2. Aunque Word2Vec mostró un desempeño alto en la clasificación, su principal ventaja radica en su bajo costo computacional en comparación con TF-IDF y BoW-LSA. Esto lo convierte en una opción viable para escenarios en los que el tiempo de ejecución es una limitante, a pesar de no alcanzar los mismos niveles de precisión, comparado con otros algoritmos, este enfoque logra una precisión de hasta 93%
3. La implementación del tablero de Power BI complementa los resultados obtenidos, permitiendo un análisis del flujo de mensajes por región y área de la organización. Esta herramienta no solo facilita la visualización de métricas de clasificación, sino que también proporciona información clave para la priorización de áreas críticas, permitiendo la toma de decisiones.
4. Los resultados obtenidos con las técnicas de caracterización implementados permitieron cumplir con el objetivo de diferenciar los mensajes entre las clases laboral y no laboral. Debido al buen rendimiento de las 4 caracterizaciones utilizadas, no se requirió explorar otras técnicas de representación como GloVe o BERT.
5. Aunque los resultados son satisfactorios, en el futuro se podría pensar en un análisis más profundo acerca del contenido de los mensajes laborales. Estos mensajes laborales podrían ser categorizados según el proyecto o tema de interés al que pertenecen. Esto con el fin de tener un control más personalizado acerca de las interacciones de los colaboradores mediante la plataforma Teams.

Departamento de Ingeniería Electrónica y de Telecomunicaciones

MODELO ANALÍTICO PARA CLASIFICACIÓN DE MENSAJES LABORALES USANDO NLP

UNIVERSIDAD DE ANTIOQUIA
1803
Facultad de Ingeniería

PRACTICANTE: Camilo Orbes Cabrera

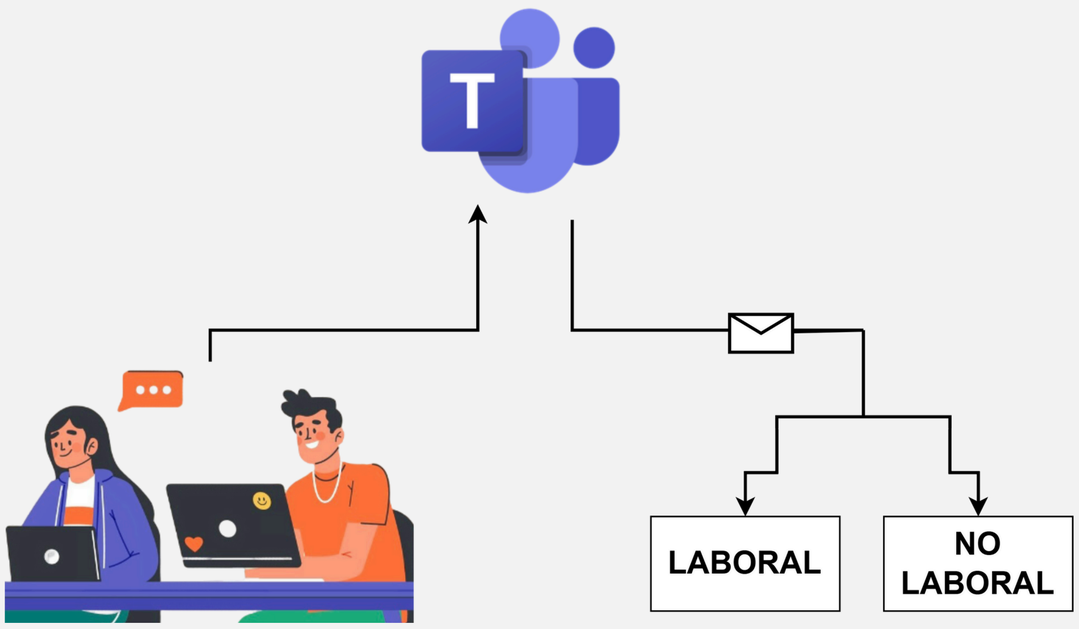
ASESORES: Daniel Escobar Grisales, David Esteban Betancur

PROGRAMA: Ingeniería Electrónica

Semestre de la práctica: 2024-2

Introducción

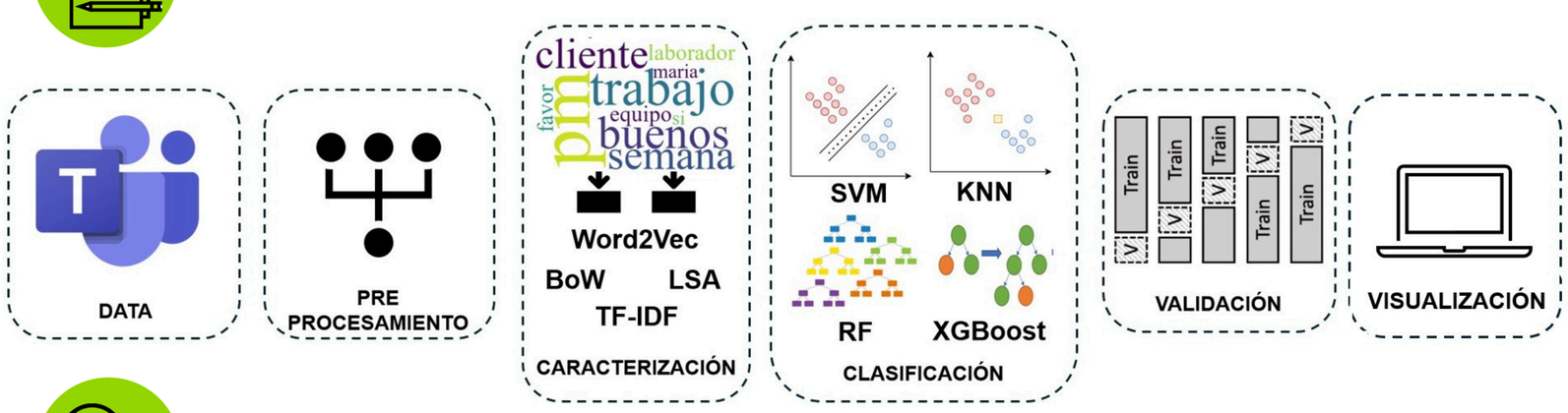
La comunicación mediante plataformas digitales es una práctica común en las empresas donde se tiene una gran cantidad de personal y donde todos los colaboradores podrían estar distanciados geográficamente. Plataformas como Microsoft Teams ofrecen servicios para la comunicación interna en una empresa, pero estos servicios tienen un costo asociado. En la empresa Bancolombia se ha evidenciado un sobrecosto respecto al intercambio de mensajes fuera del límite contratado. Dentro de los diferentes análisis realizados internamente en el banco, se ha evidenciado que muchos de los mensajes que se intercambian no tienen un contenido laboral. Este proyecto propone una estrategia basada en técnicas de PLN, para clasificar mensajes en laborales y no laborales.



Objetivos

- Etiquetar la base de datos usando dos categorías (laboral y no laboral), para el uso de diversos algoritmos de aprendizaje supervisado tales como: Bosques aleatorios, máquinas de soporte vectorial, XGboost y k vecinos más cercanos
- Seleccionar diversas técnicas de procesamiento del lenguaje natural como: TF IDF, LSA, Word2vec y GloVe, para caracterizar los chats laborales y no laborales.
- Evaluar diferentes algoritmos de clasificación para distinguir entre chats laborales y no laborales.
- mplementar un entorno de visualización interactiva, mediante power BI, con el fin de analizar un conjunto de chats.

Metodología



Resultados

En general, los modelos de XGBoost y RF alcanzaron los resultados mas precisos, especialmente cuando se utilizó la técnica de TF-IDF. Además, los modelos TF-IDF - LSA y TF-IDF se destacaron por obtener una sensibilidad y especificidad mas balanceada.

Clasificador RF

Técnica de caracterización	Precisión	Sensibilidad	Especificidad
BoW-LSA	93%	92%	93%
TF IDF - LSA	94%	95%	94%
TF IDF	96%	95%	96%
Word2Vec	95%	97%	95%
BoW-LSA	93%	92%	93%

Conclusiones

- Los modelos de clasificación XGBoost y RF sobresalieron como los mejores rendimientos, alcanzando métricas de precisión, sensibilidad y especificidad superiores al 96% cuando se utilizaron técnicas de caracterización basadas en TF-IDF.
- Aunque Word2Vec mostró un desempeño alto en la clasificación, su principal ventaja radica en su bajo costo computacional en comparación con TF-IDF y BoW-LSA.
- Los resultados obtenidos con las técnicas de caracterización implementados permitieron cumplir con el objetivo de diferenciar los mensajes entre las clases laboral y no laboral. Debido al buen rendimiento de las 4 caracterizaciones utilizadas, no se requirió explorar otras técnicas de representación como GloVe o BERT.

DATOS DE CONTACTO DEL AUTOR:

57 3015179965 camilo.orbes@udea.edu.co www.linkedin.com/in/camilo-o-191481116

Referencias

Amin, M. M. (2023). Will affective computing emerge from foundation models and general artificial intelligence? A first evaluation of ChatGPT. *IEEE Intelligent Systems*, 15-23.

Antony Vijay, J. A. (2020). A dynamic approach for detecting the fake news using random forest classifier and NLP. *Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 2*, 331-341.

Chiru, C. R. (2014). Comparison between LSA-LDA-lexical chains. *International Conference on Web Information Systems and Technologies*, 255-262.

Esnaola, L. e. (2019). Análisis comparativo de tareas de pre procesamiento de textos sobre contenido extraído de redes sociales. *XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan)*.

Fernández, C. (2021). Derecho a la desconexión digital: qué es y por qué es tan difícil respetarlo en tiempos de la COVID-19. *La Vanguardia*.

Funes, F. M. (2022). Geolocalización de usuarios en Twitter utilizando redes convolucionales de grafos.

García Vidal, J. F. (2024). ProspectAI-Modelos de Machine Learning y Preprocesamiento de Lenguaje Natural para la Clasificación Efectiva de Clientes.

García, M. R. (2024). Clasificación de reseñas de Amazon utilizando NLP y Random Forest.

Gupta, A. G. (2021). Machine learning classifier approach with gaussian process, ensemble boosted trees, SVM, and linear regression for 5g signal coverage mapping. *IJIMAI*, 156-163.

Hamid, A. e. (2020). Fake news detection in social media using graph neural networks and NLP Techniques: A COVID-19 use-case. *arXiv preprint*.

Hasan, M. R. (2019). Sentiment analysis with NLP on Twitter data. *IC4ME2*, 1-4.

Herrera Fernández, F. (2022). Análisis de redes sociales en Twitch. *tesis doctoral, Universitat Politècnica de València*.

Herrera Rojas, C. A. (2024). Análisis de textos utilizando técnicas de NLP: Análisis de las respuestas de los ciudadanos que participaron en la iniciativa “El Chile que Queremos”. *tesis doctoral, Universidad del Desarrollo. Facultad de Ingeniería*.

Hung, C. Y. (2017). Comparing deep neural network and other machine learning algorithms for stroke prediction in a large-scale population-based electronic medical claims database. *EMBC*, 3110-3113.

Karani, D. (2018). Introduction to word embedding and Word2Vec. *Towards Data Science*.

Landín Casal, e. a. (2022). Procesamiento de lenguaje natural y generación automática de alertas de las reseñas de clientes, en una empresa de telecomunicaciones del Ecuador. *tesis doctoral, ESPOL. FIEC*.

Peña Quintero, D. A. (2024). Desarrollo de un producto mínimo viable para la gestión de encuestas y medición de la calidad del servicio al cliente como complemento a la tecnología de Microsoft Teams. *tesis de licenciatura, Ingeniería de Sistemas*.

Petropoulos, A. &. (2021). Can central bank speeches predict financial market turbulence? Evidence from an adaptive NLP sentiment index analysis using XGBoost machine learning technique. *Central Bank Review*, 141-153.

Riza, L. S. (2023). Automatic generation of short-answer questions in reading comprehension using NLP and KNN. *Multimedia Tools and Applications*, 41913-41940.

Rodriguez-Bazan, H. S.-A. (2020). Revisión del estado del arte en técnicas de procesamiento de lenguaje natural para análisis de malware. *Res. Comput. Sci*, 1105-1115.

Romanisio, A. &. (2024). Predicción de la satisfacción del usuario a partir de chats de atención al cliente. *EJS*, 2-24.

Santos, R. G. (2019). An Overview of User Feedback Classification Approaches. *REFSQ workshops*, 357-369.

Soufyane, A. A. (2021). An intelligent chatbot using NLP and TF-IDF algorithm for text understanding applied to the medical field. *Emerging Trends in ICT for Sustainable Development: The Proceedings of NICE2020 International Conference*, 3-10.

Toba, H. e. (2024). Bloom-epistemic and sentiment analysis hierarchical classification in course discussion forums. *arXiv preprint*.

Van Rousselt, R. &. (2021). Natural language processing bots. *Pro Microsoft Teams Development: A Hands-on Guide to Building Custom Solutions for the Teams Platform*, 161-185.

Wang, M. &. (2021). The application of nltk library for python natural language processing in corpus research. *Theory and Practice in Language Studies*, 1041-1049.

Word2vec/wikipedia2vec_eswiki_20180420_300d. (11 de 03 de 2001). *Hugging Face*.

Zhang, D. e. (2018). A data-driven design for fault detection of wind turbines using random forests and XGboost. *IEEE Access*, 21020-21031.

.