MODELO DE CLASIFICACIÓN DE INCIDENCIAS UTILIZANDO APRENDIZAJE DE MÁQUINA PARA LA EMPRESA SIGMA INGENIERÍA S.A EN LA LÍNEA DE SOPORTE TÉCNICO

MARÍA XIMENA ARIAS BARAHONA

UNIVERSIDAD AUTÓNOMA DE MANIZALES

FACULTAD DE INGENIERÍA

MAESTRÍA EN INGENIERÍA

MANIZALES

2022

MODELO DE CLASIFICACIÓN DE INCIDENCIAS UTILIZANDO APRENDIZAJE DE MÁQUINA PARA LA EMPRESA SIGMA INGENIERÍA S.A EN LA LÍNEA DE SOPORTE TÉCNICO

Autora

MARÍA XIMENA ARIAS BARAHONA

Proyecto de grado para optar al título de Magíster en Ingeniería

Director (a):

Ph.D. Reinel Tabares Soto

Codirectores:

Ph.D. Simón Orozco Arias

M.Sc. Juan Camilo Flórez

UNIVERSIDAD AUTÓNOMA DE MANIZALES

FACULTAD DE INGENIERÍA

MAESTRÍA EN INGENIERÍA

MANIZALES

2022

DEDICATORIA

Le dedico todo mi esfuerzo y trabajo entregado para el desarrollo de esta tesis a mi familia, por ser el pilar más importante, por brindarme su amor incondicional y siempre creer en mí. A mi futuro esposo por su paciencia, por su comprensión, por su tiempo, por su apoyo en este proceso, por su amor, por ser tal y como es.

AGRADECIMIENTOS

Expreso mis agradecimientos a Reinel Tabares Soto, director de la tesis de maestría, por su apoyo incondicional, su tiempo y paciencia y su gran virtud de sacarme siempre de callejones sin salida.

Agradezco a Simón Orozco Arias y Juan Camilo Flórez, Co-directores de mi tesis, por el tiempo dedicado y sus valiosos aportes al desarrollo del proyecto.

Mis más sinceros agradecimientos al ingeniero Harold Brayan Arteaga Arteaga, por su apoyo incondicional en la ejecución de un sinfín de experimentos que permitió la publicación del artículo científico producto de la tesis y sus valiosos aportes al desarrollo de este trabajo.

Agradezco a la Universidad Autónoma de Manizales, al programa de Maestría en Ingeniería y todos los docentes por permitirme crecer a nivel académico y permitirme haber llegado hasta este momento tan importante en mi formación profesional.

Agradezco al equipo de SIGMA Ingeniería S.A; a Claudia Cuervo, Alejandra Restrepo y Mónica Cifuentes por todo su apoyo durante el proceso y a Mario Andrés Valencia por confiar en mí, por depositar una gran idea en mi mente que logramos hacer realidad y brindarme todo el apoyo para el desarrollo de este proyecto.

Gracias a la empresa SIGMA Ingeniería S.A y al Fondo de Becas de Investigación Manizales + Innovadora, creado por la Alcaldía de Manizales y Manizales Campus Universitario que me apoyaron en la financiación de esta tesis.

Agradezco a mi familia y a mi prometido por ser la fuente de inspiración y brindarme todo el apoyo durante los momentos más difíciles de este proceso.

RESUMEN

La Inteligencia Artificial es uno de los componentes reconocidos por su potencial para transformar radicalmente el modo en el que vivimos actualmente. Hace posible que las máquinas aprendan de la experiencia, se ajusten a nuevas aportaciones y realicen tareas emulando a seres humanos. Esta investigación se centra en el ámbito empresarial y propone el desarrollo de un modelo computacional de clasificación de incidencias utilizando Aprendizaje Automático (ML por sus siglas en inglés) y Procesamiento de Lenguaje Natural (NLP por sus siglas en inglés) enfocado al área de soporte técnico en una empresa de desarrollo de software que actualmente resuelve los requerimientos de los clientes de forma manual. A través de técnicas de ML y NLP aplicadas a los datos de la empresa, es posible conocer la categoría de una solicitud dada por el cliente. Este proceso permite aumentar la satisfacción del cliente al revisar los registros históricos para analizar su comportamiento y proporcionar correctamente el protocolo de solución esperada a las incidencias presentadas. Además, reduce el coste y el tiempo dedicado a la gestión relacional con el consumidor potencial. Este modelo evalúa diferentes técnicas de ML como Máquinas de Vectores de Soporte (SVM por sus siglas en inglés), Árboles Adicionales, Bosques Aleatorios, Regresión Logística, Árboles de Decisión, Análisis Discriminante Lineal, Naïve Bayes y K Vecinos más cercanos. El algoritmo SVM entrega el mayor desempeño, con un 90.47% de exactitud, una precisión de 91.21%, un *Recall* de 90.47% y un *F1-Score* de 90.29%, aplicando técnicas de preprocesamiento, balanceo de clases y optimización de hiperparámetros.

Palabras clave: Procesamiento del lenguaje natural, Aprendizaje Automático, Servicio al cliente, Clasificación de requerimientos, Clasificación de texto.

ABSTRACT

Artificial Intelligence is one of the components recognized for its potential to transform the way we live today radically. It makes it possible for machines to learn from experience, adjust to new contributions and perform tasks like human beings. The business field is the focus of this research. This paper proposes to implement an incident classification model using Machine Learning (ML) and Natural Language Processing (NLP). The application focuses on the technical support area in a software development company that currently resolves customer requests manually. Through ML and NLP techniques applied to company data, it is possible to know the category of a request given by the client. It increases customer satisfaction by reviewing historical records to analyze their behavior and correctly provide the expected solution to the incidents presented. Also, this practice would reduce the cost and time spent on relationship management with the potential consumer. This work evaluates different Machine Learning models, such as Support Vector Machines (SVM), Extra Trees, Random Forests, Logistic Regression, Decision Trees, Linear Discriminant Analysis, Naïve Bayes y K-Nearest Neighbors. The SVM algorithm demonstrates the highest performance, with 90.47% accuracy, 91.21% precision, 90.47% Recall and 90.29% F1-Score, applying preprocessing, class balancing and hyperparameter optimization techniquess

Keywords: Natural language processing, Machine learning, Consumer service, Requests classification, and Text classification.

TABLA DE CONTENIDO

LISTA DE ABREVIATURAS

CM: Matriz de Confusión

CR: Reporte de clasificación.

CV: Validación Cruzada

DP: Datos preprocesados

DPB: Datos preprocesados y con balanceo

DPBO: Datos preprocesados, con balanceo y con optimización de hiperparámetros

DT: Árbol de Decisión

EDA: Análisis exploratorio de datos

ET: Árboles Adicionales

IA: Inteligencia Artificial.

KNN: K vecinos más cercanos

LDA: Análisis discriminante lineal

LR: Regresión Logística

ML: Machine Learning.

NB: Naive Bayes

NLP: Procesamiento de lenguaje Natural

OD: Datos Originales

ODL: Datos Originales Limpios

RF: Bosque Aleatorio

ROC: curvas característica operativa del receptor

SVM: Máquina de vectores de soporte

TC: Clasificación de Texto

GLOSARIO

TF-IDF: Frecuencia de términos – Frecuencia inversa del documento».

Inteligencia Artificial (AI): la AI es un sistema computarizado que exhibe un comportamiento que comúnmente se considera que requiere inteligencia. En términos generales, es la ciencia y la ingeniería de hacer máquinas inteligentes, especialmente programas de computadora inteligentes. Está relacionado con la tarea similar de usar computadoras para comprender la inteligencia humana, pero la AI no tiene que limitarse a métodos que son biológicamente observables [1].

Machine Learning (ML): el ML o Aprendizaje de Máquina es una rama en evolución de los algoritmos computacionales que están diseñados para emular la inteligencia humana aprendiendo del entorno circundante. Las técnicas basadas en el aprendizaje automático se han aplicado con éxito en diversos campos que van desde el reconocimiento de patrones, la visión por computadora, la ingeniería de naves espaciales, las finanzas, el entretenimiento y la biología computacional hasta las aplicaciones biomédicas y médicas [2].

Procesamiento de lenguaje Natural (NLP): el NLP es una rama de la AI que ayuda a las computadoras a entender, interpretar y manipular el lenguaje humano, toma elementos prestados de muchas disciplinas, incluyendo la ciencia de la computación y la lingüística computacional, en su afán por cerrar la brecha entre la comunicación humana y el entendimiento de las computadoras [3].

Timework: herramienta organizacional de SIGMA Ingeniería S.A que permite conocer, gestionar y controlar las solicitudes de los clientes.

Geolúmina: consolida y sistematiza a empresas y concesiones de alumbrado, permitiendo dar cumplimiento al reglamento técnico de iluminación y alumbrado público en SIGMA Ingeniería S.A.

Geoaseo: optimiza el ejercicio de las empresas de aseo en operaciones como rutas, recolección, barrido, entre otras variables en las ciudades y municipios de un país en SIGMA Ingeniería S.A.

Geoambiental: herramienta tecnológica basada en información gerencial y geográfica que tiene como fin fortalecer los procesos y procedimientos de las organizaciones que gestionan el medio ambiente en SIGMA Ingeniería S.A.

Categoría: clase que resulta de una clasificación de personas o cosas según un criterio o jerarquía

Incidencias: requerimientos, solicitudes o eventos presentados en el área de servicio al cliente

Exactitud (*Accuracy*): es la relación entre el número de predicciones correctas y el número total de muestras de entrada [4][5].

Precisión: es una métrica que cuantifica el número de predicciones positivas correctas realizadas [4].

Sensibilidad (*Recall*): es una métrica que cuantifica el número de predicciones positivas correctas realizadas a partir de todas las predicciones positivas que podrían haberse realizado [4].

Valor-F1 (*F1-score*): el valor F1 es la media armónica entre precisión y sensibilidad [4][5].

Matriz de confusión (CM): resume el número de predicciones realizadas por un modelo para cada clase y las clases a las que realmente pertenecen esas predicciones. Ayuda a comprender los tipos de errores de predicción que comete un modelo.[5]

Informe de clasificación (CR): crea un informe de texto que muestra las principales métricas de clasificación, como precisión, Sensibilidad, Valor-F1 y soporte, este último es el número de ocurrencias de la clase dada en el conjunto de datos [6].

Curva ROC: constituye un método estadístico para determinar la exactitud diagnóstica de los datos etiquetados como *test*, siendo utilizadas con tres propósitos específicos: determinar el punto de corte de una escala continua en el que se alcanza la sensibilidad y especificidad más alta, evaluar la capacidad discriminativa que los datos de *test* tienen para diferenciar categorías y comparar la capacidad discriminativa de dos o más datos de *test* categorizados que expresan sus resultados como escalas continuas [7].

Balanceo al mayor o Sobremuestreo: el sobremuestreo se puede realizar aumentando la cantidad de instancias o muestras de clases minoritarias produciendo nuevas instancias o repitiendo algunas de ellas para igualar en cantidad a las clases con mayor cantidad de muestras [8].

Lematización: proceso que reduce la forma de una palabra y halla el lema correspondiente. El lema es la forma que por convenio se acepta como representante de todas las formas flexionadas de una misma palabra, de este modo se reduce el tamaño del conjunto de características inicial y unifica todas las palabras a su raíz o lema [9].

Stop Words: palabras sin significado que se eliminan como una tarea de preproceso [10].

Overffiting: es un problema fundamental en el aprendizaje automático supervisado que nos impide generalizar perfectamente los modelos para que se ajusten bien a los datos observados en los datos de entrenamiento, así como a los datos no vistos en el conjunto de pruebas [11].

LISTA DE TABLAS

LISTA DE FIGURAS

1 PRESENTACIÓN

La Inteligencia Artificial (AI) es la ciencia y la ingeniería de la creación de máquinas que presenten capacidades semejantes al ser humano, está relacionada con la tarea similar de usar computadoras para comprender la inteligencia humana [12]. Uno de los campos de énfasis de la AI es el *Machine Learning* (ML) o Aprendizaje de Máquina que hace referencia al aprendizaje automático de las máquinas; las herramientas que cuentan con esta característica desarrollan soluciones para evolucionar y aprender de los datos con los que trabajan, de este modo, se automatizan procesos sin intervención humana [13]. Algunas de las ventajas de aplicar ML en las empresas son: mejorar el servicio al cliente, analizar preferencias de los clientes, ofrecer productos personalizados de forma automática, mejorar la percepción de la empresa, potenciar la fidelización de clientes y automatizar procesos de rutinas o tareas mecánicas [14],[15].

El presente trabajo tiene como unidad de análisis la empresa SIGMA Ingeniería S.A, una empresa de Manizales enfocada en el desarrollo de Software de georreferenciación y Sistemas de Información Geográfica (SIG) que soportan la gestión pública en Colombia. El objetivo de este proyecto es construir un modelo computacional basado en aprendizaje de máquina que le permita al área de soporte técnico, la clasificación de incidencias, requerimientos o peticiones de los clientes, con el fin de obtener protocolos para la solución de problemas específicos con un tiempo de respuesta óptimo. Los campos de aplicación que abarca el desarrollo de este modelo computacional son procesamiento de lenguaje natural o NLP por sus siglas en inglés (Natural Language Processing) y técnicas de ML aplicados a la preparación, procesamiento y clasificación de texto.

El contenido de este documento inicia con una recopilación de antecedentes basado en estudios de AI aplicado a la clasificación de texto, al NLP y su aplicación en las empresas. Seguido, se encuentra el planteamiento del problema de investigación y su justificación donde se describe la problemática encontrada en la empresa SIGMA Ingeniería S.A que motivó el desarrollo de esta tesis; posteriormente, se presenta el referente conceptual, contextual y legal, seguido de los objetivos y la sección de metodología donde se muestran las diferentes técnicas aplicadas en la investigación y el proceso realizado. Por último, se

presentan los resultados obtenidos juntos con su discusión, recomendaciones, contribuciones de la tesis y referencias bibliográficas usadas

2 ANTECEDENTES

La tecnología no para de revolucionar las industrias y la aparición de tendencias como la AI seguirán transformando los negocios. La causa se reduce a un simple hecho: hasta ahora, la tecnología ha sido creada para ser usada por seres humanos, y con base a ese uso, se desarrollan relaciones, negocios, aplicaciones y soluciones para la sociedad [16].

Los inicios de la AI se remontan a la filosofía, la ficción y la imaginación nacida a mediados de los años 50, gracias al informático John McCarthy y a la ayuda de Marvin Minsky y Claude Shannon [17]. Este concepto, designado en la conferencia de Dartmouth, definía la palabra como la ciencia e ingeniería de “hacer máquinas inteligentes.”, sin embargo, no es hasta la década de los 90 que la AI empieza a adoptarse como lo conocemos hoy en día [18].

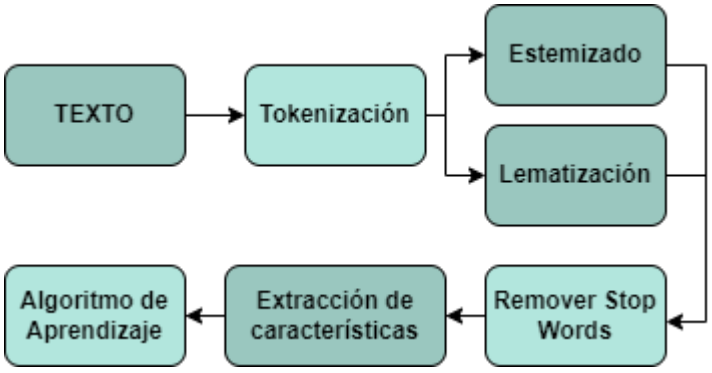
Una de las ramas de la AI es el NLP que se encarga de explorar cómo los computadores pueden ser usados para entender y manipular el lenguaje humano, las tareas en el NLP con frecuencia involucran el reconocimiento de voz, la comprensión del lenguaje natural y la generación del mismo [19].

Según Li Deng [20], el NLP es una gama de técnicas computacionales motivadas teóricamente para procesar y comprender textos con el fin de desarrollar aplicaciones prácticas novedosas para facilitar las interacciones entre computadoras y lenguajes humanos. Esta técnica tomó gran importancia porque existe un amplio almacenamiento de información registrada o almacenada en lenguaje natural que podría ser accesible a través de ordenadores. La información se genera constantemente en forma de libros, noticias, informes y artículos científicos. Un sistema que requiere una gran cantidad de información debe poder procesar el lenguaje natural para recuperar gran parte de la información disponible en computadoras [21]. El NLP es un campo en el que se tienen que desarrollar, evaluar o analizar teorías de representación y razonamiento. Todas las dificultades de la AI surgen en el ámbito de resolver "el problema del lenguaje natural" que es tan difícil como resolver "el problema de la AI" porque cualquier campo puede expresarse o representarse en lenguaje natural.

A. Masood Khan [22], expone la minería de texto y análisis de texto como términos similares y lo que hacen es relacionar estos dos términos y enfocarlos al NLP, ya que explican que la unión de estas dos técnicas puede permitir comprender la sintaxis (lo que dice la palabra) y la semántica (lo que significa la palabra) a partir de información y datos obtenidos por seres humanos para finalmente estar en la capacidad de llevar a cabo una clasificación como lo es para este caso la clasificación de textos (TC por sus siglas en inglés).

Cabe resaltar que algunas de las aplicaciones de AI están basadas en la clasificación automática de texto, la cual puede definirse como una técnica de ML que asigna una instancia determinada a etiquetas predefinidas en función de su contenido [23]. En general, la TC, juega un papel importante en la extracción y resumen de información, la recuperación de texto y la respuesta a preguntas, es así donde [24] toma gran importancia al recomendar diferentes etapas y procesos a llevar a cabo al trabajar con textos; la Figura 1 muestra los procesos recopilados para aplicar a conjuntos de datos textuales:

Figura 1. Proceso para clasificación de texto.



Adaptado de [24].

La primera sección hace referencia al debido pre-procesamiento a realizar al texto a trabajar, donde se hace importante por un lado aplicar el proceso de *Stemming* (Estemizado) o *Lemmatization* (Lematización) ya que ayuda a eliminar palabras mal escritas y a reducir el tamaño inicial de la palabra convirtiéndola a su raíz. Adicionalmente, se recomienda

realizar la limpieza de las palabras vacías más conocidas como *stop words*, ya que son palabras que no aportan significado por sí solas y reducen significativamente la dimensionalidad de los datos e información a procesar. Por otro lado, se realiza el proceso de extracción o transformación de características, que es una técnica de ponderación que utiliza un método estadístico para calcular la importancia de una palabra en todo el corpus de texto en función del número de veces que la palabra aparece [25], puede reducir el efecto de algunas palabras irrelevantes, mientras retiene palabras importantes que afectan todo el texto. Al aplicar estos procesos se puede producir una mejor precisión al aplicar algoritmos de ML.

El ML es una rama de la AI y se encarga de estudiar algoritmos y métodos estadísticos que utilizan los sistemas informáticos para realizar de forma eficaz una tarea específica sin utilizar instrucciones explícitas, basándose en patrones e inferencias [26]. Las técnicas de ML que se destacan para resolver tareas de clasificación son: máquinas de vectores de soporte (SVM por sus siglas en inglés), bosques aleatorios (RF por sus siglas en inglés), Regresión logística (LR por sus siglas en inglés), árboles de decisión (DT por sus siglas en inglés), análisis discriminante lineal (LDA por sus siglas en inglés), Naïve Bayes (NB por sus siglas en inglés) y K Vecinos más cercanos (KNN por sus siglas en inglés)[27], [28].

Para problemas de clasificación, específicamente de texto, A. I. Kadhim [29] analiza las diferentes técnicas de aprendizaje automático supervisado como son los clasificadores NB, SVM, KNN y analiza el efecto de cada técnica para clasificar documentos en una o más clases según su contenido. Los resultados muestran que, en términos de precisión, SVM es el mejor algoritmo para todas las pruebas que utilizan reseñas de películas en el conjunto de datos y tiende a ser más precisa que otros métodos, sin embargo, demuestra que las técnicas funcionan de manera diferente dependiendo del texto del conjunto de datos (corto y largo). Por lo que se obtuvo que KNN en general funciona bien en varios algoritmos de clasificación de texto, partiendo de esto, C. Sharpe et al. [30] explica que se deben tener en cuenta diferentes factores al implementar una tarea de clasificación; hay dos grandes categorías de clasificadores: generativos y discriminativos [31]. Los clasificadores generativos incluyen clasificadores bayesianos como Naive Bayes (NB) y los clasificadores

discriminativos incluyen las máquinas de soporte de vectores y redes neuronales donde en muchos de los casos los clasificadores discriminativos producen una mejor precisión de clasificación [30], y evidencian que las técnicas de clasificación más populares para problemas de optimización son SVM, RF y NB que están muy influenciados por la calidad de la fuente de datos y las técnicas de representación de características, ya que las características irrelevantes y redundantes de los datos degradan la precisión y el rendimiento del clasificador.

En este sentido, a partir de la revisión de literatura, se muestra que SVM ha sido reconocido como uno de los métodos de TC más efectivos [32]–[35]. Esta técnica tiene sólidos fundamentos teóricos y excelentes éxitos empíricos que se han aplicado a tareas como el reconocimiento de dígitos escritos a mano, el reconocimiento de objetos y como se nombró anteriormente, en la clasificación de texto [33], [36], sin embargo, SVM es más complejo, por lo tanto, exige mayores consumos de tiempo y memoria durante la etapa de entrenamiento y la etapa de clasificación [23].

Luego de tener en cuenta la clasificación de textos como se discutió en los estudios anteriores, es fundamental combinarla con la clasificación de textos multi-etiqueta ya que este trabajo contiene una cantidad importante de clases. Wang et al. [37] diseñó un algoritmo basado en razonamiento llamado Multi-Label Reasoner (ML-Reasoner) para la tarea de clasificación multi-label, en este estudio trabaja con técnicas como *Convolutional neural network* (CNNs), *Recurrent Neural Network* (RNN), una versión más avanzada de RNN como *Long short-term memory* (LSTM*), Bidirectional Encoder Representations from Transformers* (BERT), *Binary relevance* (BR), *Classifier chains* (CC), *Label Powerset* (LP) y otros algoritmos que han sido adaptados de SVM. [37] muestra los mejores resultados usando la red LSTM con 77.60% de precisión con 54 categorías diferentes en un conjunto de datos.

En este punto, vale la pena hablar de los estudios realizados en esta rama enfocados al área de soporte y servicio al cliente en las empresas, es por este motivo que se tiene en cuenta a Y. Xu et al. [38], donde explican que la AI en el contexto del servicio al cliente se encarga

de brindar recomendaciones, alternativas y soluciones personalizadas a los mismos. Este trabajo expone un experimento que muestra si los consumidores de un banco prefieren aplicaciones de servicio al cliente en la línea de AI o directamente con humanos. Los resultados muestran que, en el caso de tareas de baja complejidad, los consumidores consideraban que la capacidad de resolución de problemas en la línea de AI era mayor que la del servicio al cliente humano, sin embargo, para tareas de alta complejidad, consideraban que el servicio al cliente humano era superior y era más probable que lo usaran que la línea de IA. De modo que, para aplicar AI en las empresas, el modelo desarrollado debe estar muy bien entrenado y ser capaz de resolver problemas a los clientes tanto de alta como de baja complejidad. Por este motivo, se hace necesario enfocarse en procesos de clasificación y categorización de textos adecuados y acordes a los datos relacionados con los clientes.

Por otro lado, M. Raza et al. [39] se centran en la realización de un análisis de las opiniones de clientes relacionadas con los productos Software como Servicio (SaaS). Los autores usaron once enfoques tradicionales de clasificación de aprendizaje automático para lograr esta tarea y probar el rendimiento de cada uno de ellos. Este estudio es importante ya que no solo determinaron los mejores parámetros para cada algoritmo de aprendizaje, sino que también usaron un conjunto de datos desequilibrado en términos de distribución de la muestra por clase, lo cual es de gran utilidad para tener como referencia en los datos a procesar en este trabajo ya que cuentan con las mismas características.

Finalmente, se puede hablar de otro tipo de búsquedas de textos como lo expone S. A. Pérez at al. [40] en su trabajo, que tiene como objetivo la implementación de un sistema de clasificación automática de textos de opiniones realizadas por clientes de cierta compañía. Lo que se hizo en este desarrollo fue clasificar las opiniones según categorías establecidas (Pregunta, Sugerencia, Información Usuarios, Otras e Información Empresa), y aplicaron diferentes técnicas de ML en las que incluyeron DT, SVM, KNN y algunas variantes del modelo de NB. A partir de la aplicación de las diferentes técnicas nombradas, para este caso se evidencia que obtuvieron los mejores resultados y mayor porcentaje de eficiencia

con el modelo de DT y nuevamente con el método de SVM mediante la representación binaria.

Esta revisión de antecedentes ha permitido entender las técnicas supervisadas de ML más comunes enfocadas en la TC, además de encontrar las técnicas que tienden a brindar mayores desempeños para las tareas relacionadas con este problema de clasificación. Se concluye que es de gran importancia realizar procesos de NLP, reducción de características y transformación de características para que los resultados de desempeños obtenidos puedan crecer de manera significativa [41]. Además, se ha podido evidenciar que existen pocos estudios que relacionen técnicas de analítica de datos enfocado en las empresas y clasificación multi-etiqueta que promuevan la mejora continua a partir de soluciones de ML e AI para brindar soluciones de una manera más rápida y eficiente al cliente; uno de los objetivos de mayor importancia del presente trabajo.

3 ÁREA PROBLEMÁTICA Y PREGUNTA DE INVESTIGACIÓN

3.1 DESCRIPCIÓN DEL ÁREA PROBLEMÁTICA

El desarrollo de la tecnología junto con el gran crecimiento que ha tenido el internet impactó profundamente el desarrollo de diferentes ámbitos en el mundo en que vivimos, estos dos fenómenos trajeron consigo un cambio trascendental donde cada individuo puede recibir y enviar información a través del internet desde cualquier lugar y en cualquier momento, incluso almacenar y manejar esta información. La recopilación de datos ha permitido un gran avance en la tecnología, específicamente en el campo de la AI. En Colombia se han adoptado marcos normativos y éticos vinculantes que respondan de manera directa a la implementación de la AI en el sector público, sin embargo, se debe considerar cómo deben incorporarse estos marcos éticos en el proceso de toma de decisiones en el sector público y el privado [42].

La AI no solo ha permitido posicionar las empresas en mercados internacionales, sino que también ha servido de herramienta para el uso de la tecnología en las empresas. Ante la realidad planteada surge el campo de la ciencia de datos, ya que una de las ramas de la AI se caracteriza por la utilización de herramientas propias del NLP como es el caso del análisis de texto y los sentimientos en las opiniones de los usuarios [43].

En la actualidad, las grandes bases de datos se han convertido en uno de los recursos productivo más importante para las organizaciones; su gestión y procesamiento a través de la ciencia de datos permite la detección de tendencias invisibles o complicadas de detectar por un analista. Esto se traduce en un aumento de la productividad al permitir identificar patrones ineficientes y aplicar las soluciones correspondientes desde el análisis de datos ara mejorar toda la cadena productiva [44]. Si una base de datos se gestiona adecuadamente, la organización obtendrá diferentes ventajas: aumentará su eficacia, realizará los trabajos con mayor rapidez y agilidad, simplificará los procesos, mejorará la protección y seguridad de los datos que se almacenan y maximizará los tiempos y la productividad con efectos en la competitividad a un mayor nivel de la compañía [45].

Para manejar una base de datos textuales apropiadamente mediante AI se requieren de técnicas de NLP para realizar una limpieza adecuada de los datos trabajados y encontrar similitudes e importancias de palabras entre párrafos o cantidades significativas de textos, allí es donde radica el objetivo de vincular el NLP con la AI a partir de técnicas de ML para que sirvan de herramientas conjuntas tanto en el procesamiento de conjuntos de datos textuales como en la clasificación de múltiples categorías. Partiendo de este análisis y pruebas posteriores de funcionamiento, estos resultados de investigación se aplicarán localmente a una empresa colombiana.

En este sentido, es la empresa SIGMA Ingeniería S.A donde se aplicaron los conocimientos aprendidos a partir de esta investigación. Esta empresa tiene como punto de partida de su labor, el manejo de datos dentro de los sectores que se abarcan en sus líneas de negocio conocidas como Geolumina, Geoaseo y Geoambiental. Los requerimientos, incidentes y peticiones presentados por parte del cliente son atendidos mediante una herramienta organizacional denominada Timework, que permite conocer, gestionar y controlar las solicitudes de los clientes. Esta herramienta permite registrar los tiempos de las actividades ejecutadas, priorizaciones de los equipos de trabajo, gestión de compromisos, gestión de riesgos y planeación estratégica; adicionalmente, genera datos diarios de todos los equipos de trabajo, con el fin de medir la operación para mejorar la productividad, la eficiencia y desarrollar procesos de mejora continua.

En su tiempo de operación de 4 años, la plataforma cuenta con aproximadamente 15.000 datos o tickets de servicio generados por la organización y los equipos de trabajo de soporte técnico. Estos tickets de servicio describen todas las actividades y procedimientos llevados a cabo para la solución de los requerimientos presentados y resultan fundamentales para darle orden a la gestión de procesos internos de la empresa, encontrar patrones y predecir comportamientos.

Sin embargo, SIGMA Ingeniería S.A ha identificado por medio de esta herramienta una insatisfacción por parte del cliente debido al incumplimiento del tiempo de respuesta estimado para la solución de requerimientos y solicitudes. Esto se debe a que la empresa no cuenta con bases de datos que almacenen los protocolos establecidos para la solución de los

requerimientos presentados por parte del cliente. Por este motivo, se hace necesario contar con un modelo computacional aplicado al ML que permita clasificar todos los requerimientos y peticiones en diferentes categorías. Estas categorías tendrán relación directa con las diferentes solicitudes que puede presentar el cliente y cada categoría brindará los protocolos a seguir para darle una solución satisfactoria en tiempo y en calidad al cliente por parte del área de soporte técnico de la empresa [46].

Este modelo computacional le permitirá al área de soporte técnico encontrar de manera eficiente y rápida la solución que le debe brindar al cliente sin necesidad de escalamientos y reprocesos; que son la causa principal del incumpliendo en los tiempos de solución pactados hacia el cliente y, por consiguiente, la razón directa de su insatisfacción.

3.2 HIPÓTESIS DE LA INVESTIGACIÓN

En base a la problemática expuesta surge la siguiente hipótesis:

> El uso de arquitecturas computacionales basadas en técnicas de ML y NLP será beneficioso para la clasificación de requerimientos de clientes en múltiples categorías para la futura automatización de procesos y protocolos de solución brindados por el área de servicio y soporte técnico en las empresas.

3.3 FORMULACIÓN DEL PROBLEMA

Del planteamiento anterior surge la pregunta:

- ¿Cómo automatizar la clasificación de múltiples categorías y protocolos de solución del área de servicio y soporte técnico en empresas utilizando técnicas de ML y NLP?

4 JUSTIFICACIÓN

La AI es uno de los componentes reconocidos por su potencial para transformar de manera

radical la forma como hoy vivimos, pues hace posible que las máquinas aprendan de la experiencia, se ajusten a nuevas aportaciones y realicen tareas como seres humanos [1]. Por lo tanto, empleando estas tecnologías, las computadoras pueden ser entrenadas para realizar tareas específicas y procesar grandes cantidades de información mediante el reconocimiento de patrones en los datos.

La IA, tiene un gran potencial para acelerar el progreso de los Objetivos de Desarrollo Sostenible (ODS) [47]. Un equipo de Científicos del Instituto Andaluz de Investigación en *Data Science and Computational Intelligence* (DaSCI) de la Universidad de Granada (UGR), junto a la compañía Ferrovial y la Real Academia de la Ingeniería (RAI), han realizado un estudio que evidencia cómo la ingeniería y la implantación de soluciones tecnológicas fuertemente ancladas en la AI favorece el progreso de los 17 ODS [48]. Esto, debido a que la AI está optimizando métodos de razonamiento y ejecución más precisos, seguros y eficientes, en diversas tareas y ámbitos, por lo que diferentes organismos internacionales, entre ellos la Organización para la Cooperación y el Desarrollo Económico (OCDE) y la Organización de las Naciones Unidas (ONU) han destacado el papel relevante que estas tecnologías pueden tener para su cumplimiento [49].

El valor social que los datos han creado en los últimos años supone un nuevo uso de la información que se genera mediante servicios tecnológicos, uno de estos servicios es la minería de datos; que es el proceso de hallar anomalías, patrones y correlaciones en grandes conjuntos de datos para predecir resultados [50]. Los beneficios que se obtienen de la minería de datos a partir de la TC y el ML tienen gran importancia en el entorno empresarial altamente competitivo de hoy en día. Todo ello, es debido a que la minería de datos descubre información útil que contribuye a la toma de decisiones tácticas y estratégicas para detectar los datos claves que permite encontrar, atraer y retener a los clientes y reducir el riesgo de perderlos. Así mismo, las empresas pueden mejorar la atención al cliente a partir de la información obtenida, abre nuevas oportunidades de negocios y además, ahorra costes a las empresas [51].

El ML enfocado a las empresas, es un campo de investigación y aplicación prometedor dado que cada vez las organizaciones están más interesadas en aprovechar las grandes cantidades de datos e información del que disponen. Las técnicas de recuperación de la información y de aprendizaje automático facilitan la tarea de extracción de conocimiento, por lo que no solo permite hacer un análisis profundo de todos los datos, sino que a través de los algoritmos desarrollados dota a los ordenadores de la capacidad de identificar patrones que brinden información y conocimiento relevante dentro de las empresas [46].

Esta investigación se centra en organizaciones que quieran valorizar sus datos y tomar decisiones más acertadas, que les permita ser eficientes en los diferentes procesos internos y externos debido a los incrementos en la capacidad, agilidad del procesamiento y análisis de la información. A partir de este desarrollo, se pretende generar la capacidad en las empresas de responder a condiciones cambiantes de la manera más rápida y óptima posible, lo cual es clave en el desarrollo empresarial, mantenimiento y crecimiento de los negocios.

El factor de novedad de esta investigación está enfocada al ámbito empresarial, ya que propone implementar un modelo computacional de clasificación en las empresas donde el ML pueda descubrir ideas y patrones ocultos en los datos de la entidad. Esto con el fin de incrementar la satisfacción del cliente a través de la revisión de registros históricos para analizar su comportamiento y brindar correctamente la solución esperada a las incidencias presentadas. Esta práctica reduce el costo y la cantidad de tiempo invertido en la gestión relacional con el consumidor potencial.

Este trabajo le pretende dar solución a la dificultad a la que se enfrenta SIGMA Ingeniería S.A, donde por medio de diferentes técnicas de analítica de datos y la correcta implementación del NLP permita al área de soporte técnico interactuar con un modelo computacional que le brinde por un lado la categoría a la que pertenece dicho requerimiento; las cuales están divididas en 41 categorías, por otra parte, que le brinde automáticamente los diferentes protocolos de solución a estos requerimientos con un alto grado de confiabilidad y en tiempos cortos, lo que conlleva a una reducción significativa en el tiempo tomado por el área de soporte técnico en brindar una respuesta y solución al

cliente.

El uso y aplicación del ML e AI se ha convertido en una herramienta de gran utilidad para los negocios al poder mejorar las operaciones comerciales y el análisis de variables importantes para la organización. Es así como las empresas pueden obtener múltiples beneficios al acceder al ML y convertirlo en un aliado estratégico de los procesos al poder resolver problemas complejos y predecir comportamientos del mercado, de los clientes y de diferentes grupos de interés [46].

5 REFERENTE TEÓRICO

5.1 REFERENTE CONCEPTUAL

En esta sección se explican los conceptos más relevantes a tener en cuenta para el desarrollo del trabajo, los términos se presentan en orden lógico de tal manera que se pueda brindar una mejor comprensión entre los conceptos brindados y, un mejor entendimiento del proceso a realizar en la sección de metodología (ver Sección 7.)

Nivel TRL:

Los Niveles de Madurez de la Tecnología o Technology Readiness Levels (TRLs) han empezado a usarse en las convocatorias como una guía para definir las fases de un proyecto de Investigación y Desarrollo para la creación de innovaciones.

El nivel TRL es una medida para describir la madurez de una tecnología. Este concepto surge en la NASA, pero se ha generalizado para ser utilizado en el desarrollo de proyectos de cualquier tipo de industria y no sólo para proyectos espaciales. En concreto, un TRL es una forma aceptada de medir el grado de madurez de una tecnología [52]. La Figura 2 muestra los nueve niveles tecnológicos.

Figura 2. Infografía de la escala de los niveles tecnológicos.

NIVELES DE MADUREZ DE LA TECNOLOGÍA TRLs			
TRL 1	ENTORNO DE LABORATORIO	INVESTIGACIÓN	PRUEBA DE CONCEPTO/ INVESTIGACIÓN INDUSTRIAL
TRL 2			
TRL 3			
TRL 4	ENTORNO DE SIMULACIÓN	DESARROLLO	PROTOTIPO DEMOSTRADOR/ DESARROLLO TECNOLÓGICO
TRL 5			
TRL 6			
TRL 7	ENTORNO REAL	INNOVACIÓN	PRODUCTO COMERCIALIZABLE/ CERTIFICACIONES
TRL 8			
TRL 9			

Adaptado de [53].

Específicamente es al TRL 4 al que pertenece el presente trabajo, el TRL 4 o validación de sistema en un entorno de laboratorio [54], es el primer paso para determinar si los componentes individuales funcionarán juntos como un sistema. El sistema de laboratorio será una combinación de pruebas piloto capaces de ejecutar todas las funciones planteadas dentro del alcance del proyecto para la empresa SIGMA Ingeniería S.A y se espera que el modelo presente pruebas superadas de factibilidad en condiciones de operación y funcionamiento simuladas.

Técnicas de ML:

En 1956, en una conferencia en la Universidad de Dartmouth, se propuso formalmente el término "inteligencia artificial". Ese momento fue el primer paso en un nuevo tema de estudio de cómo las máquinas aprenden de la experiencia y simulan actividades de inteligencia humana [1]. El desarrollo de la AI ha aportado enormes beneficios económicos y sociales a la humanidad, muchos académicos iniciaron investigaciones relacionadas con AI desde finales del siglo XX donde utilizan computadoras para simular comportamientos humanos inteligentes y entrenan a las computadoras para que aprendan también comportamientos humanos como el análisis y la toma de decisiones [55]. Una de las ramas de la AI basada en la idea de que los sistemas pueden aprender de datos, identificar patrones y tomar decisiones con mínima intervención humana es llamado ML [1]; consiste básicamente en automatizar, mediante distintos algoritmos, la identificación de patrones o tendencias que se “esconden” en los datos, estos distintos algoritmos permiten resolver problemas tanto de clasificación como de agrupamiento y de regresión [56]. Los tipos de implementación de ML pueden clasificarse en diferentes categorías uno de ellos son:

Aprendizaje Supervisado:

En el aprendizaje supervisado, los algoritmos trabajan con datos “etiquetados”, intentando encontrar una función que, dadas las variables de entrada (input data), les asigne la etiqueta de salida adecuada. El algoritmo se entrena con un “histórico” de datos y así “aprende” a asignar la etiqueta de salida adecuada a un nuevo valor, es decir, predice el valor de salida [57].

Las técnicas de ML que destacan para resolver tareas de clasificación son: SVM, Árboles Adicionales (ET), RF, LR, LDA, NB y KNN [33], [58], [59].

LR. La regresión logística es un proceso de modelado de la probabilidad de que exista una determinada clase o evento. La LR utiliza la función logística para modelar una variable dependiente binaria. Esto se puede aplicar al uso de este modelo con clases o eventos más complejas que involucran más de dos categorías [60][5].

La función logística, representa una curva en forma de "S" que toma cualquier valor numérico real y lo mapea entre valores de 0 y 1 [39] . LR se basa en suposiciones sobre la relación entre las variables dependientes e independientes. La distribución condicional usada en este modelo es una distribución de Bernoulli, ya que la variable dependiente tiene la forma de una variable binaria (presencia o ausencia de deslizamientos) [61]. En el análisis de regresión logística, la relación entre el evento y su dependencia se puede expresar mediante la Ecuación 1.

$$y = \frac{1}{1+e^{-z}} \qquad (1)$$

Donde "z" es cualquier valor numérico que se pretende transformar y mapear entre 0 y 1 y "y" es la probabilidad de ocurrencia de un evento. La LR implica ajustar una ecuación de la siguiente forma (ver Ecuación 2) a los datos.

$$z = b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_n x_n \qquad (2)$$

donde b_0 es la intersección del modelo, b_i(i = 0, 1, 2, ..., n) son los coeficientes de pendiente del modelo de LR y x_i (i = 0, 1, 2, ..., n) son las variables independientes. El modelo lineal formado es finalmente una regresión logística de presencia o ausencia de las condiciones presentes (variables dependientes) sobre las condiciones previas a la falla (variables independientes) [61][49] .

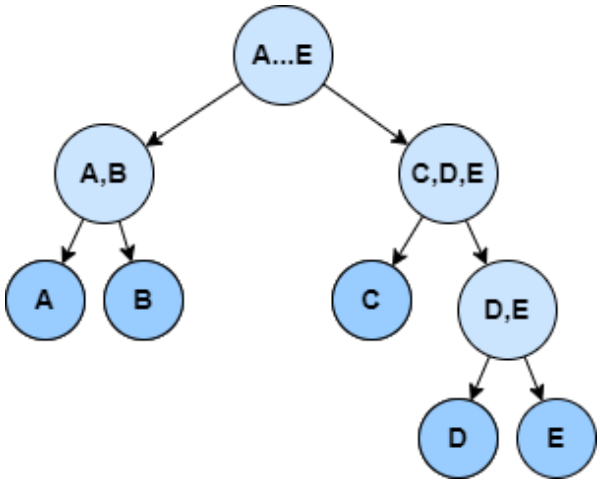
LDA. El análisis discriminante lineal es un método utilizado para encontrar una combinación lineal de características que separa dos o más clases [62]. La técnica LDA se usa comúnmente para la clasificación de datos y la reducción de dimensionalidad [63][5][64]. Este enfoque trata de encontrar la dirección de proyección en la que datos

pertenecientes a diferentes clases se separan al máximo. Matemáticamente, trata de encontrar la matriz de proyección o los pesos de los eventos de tal manera que la relación entre la matriz de dispersión entre clases y la matriz de dispersión dentro de las clases proyectadas se maximice. A diferencia de los algoritmos basados en PCA, la técnica LDA considera la pertenencia a clases para la reducción de dimensiones [65].

DT. Modelo de clasificación mediante la construcción de un árbol de decisión, está en la capacidad de realizar clasificaciones multiclase [66]. Este algoritmo predice el valor de una variable objetivo mediante el aprendizaje sucesivo de reglas de decisión simples que se deducen de las bases de datos. Esta estrategia de clasificación se puede describir mediante la Figura 3. Para empezar, el árbol se codifica como una cadena de símbolos de manera que existe una relación única entre la cadena y el árbol de decisión. La cadena se decodifica en la computadora y se configuran punteros para definir la ruta de clasificación adecuada para cada muestra de datos.

En general, un árbol de decisión consta de un nodo raíz, varios nodos interiores y varios nodos terminales u hojas. El nodo raíz y los nodos interiores están vinculados a etapas de decisión y los nodos terminales u hojas representan las clasificaciones finales (ver la Figura 3). En el nodo raíz se encuentra todo el conjunto completo de clases en las que se puede clasificar una muestra. Cada nodo interior consta de un conjunto de clases a clasificar el conjunto de características a utilizar y la regla de decisión para realizar la clasificación y finalmente las hojas cuentan con una clase única que suman la totalidad de las clases a clasificar [5], [67].

Figura 3. Estructura básica del DT.



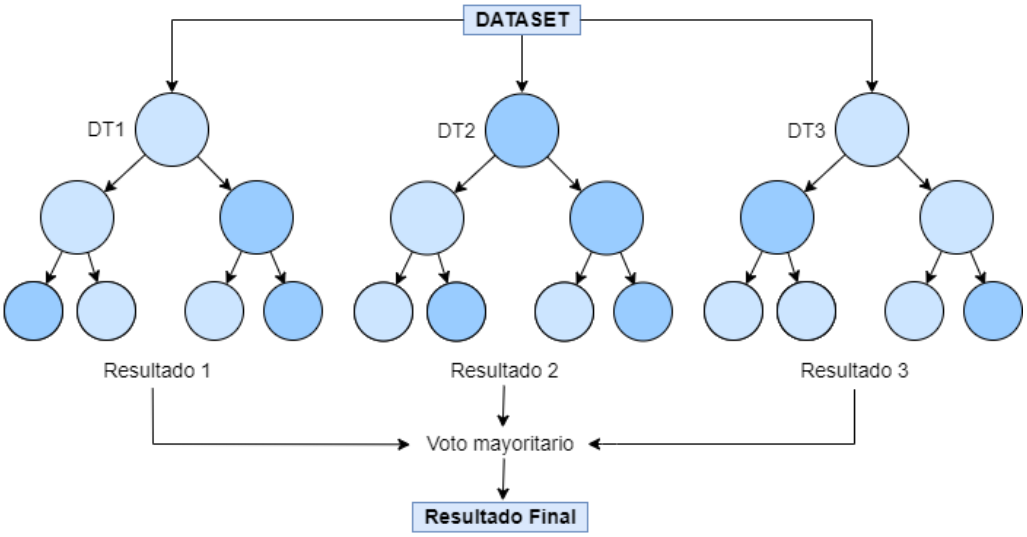
Adaptada de [67] [68].

RF. Este modelo funciona mediante la construcción de una multitud de árboles de decisión, es un método tanto para clasificación como para regresión [69]. RF crea múltiples árboles de decisión y los fusiona para generar una predicción más precisa y estable, la clase resultante es el resultado seleccionado por la mayoría de los árboles de decisión [70] (ver Figura 4).

En general, el uso de un clasificador de RF para la clasificación incluye cuatro pasos:

1. Seleccionar muestras aleatorias de un conjunto de datos dado.
2. Construir un árbol de decisión para cada muestra.
3. Obtener un resultado de predicción de cada DT.
4. Seleccionar la predicción con más votos o con más coincidencias entre todos los resultados de los DT como la clasificación final [30].

Figura 4. Estructura básica de un RF.



Adaptada de [68].

ET. Este modelo funciona mediante la construcción de una multitud de árboles de decisión como la técnica RF, ET también es un método para clasificación y regresión [69]. La diferencia es que el modelo ET agrega más aleatoriedad al proceso de entrenamiento del modelo mediante el uso de umbrales de decisión aleatorios para cada característica [71] [5]. La mejor división en un nodo interno la encuentra seleccionando al azar un solo umbral de decisión para cada característica y a partir de sus divisiones aleatorias, selecciona la que conduce al mayor aumento en la puntuación utilizada. Un mayor grado de aleatoriedad durante el entrenamiento produce árboles más independientes y, por lo tanto, disminuye aún más la varianza [72].

Sus dos diferencias principales con otros métodos de conjuntos basados en árboles son que divide los nodos internos eligiendo puntos de corte completamente al azar y que utiliza toda la muestra de aprendizaje para hacer crecer todos los árboles.

Los resultados de las predicciones de todos los árboles se acumulan para producir la predicción final a partir del campo más votado para problemas de clasificación y el promedio aritmético de los resultados de cada árbol en problemas de regresión [71].

KNN. Es un método de clasificación no paramétrico que se utiliza también para

clasificación y regresión. El modelo clasifica las muestras haciendo uso de los datos vecinos. Al evaluar una muestra, el modelo puede asignar pesos a las contribuciones de los vecinos. La elección óptima del valor de *k* depende en gran medida de los datos, este valor *k* representa la cantidad de instancias más similares y cercanos a la muestra evaluada: una *k* más grande suprime los efectos de ruido, pero hace que los límites de clasificación sean menos claros [73] [5]. El algoritmo KNN memoriza y representa todo el conjunto de datos. Las predicciones se realizan calculando la similitud entre una muestra de entrada y cada instancia de entrenamiento [74].

Para clasificar una muestra nueva y además desconocida, KNN calcula la distancia entre la nueva instancia y otras instancias en el espacio de funciones [39], estas distancias se pueden calcular usando diferentes métricas como:

Distancia Euclidiana: la distancia euclidiana entre un punto "*a"* y "*b"* se calcula como la raíz cuadrada de la suma de sus diferencias al cuadrado en todos los atributos de entrada *i* (ver Ecuación 3).

$$Distancia\ Euclidiana\ (a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \qquad (3)$$

Distancia de Manhattan: la distancia de Manhattan calcula la distancia entre dos puntos o vectores utilizando la suma de su diferencia absoluta. Esta distancia también es conocida como geometría del taxista (ver Ecuación 4).

$$Distancia\ Manhattan\ (a,b) = \sum_{i=1}^{n}|a_i - b_i| \qquad (4)$$

Distancia de Minkowski: La distancia de Minkowski es la generalización de la distancia euclidiana y de Manhattan como se indica en la Ecuación 5.

$$Minkowsky\ Euclidiana\ (a,b) = (\sum_{i=1}^{n}|a_i - b_i|^p)^{\frac{1}{p}} \qquad (5)$$

El valor de *p* en la distancia de Minkowski puede tomar valores de 1 o 2; si toma un valor de 1 es igual a la distancia de Manhattan y si toma el valor de 2 a la distancia euclidiana [39].

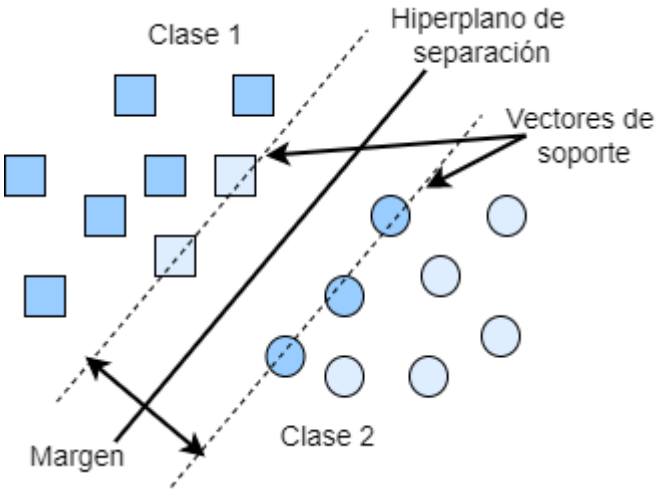
El número *k* de vecinos se selecciona se selecciona con el que tiene la distancia más baja de la nueva instancia. Este método también se aplica a conjuntos de datos con 2 o más número de clases.

SVM. La máquina de soporte de vectores resuelve tanto clasificación lineal y no lineal como análisis de regresión. Este modelo funciona mejor para conjuntos de datos pequeños y medianos. La idea fundamental de la clasificación con SVM es separar clases manteniendo el límite de decisión lo más lejos posible de las muestras de entrenamiento más cercanas. [69], [75]. En la Figura 5 se muestra un caso de dos clases con un hiperplano de separación óptimo y un margen máximo. Si las dos clases son linealmente separables, el hiperplano óptimo que separa los datos se puede expresar como la Ecuación 6.

$$g(x) = w^T * x + b = 0 \qquad (6)$$

Donde *w* es un vector normal que es perpendicular al hiperplano y *b* es el desplazamiento del hiperplano. SVM optimiza *w* y *b* para maximizar la distancia entre los hiperplanos paralelos mientras sigue separando los datos.

Figura 5. Concepto de una máquina de vectores de soporte de dos clases.



Adaptada de [30]

El SVM se implementa utilizando diferentes núcleos. El kernel más utilizado para implementar SVM se llama kernel SVM lineal. Además del núcleo lineal SVM, otros núcleos como el núcleo polinomial SVM y el núcleo de función de base radial (RBF), se pueden utilizar en escenarios más complejos de TC. En este trabajo se utiliza el núcleo RBF, al mostrar un mejor rendimiento en los datos que no se pueden separar linealmente en el espacio [30]. Adicionalmente, el SVM no se limita a problemas de clasificación de dos clases, sino que también se puede utilizar para la clasificación de problemas multiclase siendo reconocido como uno de los métodos de TC más eficaces [76], [77], [5].

NB. Este clasificador probabilístico utiliza el teorema de Bayes [78]. El modelo individualiza cada característica de la muestra de la que ninguna muestra será dependiente (suponiendo que la probabilidad de las características sea gaussiana). En otras palabras, cada pieza puede presentar una probabilidad de pertenecer a un grupo específico sin la presencia de las diferentes muestras [5].

De la notación general del teorema de Bayes, la probabilidad de que una muestra de texto *x* pertenezca una clase *c* es mostrada en la Ecuación 7.

$$P(c|x) = \frac{P(c)P(x|c)}{P(x)} \qquad (7)$$

Donde;

$P(c|x)$: es la probabilidad de que una muestra *x* pertenezca a una clase *c*

$P(c)$: es la probabilidad de que la clase *c* sea verdadera.

$P(x|c)$: es la probabilidad de observar que la muestra *x* pertenece a la clase *c* (verosimilitud).

$P(x)$: es la probabilidad de la muestra *x* independientemente de cualquier clase.

Para la clasificación de texto multiclase, el clasificador probabilístico creado con el enfoque ingenuo de Bayes da como resultado la clasificación del texto en función de la presencia de palabras en cada muestra de texto y la asignación a diferentes clases [79],[39]. En otras palabras, Naïve Bayes permite estimar la probabilidad de un suceso a partir de la probabilidad de que ocurra otro suceso, del cual depende el primero [40].

Procesamiento de lenguaje natural (NLP):

El NLP es una rama de la AI que ayuda a las computadoras a entender, interpretar y manipular el lenguaje humano, toma elementos prestados de muchas disciplinas, incluyendo la ciencia de la computación y la lingüística computacional, en su afán por cerrar la brecha entre la comunicación humana y el entendimiento de las computadoras [3]. El NLP es el enfoque computarizado para analizar texto que se basa tanto en un conjunto de teorías como en un conjunto de tecnologías, por una parte A. Chopra et al. [80] ofrece la siguiente definición:
"El NLP es una gama de técnicas computacionales motivadas teóricamente para analizar y representar textos que ocurren naturalmente en uno o más niveles de análisis lingüístico con el fin de lograr un procesamiento del lenguaje similar al humano para una variedad de tareas o aplicaciones". Por otro lado, S. Jusoh et al. [81] explica que el NLP es un subcampo de la AI y lingüística, dedicado a hacer que las computadoras comprendan las declaraciones o palabras escritas en lenguajes humanos.

Con respecto al área de investigación el NLP es un área de aplicación que explora cómo se pueden utilizar las computadoras para comprender y manipular el texto en lenguaje natural. Los investigadores en esta área tienen como objetivo recopilar conocimientos sobre cómo los seres humanos comprenden y usan el lenguaje de modo que se puedan desarrollar herramientas y técnicas de adaptación para que los sistemas informáticos comprendan y manipulen los lenguajes naturales para realizar tareas determinadas [82].
Los fundamentos de esta técnica se encuentran en varias disciplinas como informática y ciencias de la información, lingüística, matemáticas, ingeniería eléctrica y electrónica, inteligencia artificial y robótica, psicología, etc. Las aplicaciones incluyen una serie de campos de estudio, como traducción automática, procesamiento y resumen de textos en lenguaje natural, interfaces de usuario, multilingües y recuperación de información en varios idiomas, reconocimiento de voz, entre otros. [82].

Técnicas de NLP para pre-procesamiento de texto:

Eliminación de Stop Words:

El análisis de texto requiere varios pasos de pre-procesamiento para estructurarlo y extraer características [83], para empezar, se debe tener en cuenta que no todas las palabras en un documento pueden usarse para entrenar a un algoritmo de clasificación [10]. Hay palabras irrelevantes como verbos auxiliares, preposiciones, conjunciones y artículos que no brindan significados por sí solos. Estas palabras se denominan *stop words*, algunos ejemplos de estas palabras son: la, los, arriba, con, cuándo, además, qué, entre otros. Existen listas de dichas palabras [84] que se eliminan como una tarea de preproceso. El motivo por el que se deben eliminar las palabras vacías de un texto es que hacen que el texto parezca más pesado y menos importante para los analistas. La eliminación de las palabras vacías reduce significativamente la dimensionalidad de los datos e información a procesar.

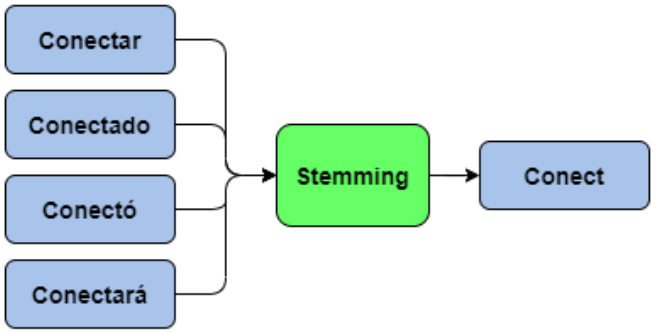
Estemizado:

El estemizado es una técnica de normalización sencilla, que a menudo se implementa como

una serie de reglas que se aplican progresivamente a una palabra para producir una forma normalizada [9]. Este método se utiliza para identificar la raíz de una palabra. Por ejemplo, las palabras conectar, conectado, conectó, conectará, todas pueden derivarse de la raíz "conect". El propósito de este método es eliminar varios sufijos, para reducir el número de palabras, para ahorrar tiempo en el procesamiento y reducir espacio en la memoria. Esto se ilustra en la Figura 6.

Algo importante que se debe tener en cuenta acerca del estemizado es que no se requiere que la palabra normalizada sea válida, sino solo que las variaciones de la misma palabra se asignen a la misma raíz, aunque pueda parecer contradictorio, esto no representa un problema. El estemizado se utiliza principalmente para indexar documentos en un motor de búsqueda, por lo que estas raíces, que pueden ser palabras no válidas, solo se procesan internamente para buscar documentos y nunca se muestran al usuario, por lo que, para el caso de este estudio, al tener relación directa con los usuarios es imprescindible conocer también la técnica *Lemmatization* (Lematización).

Figura 6. Proceso de Estemizado



Elaboración propia

Lematización:

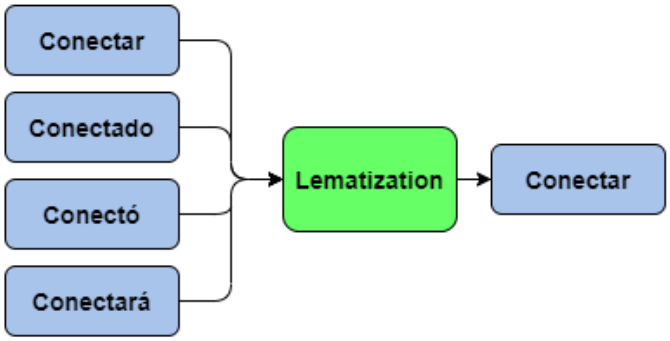
Se puede pensar en la lematización como una versión más sofisticada del estemizado ya que no solo reduce la forma de la palabra, sino que además la reduce a su forma base

adecuada, donde dada una forma flexionada de la palabra, ya sea en plural, en femenino, conjugada, entre otros (ver Figura 7), halla el lema correspondiente. El lema es la forma que por convenio se acepta como representante de todas las formas flexionadas de una misma palabra, de este modo se reduce el tamaño del conjunto de características inicial y unifica todas las palabras a su raíz o lema [85].

Para hacer esto, los algoritmos de lematización dependen de la disponibilidad de información de la parte del discurso en las palabras de entrada porque es posible que sea necesario aplicar diferentes reglas de normalización, ya sea que la palabra sea un sustantivo, verbo, adjetivo u otro.

En resumen, la lematización es casi siempre una mejor opción desde un punto de vista cualitativo. Con los recursos computacionales actuales, la ejecución de algoritmos de lematización no debería tener un impacto significativo en el rendimiento general. Sin embargo, si estamos optimizando la velocidad, un algoritmo de estemizado puede ser una posibilidad.

Figura 7. Proceso de Lematización



Elaboración propia

Extracción de características con TF-IDF:

Los datos de texto requieren una preparación especial antes de que se puedan comenzar a usar para el modelado predictivo. Las palabras del texto deben codificarse como números

enteros o valores de punto flotante para usar como entrada de un algoritmo de aprendizaje automático, llamado extracción de características o vectorización.
La extracción de características TF-IDF es una técnica de ponderación comúnmente utilizada en el procesamiento de información y la minería de datos. Esta técnica utiliza un método estadístico para calcular la importancia de una palabra en todo el corpus en función del número de veces que la palabra aparece en el texto y la frecuencia de los documentos que aparecen en todo el corpus [86]. Su ventaja es que puede filtrar algunas palabras comunes pero irrelevantes, mientras retiene palabras importantes que afectan todo el texto.

Los dos componentes principales que afectan la importancia de un término en un documento son el factor de frecuencia de término por sus siglas en inglés Term Frequency (TF) y el factor de frecuencia de documento inverso por sus siglas en inglés Inverse Document Frequency (IDF) [87]. La TF se refiere al número de veces que aparece una palabra clave en todo el corpus y la IDF se utiliza principalmente para reducir el efecto de algunas palabras comunes en todos los documentos que tienen poco efecto en el texto analizado.

El valor TF-IDF de una palabra se calcula multiplicando el componente local TF y el componente global IDF, cuanto mayor sea la importancia de una palabra para un artículo, mayor será su valor TF-IDF. Por lo tanto, los primeros valores TF-IDF representan las palabras claves del corpus que se esté trabajando [86].

Balanceo de datos:

El desbalanceo de datos en el ML se refiere a una distribución desigual de clases dentro de un conjunto de datos. Este problema se encuentra principalmente en tareas de clasificación en las que la distribución de clases o etiquetas en un conjunto de datos determinado no es uniforme [88]. El método sencillo para resolver este problema es el método de remuestreo agregando registros a la clase minoritaria o eliminando registros de la clase mayoritaria.

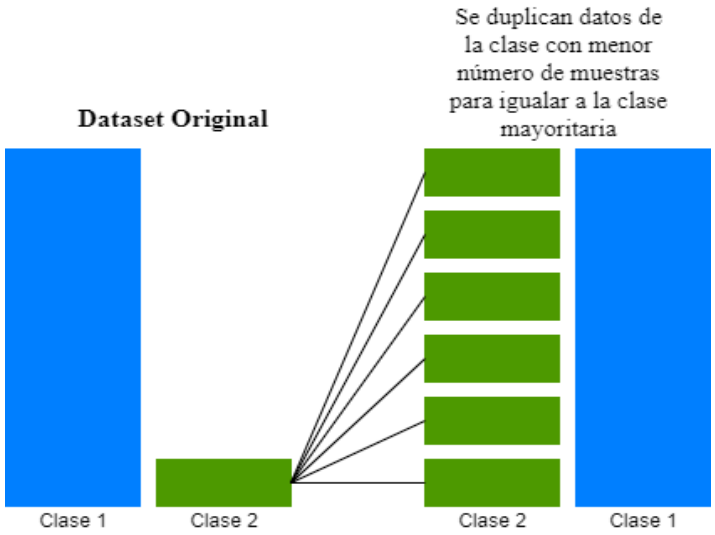
Cuando un conjunto de datos está desequilibrado es difícil obtener un modelo predictivo significativo y confiable debido a la falta de información para aprender sobre el evento

minoritario, por lo tanto, aplicar balanceo de datos implica volver a muestrear para reducir el desequilibrio de clases. Las dos técnicas básicas de muestreo incluyen el sobremuestreo aleatorio o Random Oversampling (ROS) por sus siglas en inglés y el submuestreo aleatorio o Random Undersampling (RUS) por sus siglas en inglés [8]. El sobremuestreo duplica aleatoriamente las muestras de clases minoritarias, mientras que el submuestreo descarta aleatoriamente las muestras de clases mayoritarias para modificar la distribución de clases.

Balanceo al mayor o Sobremuestreo:

El sobremuestreo se puede realizar aumentando la cantidad de instancias o muestras de clases minoritarias produciendo nuevas instancias o repitiendo algunas de ellas para igualar en cantidad a las clases con mayor cantidad de muestras (ver Figura 8).

Figura 8. Proceso para balancear al mayor (Sobremuestreo)

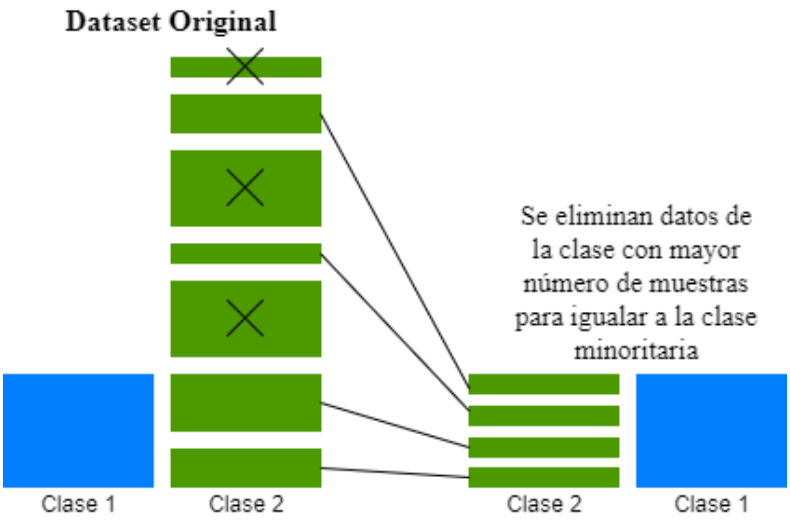


Elaboración propia

Balanceo al menor o Submuestreo:

El submuestreo es el proceso de disminuir la cantidad de instancias o muestras objetivo de las clases mayoritarias con el fin de tener una misma cantidad de datos entre todas las clases a analizar y conseguir el balanceo entre categorías (ver Figura 9).

Figura 9. Proceso para balancear al menor (Submuestreo)



Elaboración propia

Métricas:

Evaluar algoritmos de aprendizaje automático es una parte esencial de cualquier proyecto, ya que el modelo puede brindar resultados satisfactorios cuando se evalúa con una métrica, pero puede dar resultados deficientes cuando se evalúa con otras métricas. La métrica más común utilizada es llamada ***Accuracy*** para medir el rendimiento del modelo desarrollado, sin embargo, no es suficiente para juzgar realmente el modelo. A continuación, se explican unas de las métricas utilizadas para la evaluación de desempeño de los algoritmos de ML según [6] y [89], para esto, es importante entender el significado de los siguientes cuatro términos también brindados por [6]:

- **Verdaderos positivos** (**TP,** por sus siglas en inglés): los casos en los que se predijo SÍ y la salida real también era SÍ.
- **Verdaderos negativos (TN,** por sus siglas en inglés**)**: los casos en los que se predijo NO y la salida real era NO.
- **Falsos positivos (FP,** por sus siglas en inglés**)**: los casos en los que se predijo SÍ y la salida real era NO.
- **Falsos negativos (FN,** por sus siglas en inglés**)**: los casos en los que se predijo NO y la salida real era SÍ.

Exactitud (*Accuracy*): es la relación entre el número de predicciones correctas y el número total de muestras de entrada (ver Ecuación 8).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (8)$$

Precisión: es una métrica que cuantifica el número de predicciones positivas correctas realizadas. Se calcula como la proporción de verdaderos positivos predichos correctamente dividida por el número total de verdaderos y falsos positivos predichos (ver Ecuación 9).

$$Precision = \frac{TP}{TP+FP} \qquad (9)$$

Sensibilidad (*Recall*): es una métrica que cuantifica el número de predicciones positivas correctas realizadas a partir de todas las predicciones positivas que podrían haberse realizado. Se calcula como la proporción de verdaderos positivos predichos correctamente dividida por el número total de verdaderos positivos y falsos negativos que podrían predecirse (ver Ecuación 10).

$$Recall = \frac{TP}{TP+FN} \qquad (10)$$

Valor-F1 (*F1-score*): La puntuación F1 es la media armónica entre precisión y sensibilidad. El rango de la puntuación F1 es [0, 1]. Le dice cuán preciso es su clasificador (cuántas instancias clasifica correctamente), así como cuán robusto es (no pierde una cantidad significativa de instancias).
Cuanto mayor sea el F1 Score, mejor será el rendimiento de nuestro modelo. Matemáticamente, se puede expresar como la Ecuación 11.

$$F1 - Score = 2 \times \frac{1}{\frac{1}{Precision}+\frac{1}{recall}} \qquad (11)$$

Matriz de confusión (CM): Una matriz de confusión resume el número de predicciones realizadas por un modelo para cada clase y las clases a las que realmente pertenecen esas predicciones. Ayuda a comprender los tipos de errores de predicción que comete un modelo.

Reporte de clasificación (CR): crea un reporte de texto que muestra las principales métricas de clasificación, como precisión, Sensibilidad, Valor-F1 y soporte, este último es el número de ocurrencias de la clase dada en el conjunto de datos [4], [5].

Curva ROC: El análisis de curvas ROC por sus siglas en inglés *Receiver Operating Characteristic curve* o Característica Operativa del Receptor constituye un método estadístico para determinar la exactitud diagnóstica de los datos etiquetados como *test*, siendo utilizadas con tres propósitos específicos: determinar el punto de corte de una escala continua en el que se alcanza la sensibilidad y especificidad más alta, evaluar la capacidad discriminativa que los datos de *test* tienen para diferenciar categorías y comparar la capacidad discriminativa de dos o más datos de *test* categorizados que expresan sus resultados como escalas continuas [7].

En las curvas ROC, nos interesa que la curva se acerque lo máximo posible a la esquina superior izquierda de la gráfica, de manera que el hecho de aumentar la sensibilidad (*Recall*) no haga que nuestro modelo introduzca más falsos positivos.

5.2 REFERENTE NORMATIVO

A través de la Ley 1581 de 2012 y el Decreto 1377 de 2013, se desarrolla el derecho constitucional que tienen todas las personas a conocer, suprimir, actualizar y rectificar todo tipo de datos personales recolectados, almacenados o que hayan sido objeto de tratamiento en bases de datos en las entidades del públicas y privadas [90].

Se requiere que se cumpla con todo el marco legal de *Habeas Data;* recurso de agravio constitucional que protege dos derechos fundamentales: el derecho a la información y la autodeterminación informativa o protección de datos personales; ambos, forman parte del ámbito de los derechos humanos, reconocidos y protegidos por los Tratados Internacionales y las Cartas Constitucionales de los diferentes países en los que impera el estado de derecho [91].

SIGMA Ingeniería S.A está certificada bajo modelos internacionales de calidad CMMI-

DEV nivel 3 que supone el reconocimiento a los altos niveles de calidad y exigencia los desarrollos y productos de hardware y software de la empresa, además, cuenta con certificaciones como el ISO27000 que garantiza las buenas prácticas para el establecimiento, implementación, mantenimiento y mejora de Sistemas de Gestión de la Seguridad de la Información. Por lo tanto, requiere de permisos legales por parte de la empresa para hacer uso de los datos con los que posee; que son imprescindibles para el desarrollo y estudio del presente trabajo.

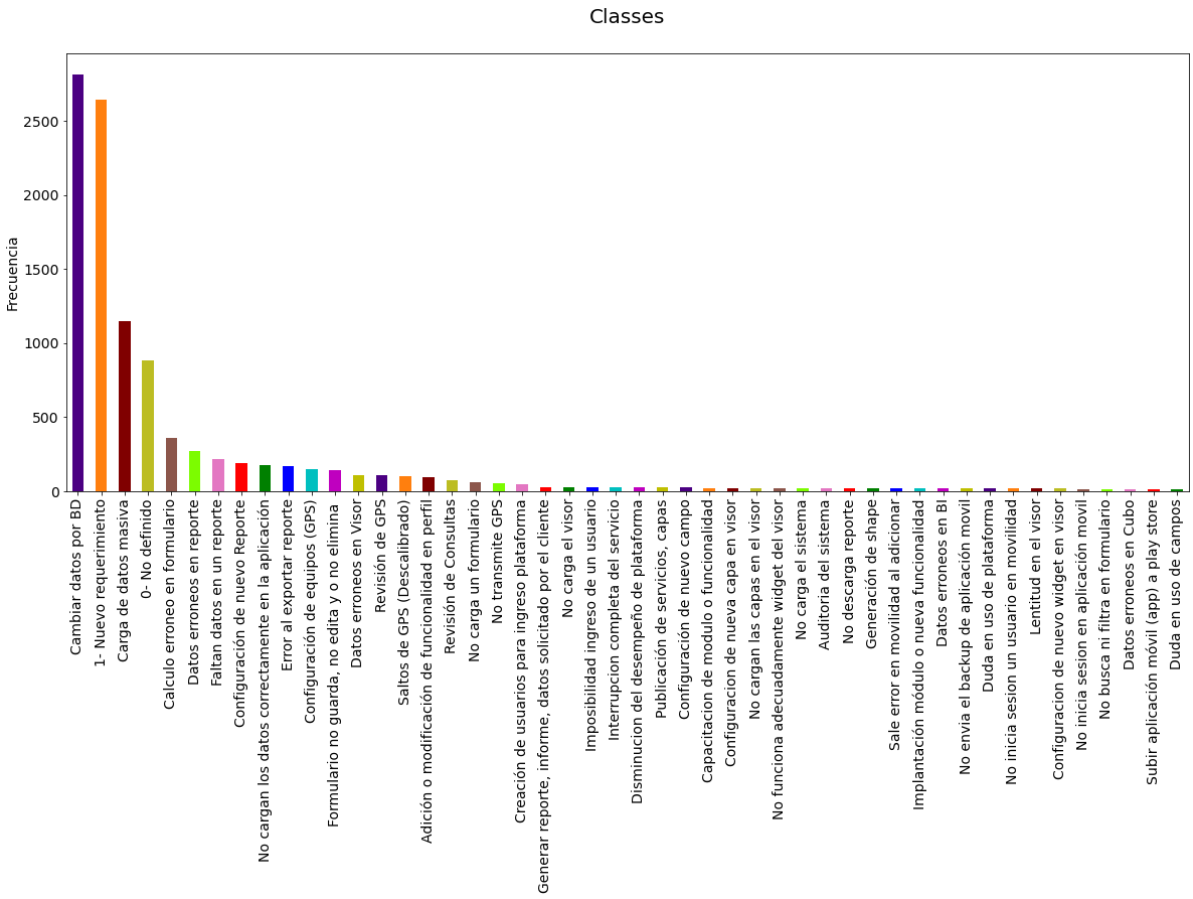
5.3 REFERENTE CONTEXTUAL

SIGMA Ingeniería S.A es una empresa de Manizales enfocada en el desarrollo de Software de georreferenciación y Sistemas de Información Geográfica para la gestión pública en Colombia de los sectores energético, sanitario y ambiental. La empresa se destaca por la innovación, la aplicación de técnicas de la industria 4.0 y el manejo de datos dentro de los sectores que se abarcan en sus líneas de negocio conocidas como Geoambiental; que es una herramienta tecnológica basada en información gerencial y geográfica que tiene como fin fortalecer los procesos y procedimientos de las organizaciones que gestionan el medio ambiente, Geoaseo; que logra optimizar el ejercicio de las empresas de aseo en operaciones como rutas, recolección, barrido, entre otras variables en las ciudades y municipios de un país y, Geolumina que consolida y sistematiza a empresas y concesiones de alumbrado, permitiendo dar cumplimiento al reglamento técnico de iluminación y alumbrado público. Todas las bases de datos generadas por los procesos internos y externos de la empresa se almacenan en una plataforma organizacional denominada Timework, la cual es una herramienta propia de SIGMA Ingeniería S.A que permite registrar los tiempos de las actividades ejecutadas, priorizaciones de los equipos de trabajo, gestión de compromisos, gestión de riesgos y planeación estratégica diaria de la organización. En este sentido, es importante tener en cuenta el concepto **Software as a Service (SaaS)**, debido a que es un modelo de distribución de software donde el soporte lógico y los datos que se manejan se alojan en servidores de la compañía a los que se accede vía internet desde un cliente que

permite a los usuarios conectarse a aplicaciones basadas en la nube a través de Internet y usarlas [92]. Es de esta manera como se brinda un fácil acceso a la información por parte de los clientes de SIGMA Ingeniería S.A para su posterior uso y descarga mediante la herramienta Timework.

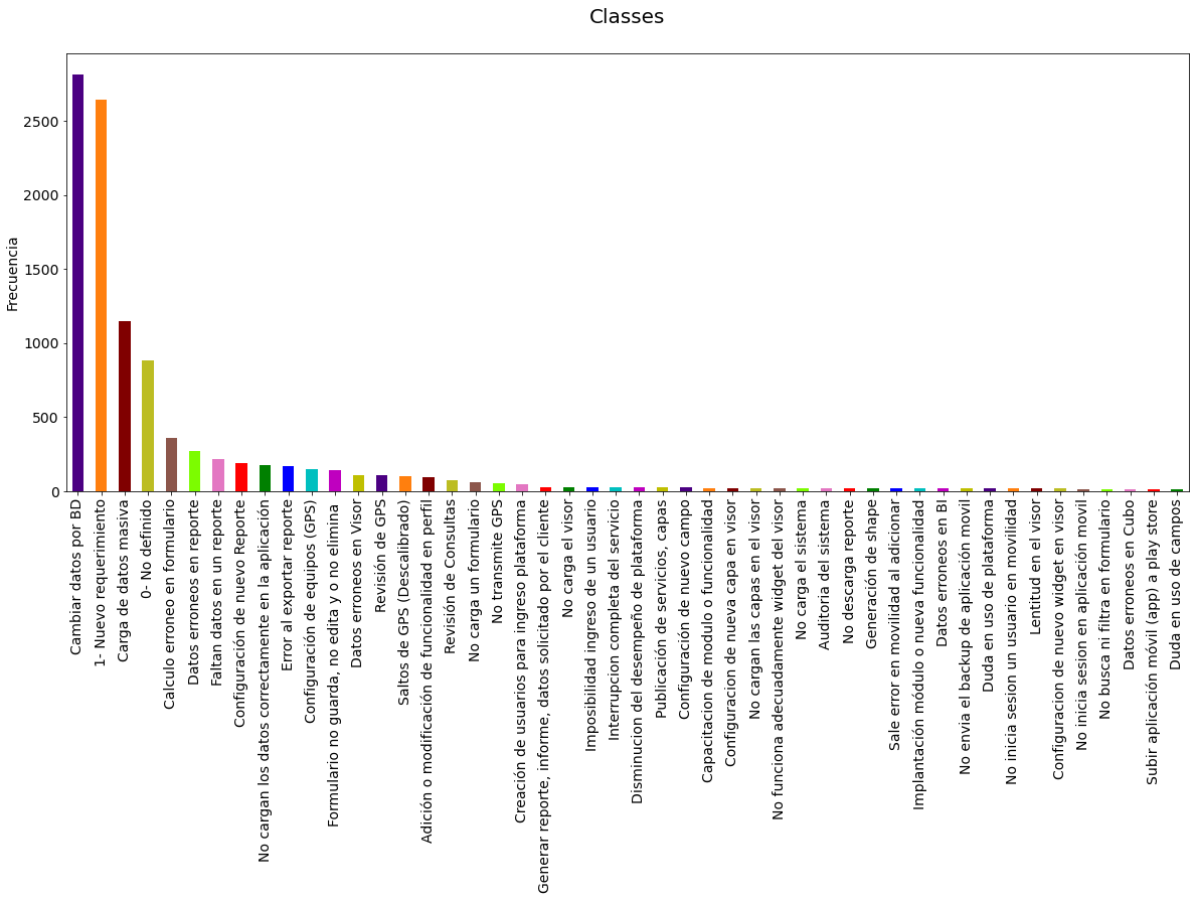
Adicionalmente, se hace necesario el uso de técnicas de NLP para permitir el entendimiento entre las máquinas y las personas mediante el uso de lenguas naturales, como el español; los datos que se trabajarán están compuestos por textos escritos almacenados en forma de tickets de servicio, que son generados por los clientes y el personal de la empresa; por lo tanto, el modelo computacional a desarrollar debe estar en la capacidad de interpretar estos textos y entender su significado e intención, tal como lo haría una persona, para posteriormente, clasificarlo en las categorías definidas por la organización, con un alto porcentaje de precisión mediante las técnicas de NLP explicadas en la sección 5.1.

Al analizar la estructura de los datos se evidencia un desbalanceo significativo entre las categorías para clasificar los tickets de servicio (ver Figura 10. Frecuencia por categoría a partir de bases de datos de Timework



). Esto se debe a que todas las categorías encontradas en la plataforma Timework cuentan con todas las posibles situaciones que un cliente puede enfrentar; hay incidencias más comunes que otras y por esta razón existe un desbalanceo en las categorías para esta etapa del proceso, es por este motivo que a partir de las técnicas de balanceo de datos explicadas en la sección 5.1 se pretende darle solución a esta problemática.

Figura 10. Frecuencia por categoría a partir de bases de datos de Timework



Elaboración propia. Basado en bases de datos de Timework

6 OBJETIVOS

6.1 OBJETIVO GENERAL

- Construir un modelo computacional basado en ML y NLP para la clasificación de incidencias y protocolos de solución en la empresa SIGMA Ingeniería S.A

6.2 OBJETIVOS ESPECÍFICOS

1. Caracterizar la información brindada desde Timework para preparar una línea base de prueba de los modelos computacionales propuestos en esta tesis.
2. Evaluar el desempeño de técnicas de ML supervisadas más utilizadas (SVM, ET, RF, LR, DT, LDA, NB, KNN) aplicadas a la clasificación de texto a partir de modelos de referencia validados basados en procesamiento de lenguaje natural.
3. Implementar un modelo computacional de aprendizaje de máquina para la clasificación de los tickets de servicio brindadas por Timework y la entrega de protocolos de solución basados en la técnica de ML resultante del objetivo específico 2 y NLP.
4. Validar el funcionamiento y desempeño del modelo computacional mediante análisis estadísticos y juicio de expertos de la organización.

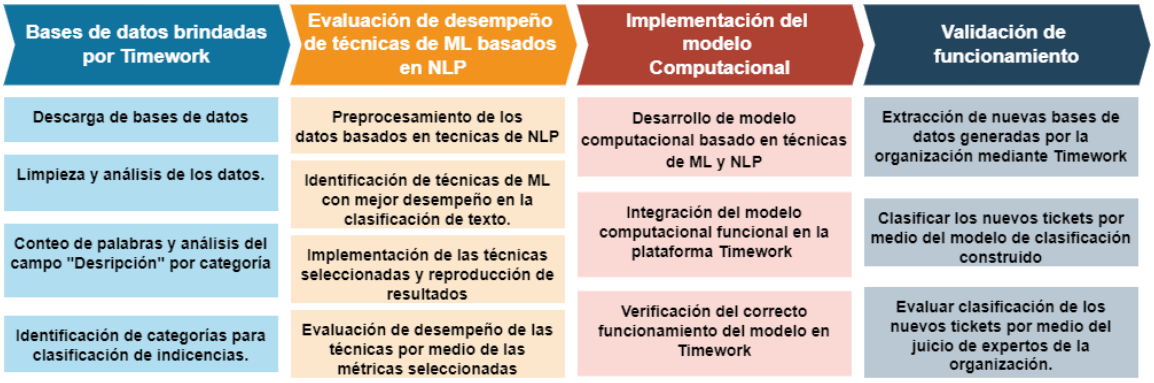
7 METODOLOGÍA

7.1 ENFOQUE Y TIPO DE INVESTIGACIÓN

El tipo de investigación a la que pertenece el presente trabajo es aplicada con un enfoque cuantitativo y un diseño no experimental transversal, satisface el nivel de madurez tecnológica TRL 4, debido a que se pretende llegar a un desarrollo tecnológico capaz de ejecutar todas las funciones planteadas dentro del alcance del proyecto para la empresa SIGMA Ingeniería S.A en un entorno de pruebas o de laboratorio, y se espera que el modelo presente pruebas superadas de factibilidad en condiciones de operación y funcionamiento simuladas ya que la muestra a trabajar son datos generados por la organización en los últimos cuatro años y el propósito es analizar estos datos para brindarle solución al problema de investigación planteado.

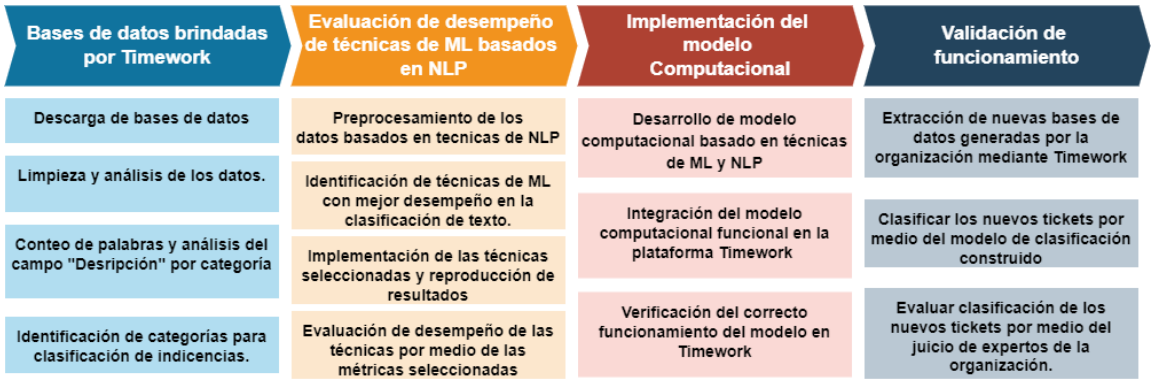
7.2 DISEÑO DE LA INVESTIGACIÓN

En la



se presenta el diseño metodológico del proyecto teniendo en cuenta cada uno de los objetivos propuestos.

Figura 11. Diseño metodológico de la investigación.



Elaboración propia

7.3 UNIDAD DE TRABAJO Y UNIDAD DE ANÁLISIS

La unidad de análisis de investigación es el área de soporte técnico de la empresa SIGMA Ingeniería S.A.

Para el objetivo 1, se busca caracterizar la información brindada desde la plataforma Timework, la muestra a trabajar cuenta con 14,385 tickets de servicios que al ser sometidos a procesos de análisis y limpieza manual pasaron a ser 2,146 datos. Cada uno de estos tickets cuenta con diferentes campos como lo son la descripción de las incidencias, la categoría a la que pertenece, el cliente y la línea de negocio a la que pertenece. Estos campos son los seleccionados para realizar el respectivo análisis y procesamiento mediante el lenguaje de programación Python que soporta un desarrollo amplio en librerías y aplicaciones relacionadas con el análisis de datos y la inteligencia artificial.

El objetivo 2, busca evaluar las diferentes técnicas de ML aplicados a la clasificación y procesamiento de texto por medio de diferentes condiciones de pruebas establecidas y técnicas recomendadas por la literatura basadas en NLP para el tratamiento y preparación de los datos.

El objetivo 3, busca implementar en la herramienta organizacional Timework el modelo computacional que permita realizar la clasificación de los tickets de servicio generados en

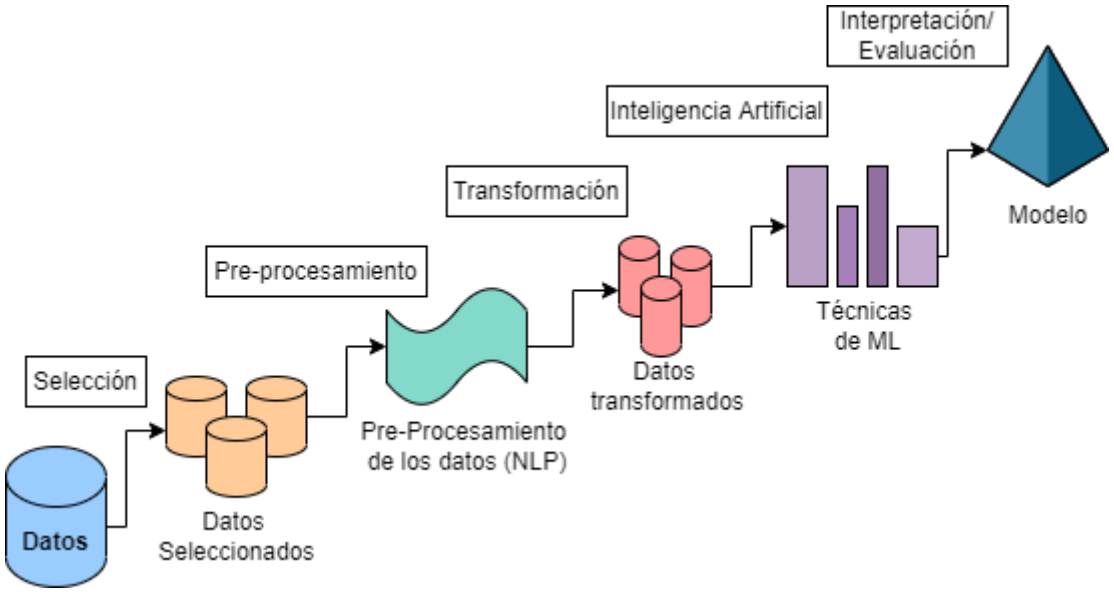
la empresa y brindar el protocolo de solución del mismo.

El objetivo 4, finalmente busca validar por medio del juicio de expertos de la organización el correcto funcionamiento del modelo computacional construido.

7.4 INSTRUMENTOS DE RECOLECCIÓN, PROCEDIMIENTO Y TÉCNICAS

El instrumento de recolección utilizado es la herramienta organizacional Timework que contiene las bases de datos necesarias para el desarrollo del proyecto mencionados en la Sección 7.5. Timework, permite conocer, gestionar y controlar las solicitudes de los clientes, registrar los tiempos de las actividades y requerimientos ejecutados.
Los procedimientos y las técnicas aplicadas se describen según los objetivos propuestos en la Figura 12, allí se presentan los respectivos pasos para el proceso de analítica de datos, mostrando además la metodología para lograrlo.

Figura 12. Proceso de analítica de datos



Elaboración propia

7.5 CARACTERIZACIÓN DE LA INFORMACIÓN BRINDADA DESDE TIMEWORK PARA PREPARAR LA LÍNEA BASE DE PRUEBA DE LOS MODELOS COMPUTACIONALES PROPUESTOS PARA ESTE TRABAJO.

El primer paso del procedimiento metodológico consistió en la exploración y análisis de los datos a trabajar, para este paso se cuenta con el diccionario de datos de las descripciones brindados por SIGMA Ingeniería S.A; su contenido se muestra en la Tabla 1. Los datos en este archivo son en su totalidad datos tipo “String”.

Tabla 1. Diccionario de Datos

Nombre del archivo:	**Descripcion.xlsx**
Cantidad de datos:	**14385 x 5**
Campo	**Descripción**
Código del ticket (tik_codigo)	Código alfanumérico de la solicitud asignado por la plataforma
Descripción (descripcion)	Descripción de la incidencia, requerimiento o petición presentada por el cliente.
Categoría (categoria)	Categoría interna a la que pertenece la incidencia, requerimiento o petición presentada por el cliente.
Nombre del cliente (nombre_cliente)	Nombre del cliente asociado al ticket de servicio creado
Línea de Negocio (Linea_Negocio)	Línea de negocio sobre la cual se aplica la incidencia, puede ser Geolúmina, Geoaseo, Geoambiental.

Elaboración propia

Estos datos cuentan con todos los tickets de servicio que se han generado en la compañía, el campo llamado “Descripción” cuenta con un texto plano y abierto de las especificaciones brindadas por el cliente con su respectiva categoría, cliente y línea de negocio.

En esta etapa del proceso se hizo un análisis profundo con respecto a las palabras y número de palabras más usadas en todos los registros del campo “Descripción”. Al trabajar únicamente con contenidos textuales se hace este análisis para establecer las palabras claves más usadas, la frecuencia y la relación entre el ticket de servicio y la categoría perteneciente para usar como guía y referencia al momento de registrar nuevos requerimientos.

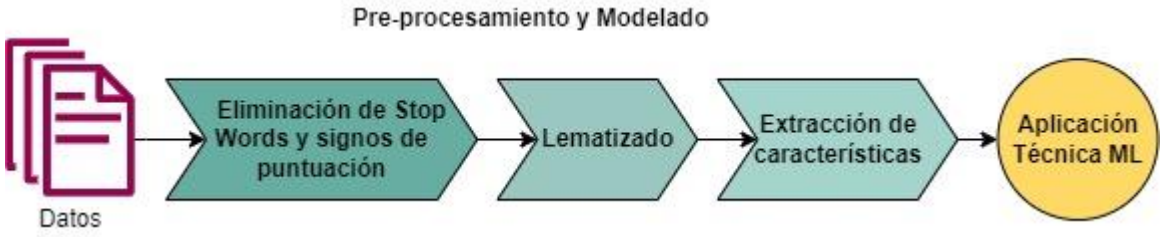
7.6 EVALUACIÓN DEL DESEMPEÑO DE LAS DIFERENTES TÉCNICAS DE APRENDIZAJE DE MÁQUINA-APLICADAS AL PROCESAMIENTO DE TEXTO A PARTIR DE MODELOS DE REFERENCIA VALIDADOS BASADOS EN PROCESAMIENTO DE LENGUAJE NATURAL (NLP).

7.1.1 Pre-procesamiento de los datos:

Como primera etapa del pre-procesamiento, se tomaron las bases de datos explicadas en la Sección 7.5 y se procedió a realizar una limpieza a los datos, en esta etapa se eliminaron las celdas vacías, los datos inconsistentes y se hizo limpieza manual de las bases de datos con expertos de la empresa para su posterior uso.

Se realizó una búsqueda e investigación de las diferentes técnicas de NLP que fueron aplicadas con éxito a las bases de datos trabajadas, teniendo en cuenta su contenido, intensión y objetivo del proyecto, a partir de esta investigación, se ha llegó a la conclusión de aplicar las técnicas de NLP mostradas en la Figura 13:

Figura 13. Técnicas de NLP de pre-procesamiento y modelado.



Elaboración propia

Se procedió a eliminar los *stop words*, los signos de puntuación de todos los textos contenidos en el campo de "Descripción", y se aplicó la técnica de lematización; técnica explicada en la Sección 5.1.

A las palabras resultantes obtenidas al aplicar los procesos anteriores se le denominará vocabulario. En este punto se aplicaron distintas técnicas de transformación de características [93], con el objetivo de reducir el tamaño de los datos a procesar donde cada una de las palabras se codificó en valores de coma flotante, se le asignó el peso correspondiente a la importancia obtenida dentro del texto, y de esta manera, se detectaron las palabras más relevantes a tener en cuenta para su uso en los algoritmos.

En este estudio se usaron técnicas de balanceo de datos mediante "imbalanced-learn" [94] y optimización de parámetros mediante "Grid Search". Se realizaron experimentos sin ningún tipo de transformación (usando los datos crudos) y usando una unión entre balanceo y optimización junto con el respectivo pre-procesamiento. Todas las transformaciones se realizaron usando el lenguaje de programación Python versión 3.8 [95] y la librería para ML Scikit-learn [96].

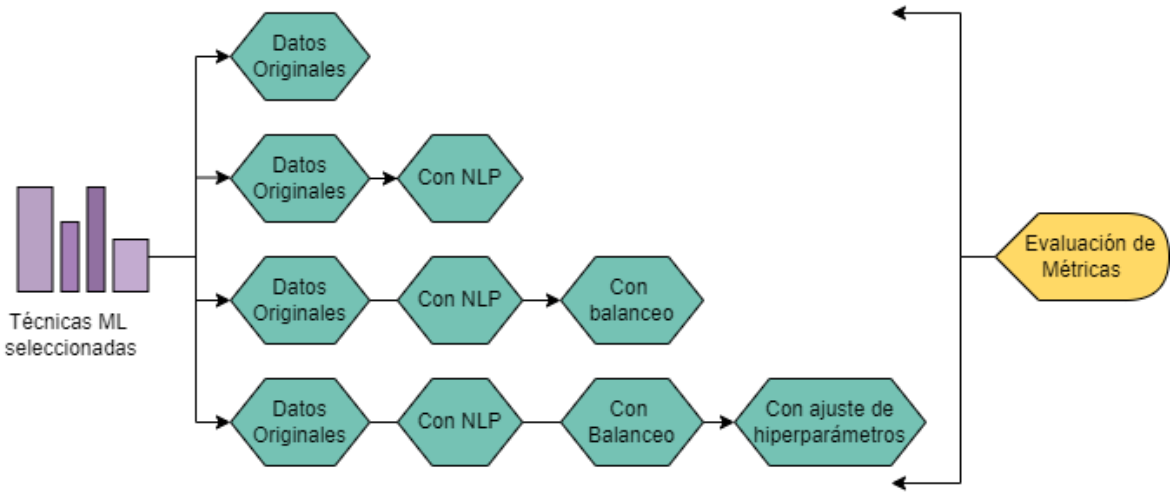
7.1.2 Selección de técnicas de ML

Los datos resultantes del proceso anterior (ver Sección 7.1.1) fueron evaluados mediante técnicas de ML con los datos pre-procesados utilizando técnicas de balanceo de datos, técnicas de NLP y optimización de parámetros. Se propuso usar los siguientes algoritmos de ML: SVM, ET, RF, LR, DT, LDA, NB y KNN y fueron evaluados mediante las métricas mencionadas en la Sección 5.1.

Se almacenaron todos los resultados obtenidos por los algoritmos y arquitecturas de ML descritos anteriormente usando los conjuntos de experimentos planteados en la Figura 14, con el objetivo de comparar y seleccionar las mejores combinaciones a través de las métricas evaluadas e implementar el modelo resultante con la técnica de mejor desempeño.

Al finalizar este objetivo, se conoció cual es el conjunto de experimentos, técnica de ML y técnicas de pre-procesamiento, que presentaron los mejores valores en las métricas anteriormente seleccionadas.

Figura 14. Conjunto de experimentos para la aplicación de técnicas de ML



Elaboración propia

El proceso que se llevó a cabo para la aplicación de las técnicas de ML corresponde a cinco etapas principales:

1. Carga de la lectura e interpretación de las bases de datos.
2. Aplicación del preprocesamiento para la limpieza de los datos mediante técnicas de NLP.
3. Balanceo de los datos y transformación de características

4. Exploración y aplicación de ocho modelos de ML.
5. Selección de la técnica de ML con el mejor desempeño en clasificación de requerimientos.

Este proceso se resume en el diagrama mostrado en la

Figura 15. Proceso a llevar a cabo para la clasificación de requerimientos.

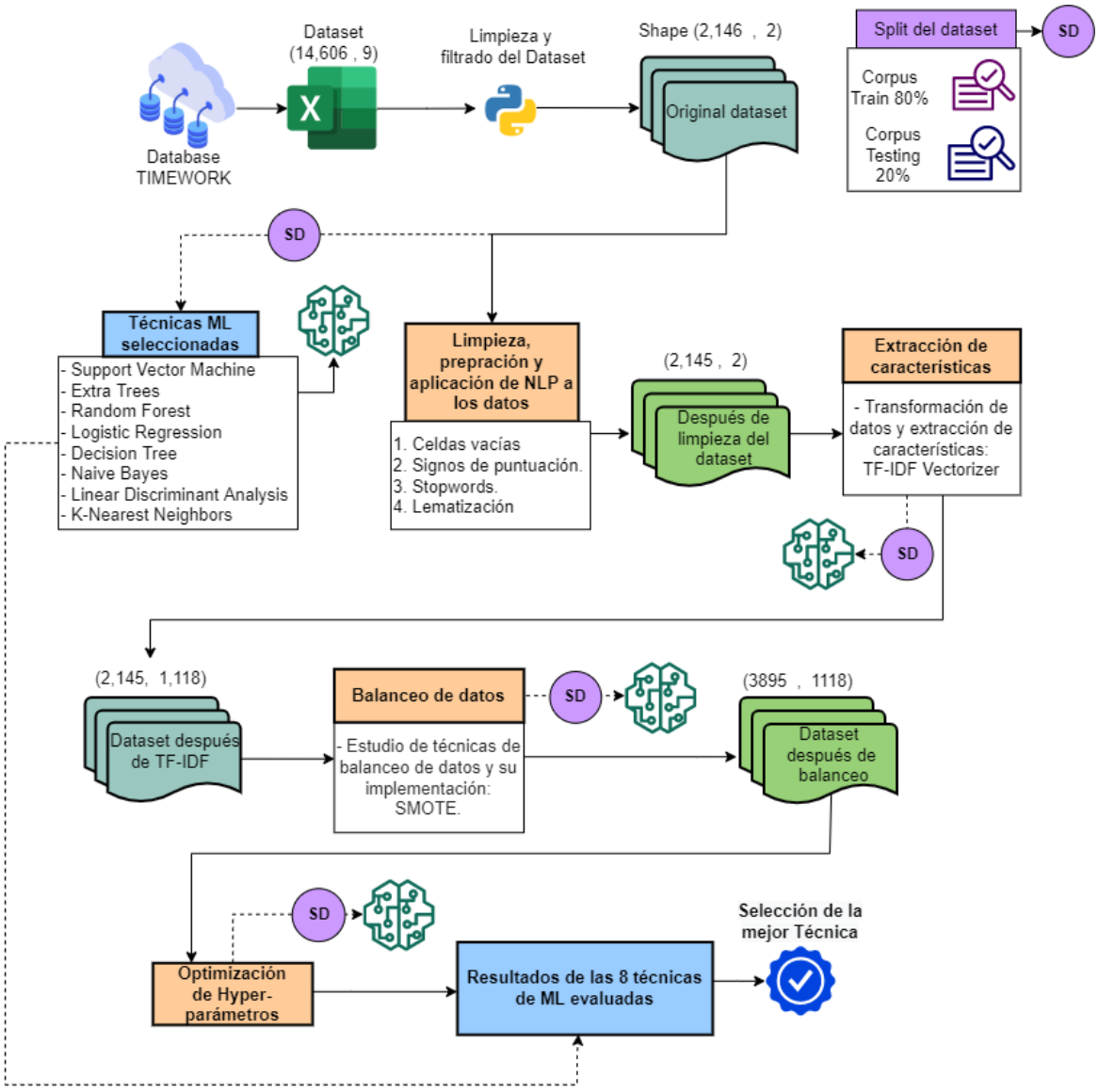
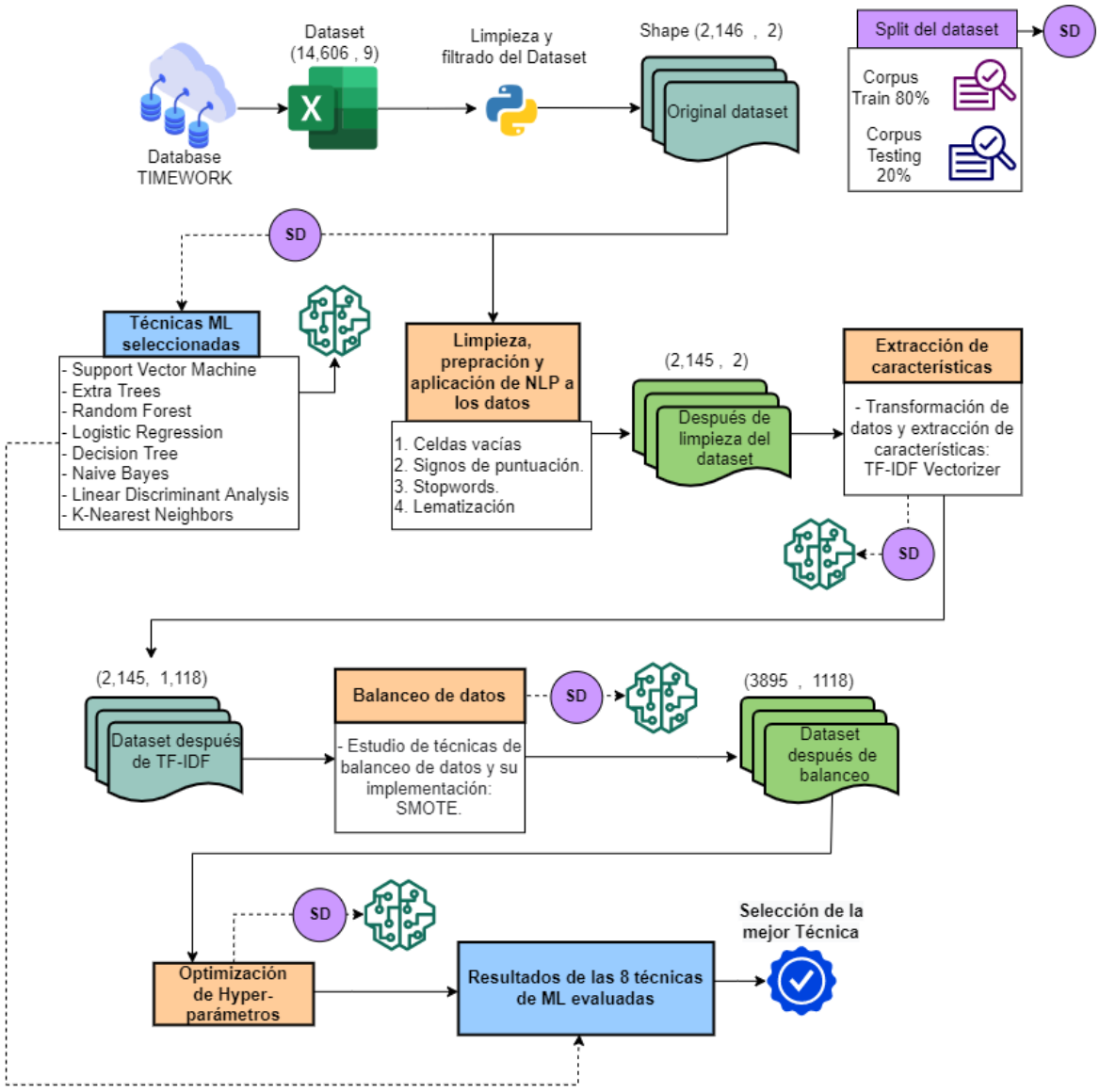


Figura 15. Proceso a llevar a cabo para la clasificación de requerimientos.



Elaboración propia

7.7 IMPLEMENTACIÓN DE MODELO COMPUTACIONAL DE APRENDIZAJE DE MÁQUINA PARA LA CLASIFICACIÓN DE LOS TICKETS DE SERVICIO BRINDADOS POR TIMEWORK BASADOS EN TÉCNICAS DE APRENDIZAJE DE MÁQUINA Y PROCESAMIENTO DE LENGUAJE NATURAL.

El desarrollo del modelo computacional se ha llevado a cabo a partir de los pasos mostrados en la Figura 16.

Figura 16. Pasos para la implementación del modelo computacional en la empresa.

Elaboración propia

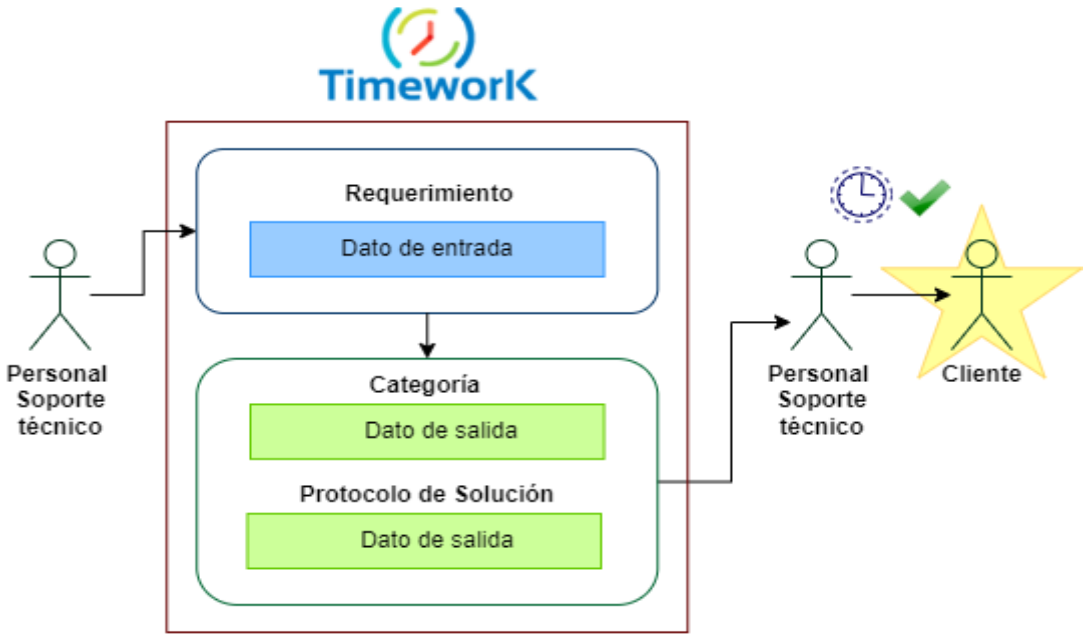
Siguiendo el orden presentado en la Figura 16; el modelo computacional se encargará de realizar la clasificación de categorías de los requerimientos presentados por el cliente y además brindará el protocolo de solución al respectivo requerimiento, de esta manera, el

personal del área de soporte técnico de la empresa pueda usarlo directamente con el cliente y así brindarle una solución más rápida y eficiente.

Después del desarrollo del modelo computacional, se procede a implementar el modelo desarrollado en la plataforma Timework, que como se nombró con anterioridad, es la herramienta que organiza todos los procesos externos e internos de la organización y, lleva el historial de todos los tickets de servicio trabajados en el área de soporte técnico.

El proceso que se realizó fue implantar una sección de analítica de datos en la plataforma Timework para la categorización y predicción de nuevos tickets de servicio. De esta manera, el área de soporte técnico podrá ingresar directamente el requerimiento presentado por el cliente y el modelo se encargará de brindar tanto la categoría a la que pertenece el ticket de servicio junto con los protocolos de solución a llevar a cabo. El proceso realizado sigue la Figura 17:

Figura 17. Proceso a realizar con el modelo computacional implementado en Timework.

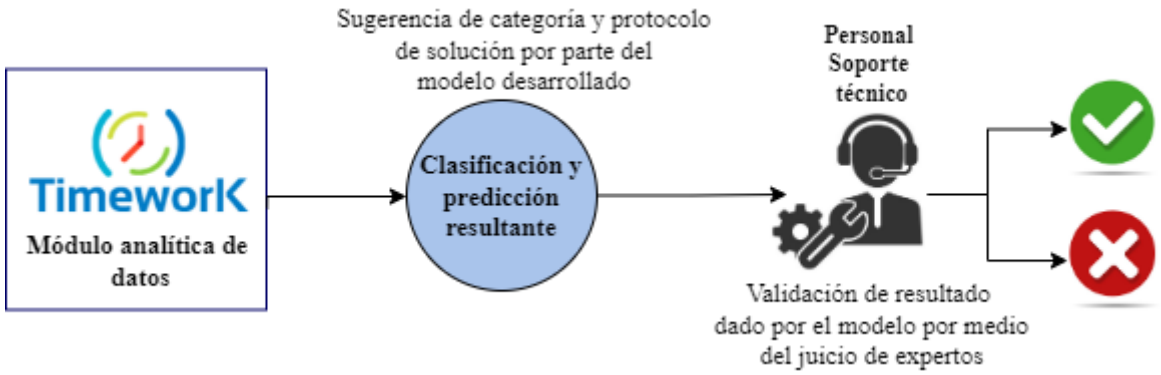


Elaboración propia

7.8 VALIDACIÓN DEL FUNCIONAMIENTO DEL MODELO COMPUTACIONAL COMPARANDO LOS RESULTADOS OBTENIDOS CON EL JUICIO DE EXPERTOS DE LA ORGANIZACIÓN.

A partir del modelo computacional implantado en Timework en la Sección 7.7 se procedió a evaluar y validar su funcionamiento por medio del juicio del área de soporte técnico de la empresa, la Figura 18**¡Error! No se encuentra el origen de la referencia.** muestra el proceso realizado:

Figura 18. Validación de clasificación y predicción por medio del juicio de expertos



Elaboración propia

La validación del correcto funcionamiento del modelo computacional se llevó a cabo mediante una prueba piloto durante 6 meses en Timework que permite comparar los resultados dados por el modelo y por el área de soporte técnico para medir la eficiencia del mismo. El proceso llevado a cabo se muestra en la Figura 19 y se explica a continuación:

1. Por medio de la prueba piloto, se registraron nuevos tickets de servicio por parte del área de soporte técnico, donde para cada nuevo registro se indicó la categoría correspondiente.

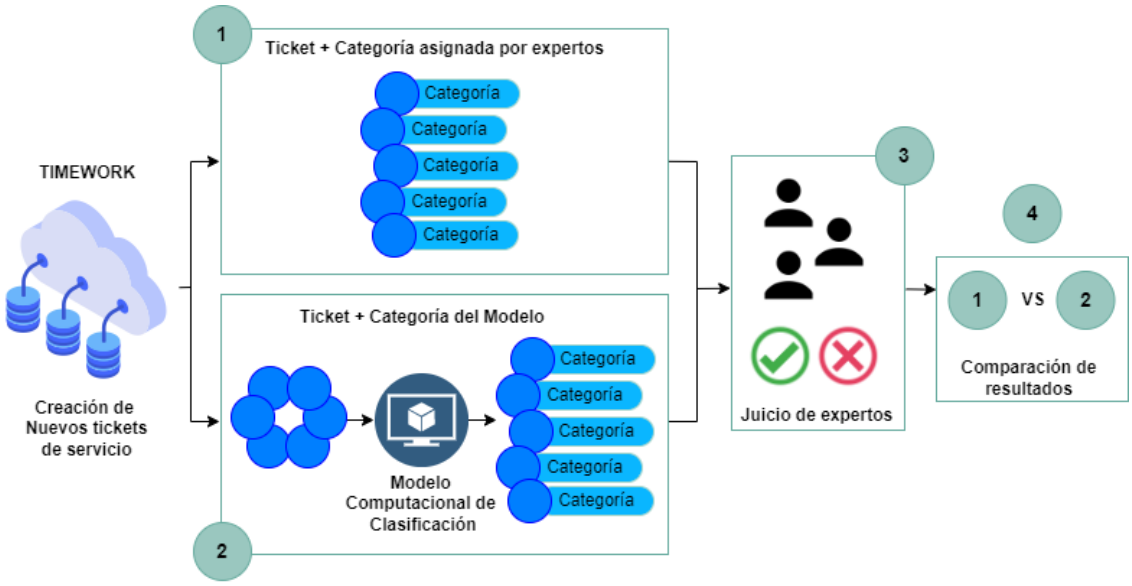
2. Por otro lado, el modelo computacional entregó el resultado de la clasificación realizada con respecto a la descripción ingresada.

3. En este punto, el área de soporte técnico indicó si la categoría predicha correspondía con

la categoría asignada por los expertos. El nuevo requerimiento creado contiene la clasificación realizada por el área de soporte técnico y la predicción hecha por el modelo.

3. Finalmente, se procedió a evaluar la coincidencia entre la categoría de los tickets de servicio clasificados por los expertos y la categoría predicha por el modelo computacional obtenidos durante la prueba piloto.

Figura 19. Proceso de validación del modelo computacional



Elaboración propia

8 RESULTADOS

8.1 RESULTADO OBJETIVO 1

Caracterizar la información brindada desde Timework para preparar una línea base de prueba de los modelos computacionales propuestos en esta tesis.

El primer paso realizado con las bases de datos se centró en el análisis de los datos y la detallada investigación del trabajo realizado por el equipo encargado de soporte técnico y servicio al cliente.

A partir de este análisis se reunieron las condiciones necesarias para proceder a revisar la calidad y pertinencia de los datos, este primer paso conllevó a una limpieza y organización exhaustiva de los registros después de evaluar la calidad de los datos en bruto (bases de datos originales) y analizar los resultados entregados, se pasó de contar con 14,385 datos a contar con 2,146 datos.

La razón por la cual la cantidad de datos bajo un 85% de registros fue debido a:

- La mayoría de los registros no contaban con una descripción acertada y no aplicaba a ninguna de las categorías especificadas al ser información reducida y sin detallar.

- Otra cantidad de registros no contaban con una categoría asignada, por lo que unas de ellas se categorizaron con la ayuda de expertos y otras por el contrario fueron suprimidas.

- Se suprimieron ciertas categorías que eran equivalentes unas a otras y se fusionaron con la autorización de los expertos en la organización.

Esto se debe a que se realizó una limpieza manual de cada uno de los requerimientos que se pretendía organizar en el conjunto de datos para asegurar que cada categoría corresponda efectivamente a la descripción ingresada.

Seguido de este proceso, el primer análisis realizado de las bases de datos es el conteo de

requerimientos por categoría. SIGMA Ingeniería pasó de contar con 48 categorías a contar con 41 categorías. La Tabla 2 muestra las categorías encontradas en Timework junto con la cantidad de registros (frecuencia) por cada una de ellas.

Tabla 2. Categorías contenidas Timework junto con sus frecuencias

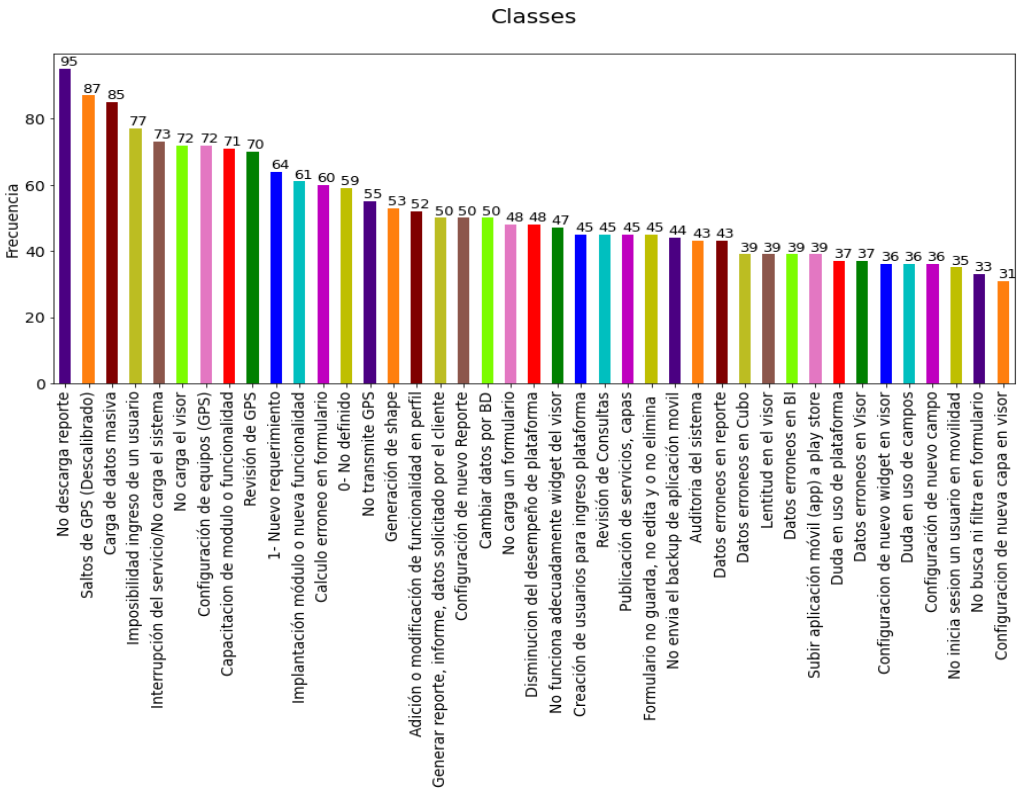
Categoría	**Frecuencia**
No descarga reporte	95
Saltos de GPS (Descalibrado)	87
Carga de datos masiva	85
Imposibilidad ingreso de un usuario	77
Interrupción del servicio/No carga el sistema	73
No carga el visor	72
Configuración de equipos (GPS)	72
Capacitación de módulo o funcionalidad	71
Revisión de GPS	70
1- Nuevo requerimiento	64
Implantación módulo o nueva funcionalidad	61
Calculo erróneo en formulario	60
0- No definido	59
No transmite GPS	55
Generación de shape	53
Adición o modificación de funcionalidad en perfil	52
Generar reporte, informe, datos solicitados por el cliente	50
Configuración de nuevo Reporte	50
Cambiar datos por BD	50
No carga un formulario	48
Disminución del desempeño de plataforma	48
No funciona adecuadamente widget del visor	47
Creación de usuarios para ingreso plataforma	45
Revisión de Consultas	45
Publicación de servicios, capas	45
Formulario no guarda, no edita y o no elimina	45
No envía el backup de aplicación móvil	44
Auditoria del sistema	43
Datos erróneos en reporte	43
Datos erróneos en Cubo	39

Lentitud en el visor	39
Datos erróneos en BI	39
Subir aplicación móvil (app) a play store	39
Duda en uso de plataforma	37
Datos erróneos en Visor	37
Configuración de nuevo widget en visor	36
Duda en uso de campos	36
Configuración de nuevo campo	36
No inicia sesión un usuario en movilidad	35
No busca ni filtra en formulario	33
Configuración de nueva capa en visor	31

Elaboración propia

La Figura 20 presenta un gráfico de barras con la cantidad de tickets de servicio encontradas en cada una de las categorías contenidas en las bases de datos:

Figura 20. Gráfico de barras de cantidad de requerimientos por categoría



Elaboración propia

Posteriormente, se procedió a analizar el campo “Descripción” de las bases de datos, este proceso se analizó de 2 formas:

1. Análisis de las bases de datos originales resultantes (OD).

2. Análisis de las bases de datos originales después de limpieza de datos donde se eliminan los “stop words” y signos de puntuación del campo de descripción (ODL).

Se comenzará con las bases de datos originales (OD), en este proceso se extraen y se cuentan la totalidad de palabras contenidas por requerimiento, la Tabla 3 muestra las medidas estadísticas entregadas a partir del conteo de palabras por ticket de servicio:

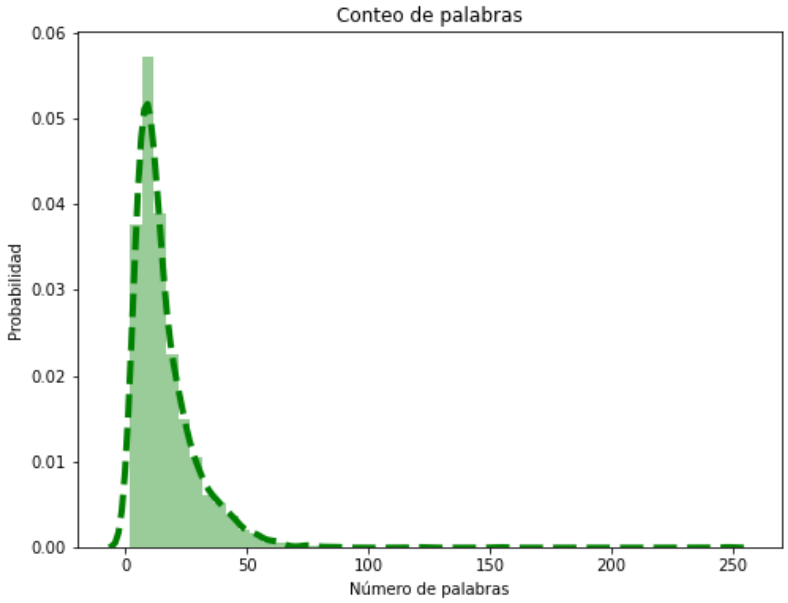
Tabla 3. Medidas estadísticas de las OD a partir del conteo de palabras

OD	
Count	2.146,00
mean	16,05
std	13,52
min	2,00
25%	8,00
50%	12,00
75%	21,00
max	248,00

Elaboración propia

A partir de los datos y las medidas estadísticas se obtiene el siguiente histograma del conteo de palabras (ver Figura 21); donde indica que la cantidad de palabras por requerimiento se encuentra en un rango de 10 a 20 palabras en promedio.

Figura 21. Histograma de conteo de palabras para OD.



Elaboración propia

Se hizo el cálculo de las 25 palabras más usadas por los miembros del equipo técnico en SIGMA Ingeniería S.A en el registro de tickets de servicio, los resultados se observan enumerados en la Tabla 4:

Tabla 4. 25 palabras más usadas en el campo de Descripción con “stop words”.

#	Palabra	Frecuencia
1	de	2654
2	el	1254
3	en	1022
4	la	958
5	se	613
6	no	609
7	y	608
8	que	606
9	los	532
10	del	483
11	para	470
12	las	368
13	a	348
14	al	312
15	con	288
16	datos	263
17	visor	248
18	por	234
19	Se	230
20	un	215
21	reporte	203
22	error	197
23	plataforma	183
24	formulario	181
25	No	168

Elaboración propia

La Figura 22 muestra en el diagrama de nubes de palabras cómo las palabras con mayor tamaño que implican una mayor frecuencia son comúnmente las “stop words” (ver sección 5.1); al no brindar información útil para el análisis el campo, se precede a suprimirlas y comenzar con el análisis del punto 2. Con las ODL, es decir, las bases de datos originales posteriores a la limpieza al eliminar “stop words” y signos de puntuación del campo de descripción.

Figura 22. Nube de palabras de las OD.

Elaboración propia

Para este caso, el análisis estadístico para el conteo de palabras por requerimiento para las ODL se encuentra en la Tabla 5:

Tabla 5. Medidas estadísticas de las ODL a partir del conteo de palabras

ODL	
count	2.146,00
mean	9,73
std	7,63
min	2,00
25%	5,00
50%	8,00
75%	12,00
max	137,00

Elaboración propia

A partir de los datos y las medidas estadísticas se obtiene el histograma mostrado en Figura 23 del conteo de palabras:

Figura 23. Histograma de conteo de palabras para ODL.

Elaboración propia

La Figura 23 indica que la cantidad de palabras por requerimiento después de suprimir las "stop words" se encuentra en un rango más reducido y además con una menor probabilidad de usabilidad comparado con el estudio OD. Para este caso, también se hizo el cálculo de las 25 palabras más usadas y se observa que el número de repeticiones es más bajo, sin embargo, ahora las palabras tienen sentido y significado por sí solas, lo que permitiría hacer un análisis profundo con respecto a las palabras más representativas de las bases de datos. Los resultados se observan en enumerados en la Tabla 6.

Tabla 6. 25 palabras más usadas en el campo de Descripción sin “stop words”.

#	Palabra	Frecuencia
1	datos	263
2	visor	248
3	reporte	203
4	error	197
5	plataforma	183
6	formulario	181
7	No	168
8	sistema	163
9	usuario	137
10	favor	132
11	nuevo	114
12	ciclos	113
13	colaboración	104
14	OT	102
15	carga	101
16	GPS	101
17	campo	96
18	solicito	95
19	permite	93
20	realizar	82
21	capas	82
22	requiere	77
23	presenta	77
24	solicita	74
25	widget	73

Elaboración propia

La Figura 24 entrega la nube de palabras de las palabras más representativas y usadas en las bases de datos al remover los “stop words”.

Figura 24. Nube de palabras de las ODL.

Elaboración propia

Finalmente, se realiza la comparación de conteo de palabras para OD y ODL. La Tabla 7 indica la diferencia de palabras encontradas a partir de la descripción sin eliminar “stop words” y al eliminar “stop words”.

Tabla 7. Diferencia en frecuencia de palabras entre OD y ODL

OD - ODL	
count	2.146,00
mean	6,32
std	6,32
min	0,00
25%	2,00
50%	5,00
75%	8,00
max	111,00

Elaboración propia

La diferencia promedio en palabras se encuentra en seis, lo que indica que al remover “stop words” en estas bases de datos se reducen alrededor de seis palabras por campo que al escalarlo en la cantidad de palabras eliminadas en todas las bases de datos significa un

ahorro en el almacenamiento y procesamiento significativo.

Gráficamente se puede observar la diferencia de palabras para 100 de los requerimientos entre las descripciones para OD y ODL, los resultados se encuentran en la Figura 25:

Figura 25. Diferencia en conteo de palabras para OD y ODL

Elaboración propia

La Tabla 8 muestra algunos ejemplos textuales sobre cómo se trabajan los datos al hacer la limpieza de “stop words”:

Tabla 8. Ejemplo de conteo de palabras para OD y ODL

Descripción con “stop words” (OD)	Conteo de palabras	Descripción sin “stop words” (ODL)	Conteo de palabras
Para que sirve el campo que aparece en el reporte	10	Para sirve campo aparece reporte	5
Trabajar sobre el cubo de datos para consolidación de los datos y permitir consultas eficientes sobre el sistema.	18	Trabajar cubo datos consolidación datos permitir consultas eficientes sistema	9
subir la aplicación versión 3.2 al play store de codensa	10	subir aplicación versión 3.2 play store codensa	7

Se reporta error al momento de realizar filtros en formulario desde el usuario de coordinador	15	Se reporta error momento realizar filtros formulario usuario coordinador	9
el usuario está tratando de ingresa a la plataforma, pero no inicia sesión	13	usuario tratando ingresa plataforma no inicia sesión	7

Elaboración propia

Finalmente, se procede a relacionar el campo de descripción con las categorías. Inicialmente se analizan las palabras más comunes y utilizadas por cada una de las categorías. La Tabla 9 muestra los unigramas o palabras claves encontrados por categoría.

Tabla 9. Unigramas/palabras claves por categoría.

Categoría	Palabras Claves (unigramas)
Cambiar datos por BD	actualizaciones, permita, microrutas, flecheo, cambio, datos, actualizar, bases, BD
Capacitación de módulo o funcionalidad	módulo, capacitar, plan, capacitación
Carga de datos masiva	masivo, masiva, hrec, ciclos, cargue
Configuración de nueva capa en visor	visor, configurar, nuevas, capa, nueva
Configuración de nuevo widget en visor	visor, localización, configuración, coordenadas, cartográfico, implementar, view, nuevo, widget
Configuración de equipos (GPS)	instalación, servidor, soluciones, cellocator, GPS, configuración
Configuración de nuevo Reporte	especificadas, proyecta, reporte, nuevo
Configuración de nuevo campo	configuración, nuevo, campo
Creación de usuarios para ingreso plataforma	usuario, gestor, creación, plataforma
Datos erróneos en BI	incorrecto, información, indicador, inconsistencia, fallos, BI
Datos erróneos en Cubo	entregables, duplicados, inconsistente, auditoría, cubo

Datos erróneos en Visor	visor, registra, datos, errores, digita, erróneos, malos
Datos erróneos en reporte	error, generado, datos, erróneos, inconsistencias, presentadas, reporte
Disminución del desempeño de plataforma	módulos, presentando, pegada, lento, disminución, rendimiento, lentitud, plataforma, desempeño
Duda en uso de campos	llenarlo, módulo, primario, explicar, sirve, refiere, entiendo, campo
Duda en uso de plataforma	inquietudes, inducción, duda, cómo, uso, plataforma
Formulario no guarda, no edita y o no elimina	edita, formulario, anexar, editarlos, cambios, tampoco, permite, elimina, editar, guardar,
Generación de shape	generar, proyección, shapes, shp, shape
Generar reporte, informe, datos solicitados por el cliente	entregada, solicitada, cliente, generar, entrega, mes, soporte, informe
Implantación módulo o nueva funcionalidad	módulos, administración, funcionalidades, funcionalidad, implantar, implantación
Imposibilidad ingreso de un usuario	puede, sesión, loguear, credenciales, imposibilidad, iniciar, olvidó, ingresar, usuario
Interrupción del servicio/No carga el sistema	incidente, entrar, pagina, cayó, fuera, sistema, caído, servicio, interrupción, completa
Lentitud en el visor	bloqueo, demorando, lentos, bloqueos, visores, lenta, lentitud
No busca ni filtra en formulario	ajustar, administrar, busca, formulario, buscar, filtrando, búsqueda, filtra, filtrar
No carga el visor	ninguna, capa, prenden, visualizando, enrutamiento, cargando, carga, visor, capas
No carga un formulario	carga, cargan, formularios, formulario
No descarga reporte	reportador, descargado, genera, intentar, falla, error, reportes, exportar, reporte, descargar
No envía el backup de aplicación móvil	visualizar, envía, aplicación, wetransfer, sincronizando, sincroniza, backup
No funciona adecuadamente widget del visor	funcionaron, alarmas, visor, selección, funcionamiento, editor, fallo, físicos, funciona, widget
No inicia sesión un usuario en movilidad	restableciendo, colaboran, usuario, restablecer, iniciando, clave, acceder, movilidad, iniciar, sesión
No transmite GPS	reporta, transmitir, vehículo, transmisión, transmitiendo, transmite

Publicación de servicios, capas	servicio, habilitar, configurada, reiniciar, publicación, publicar, servicios, geoserver
Revisión de Consultas	migración, revisión, msla, consulta, consultas
Revisión de GPS	configurar, validación, transmisión, revisión, gps
Saltos de GPS (Descalibrado)	ciudad, encuentra, presentan, vehículos, saltos, descalibrado, descalibrados
Subir aplicación móvil (app) a play store	aplicación, descarguen, apk, versión, actualización, móvil, publicar, supervisión, aplicación, subir, app, play, store

Elaboración propia

Para finalizar se hace el análisis de palabras más usadas por categoría. Se representa la nube de palabras de algunas de ellas. La Figura 26 muestra en la nube de palabras, las palabras más comunes ingresadas por los clientes de SIGMA Ingeniería S.A., que representan el requerimiento “Saltos de GPS (Descalibrado)” con palabras como: Vehículo, descalibrado y saltos.

Figura 26. Nube de palabras para categoría: Saltos de GPS (Descalibrado)

Elaboración propia

La Figura 27 muestra las palabras más usadas que representan el requerimiento “Lentitud en el visor” con palabras como: Visor, lentitud, capas y revisar.

Figura 27. Nube de palabras para categoría: Lentitud en visor

Elaboración propia

La Figura 28 muestra las palabras más usadas que representan el requerimiento “Disminución del desempeño de plataforma” con palabras como: Plataforma, lentitud, disminución y desempeño.

Figura 28. Nube de palabras para categoría: Disminucion del desempeño de plataforma

Elaboración propia

Posterior al análisis de los datos se procedió a la evaluación de las técnicas de ML seleccionadas como se observa en la siguiente sección.

8.2 RESULTADO OBJETIVO 2

Evaluar el desempeño de técnicas de aprendizaje de máquina supervisadas más utilizadas (SVM, ET, RF, LR, DT, LDA, NB, KNN) aplicadas al procesamiento de texto a partir de modelos de referencia validados basados en NLP.

Los resultados se muestran para cuatro experimentos diseñados; estas pruebas consisten en diferentes combinaciones en cuanto a preprocesamiento, balanceo y algoritmos de ML con optimización de hiperparámetros (usando GridSearchCV de scikit-learn) [97] [98], [99]. De este modo, se definieron cuatro experimentos: el conjunto de datos original (OD), un conjunto de datos con preprocesamiento (DP), un conjunto de datos con preprocesamiento y balanceo (DPB), y finalmente, un conjunto de datos con preprocesamiento, balanceo y optimización (DPBO) (ver Tabla 10).

Tabla 10. Indicadores textuales asignados a las pruebas realizadas

Configuración del experimento	Abreviatura
Conjunto de datos original	OD
Conjunto de datos con preprocesamiento	DP
Conjunto de datos con preprocesamiento y balanceo	DPB
Conjunto de datos con preprocesamiento, balanceo y optimización	DPBO

Elaboración propia

- **OD**. el conjunto de datos se toma en bruto, se le aplica únicamente el TF-IDF para que los algoritmos de ML puedan entrenarse, sin embargo, no se le hace ningún preprocesamiento, finalmente, se aplican las técnicas de ML propuestas.

- **DP.** se aplican técnicas de preprocesamiento al conjunto de datos original, como las técnicas de NLP y la extracción de características mediante el TF-IDF *vectorizer* explicadas en la sección 5.1

- **DPB.** se realizan todas las etapas aplicadas en DP, así como métodos de balanceo de datos.

- **DPBO**. se aplican todas las etapas mencionadas en DPB. Adicionalmente, se aplica la optimización de hiperparámetros para las técnicas de ML elegidas.

Los experimentos tienen dos particiones de datos (*Hold-out*), de entrenamiento con un 80% de datos y el restante 20% para el testeo [100]. Además, los resultados tienen validación cruzada (CV) con 10-folds; esta técnica permite observar que también generaliza el algoritmo de ML a partir de su precisión media y desviación estándar.

Se realiza el proceso de *Hold-out* para cada experimento (OD, DP, DPB, DPBO) a través de las ocho técnicas de ML explicadas en la sección 5.1. Estos resultados muestran el rendimiento obtenido cuando se trabaja con cada uno de los experimentos planteados. La Tabla 11 muestra los resultados al aplicar la metodología propuesta. SVM alcanza el mejor rendimiento entre todas las técnicas evaluadas. Se muestra una diferencia de alrededor de

un 3% entre los experimentos OD y DPBO para esta técnica. Además, se confirma que realizar un ajuste de hiperparámetros es un método eficaz para aumentar el rendimiento de las técnicas ML.

Tabla 11. Comparación de los resultados obtenidos entre los ocho algoritmos ML mediante *Hold-out* en cada una de las condiciones establecidas. Las entradas en negrilla indican los tres mejores resultados para cada experimento.

Experimento	Técnica de ML	*Accuracy*	*F1-Score*	*Recall*	*Precision*	*Training Time [s]*	*Prediction Time [s]*
OD	**SVM**	**87.44**	87.38	87.44	90.32	0.3261	0.05
	ET	**89.17**	87.73	87.91	89.24	0.4691	0.019
	RF	86.05	85.9	86.05	87.12	0.4061	0.017
	LR	**86.74**	86.49	86.74	89.29	0.046	0.000
	DT	76.05	76.21	76.05	78.84	0.042	0.001
	LDA	81.63	81.22	81.63	83.07	36.62	0.003
	NB	69.07	68.63	69.07	71.36	0.013	0.066
	KNN	73.49	73.17	73.49	77.58	0.001	0.017
DP	**SVM**	**89.07**	89.11	89.07	91.34	0.3521	0.052
	ET	**87.67**	87.43	87.67	88.4	0.4641	0.019
	RF	84.65	84.46	84.65	85.6	0.4001	0.017
	LR	**88.60**	88.25	88.6	89.9	0.052	0.000
	DT	80.00	79.77	80.00	81.85	0.042	0.000
	LDA	81.86	82.02	81.86	84.31	0.3141	0.002
	NB	66.05	65.19	66.05	68.27	0.013	0.066
	KNN	73.72	73.09	73.72	75.00	0.002	0.017
DPB	**SVM**	**88.84**	88.92	88.84	90.98	0.7022	0.066
	ET	**87.91**	87.72	87.91	89.09	76.12	0.02
	RF	85.81	85.41	85.81	86.86	71.12	0.018
	LR	**89.53**	89.26	89.53	90.17	0.126	0.001
	DT	81.16	81.00	81.16	83.36	0.092	0.001
	LDA	78.84	79.04	78.84	80.82	0.7162	0.001
	NB	68.84	68.31	68.84	71.77	0.026	0.088
	KNN	77.44	77.24	77.44	79.91	0.003	0.028
DPBO	**SVM**	**90.47**	90.29	90.47	91.21	0.5371	0.081
	ET	**88.84**	88.49	88.84	89.56	0.9402	0.024

	RF	**86.51**	86.10	86.51	87.63	5.6147	0.176
	LR	86.05	85.94	86.05	86.89	0.175	0.001
	DT	78.60	77.68	78.60	79.81	0.094	0.001
	LDA	82.56	82.66	82.56	84.90	0.803	0.001
	NB	69.30	68.90	69.30	72.38	0.033	0.107
	KNN	77.91	77.01	77.91	81.75	0.002	0.025

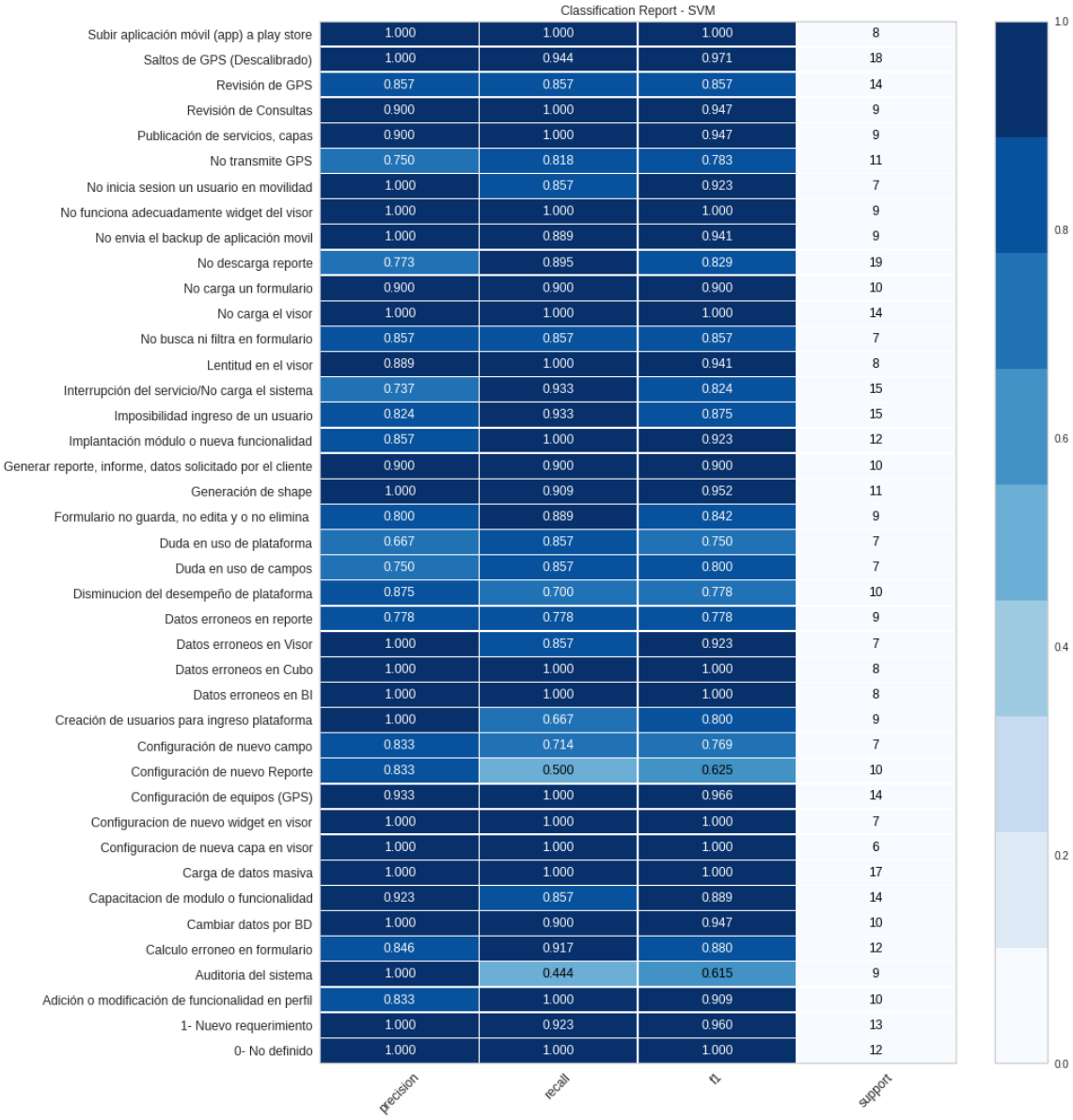
Elaboración propia

En cuanto a los resultados obtenidos mediante la técnica SVM en cada prueba, se observa que, en la mayoría de los experimentos esta técnica es la de mejor desempeño, seguido de técnicas como ET y LR. En las Figuras de la 29 a la 37 se visualizan métricas como CR, CM y curvas ROC de las 3 mejores técnicas resultantes del proceso (SVM, ET y LR), para poder analizar a profundidad los resultados obtenidos. Las métricas obtenidas para las cuatro técnicas de ML restantes junto con las 3 mostradas en este documento se encuentran en el **¡Error! No se encuentra el origen de la referencia.**. Los resultados allí mostrados son en base al experimento DPBO; experimento que brindó los mejores resultados en todas las técnicas de ML seleccionadas.

Con respecto a los resultados obtenidos para la técnica SVM se obtienen desempeños con una media de 91.21% para la Precisión, 90.47% para el *Recall*, y 90.29% para el *F1-score*. Esto implica un rendimiento equilibrado en calidad y cantidad del modelo SVM para clasificar las solicitudes en cada categoría a pesar de ser un problema inicialmente desbalanceado.

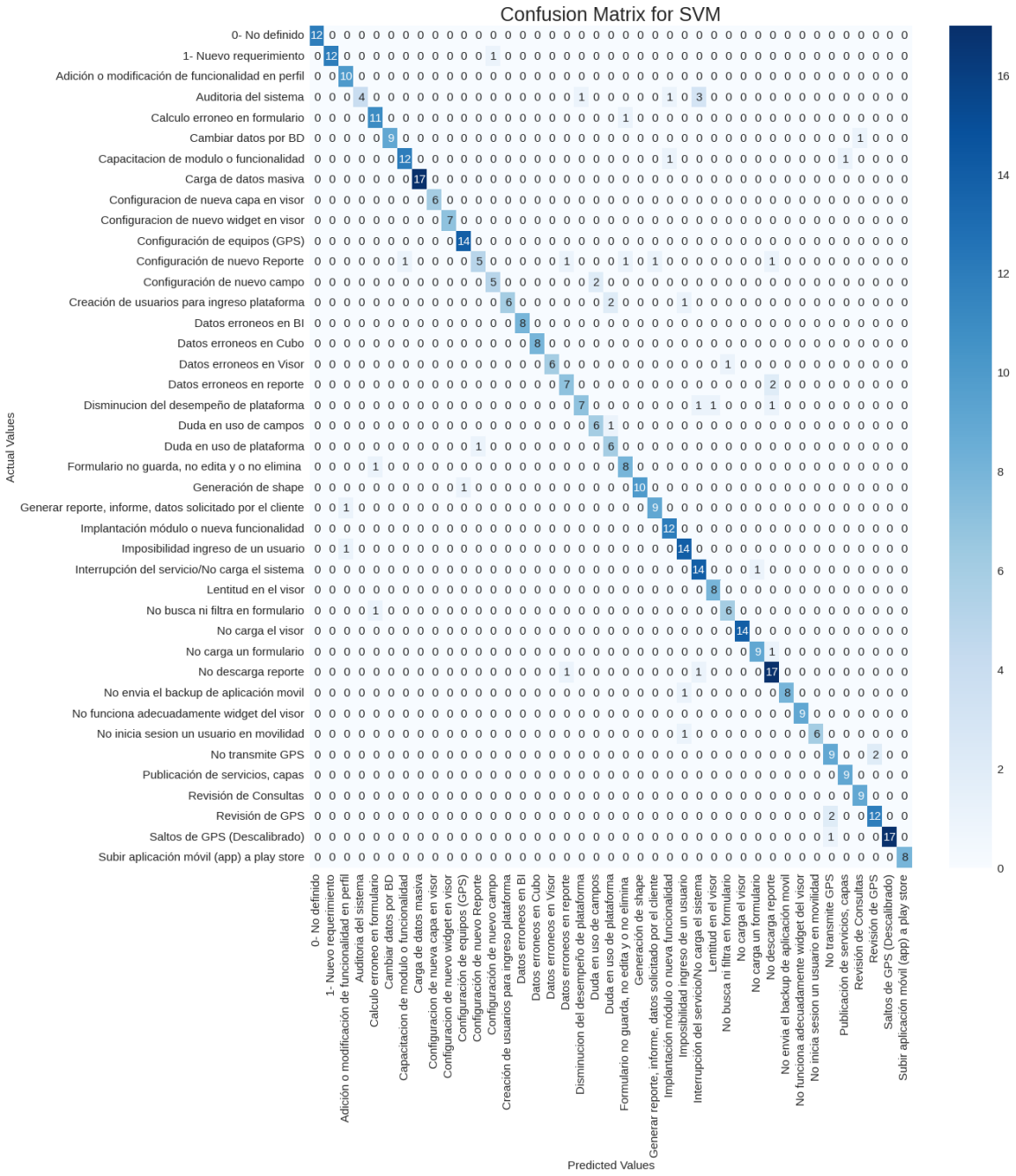
Estos resultados están precedidos de técnicas como ET donde se obtienen desempeños con una media de 89.56% para la Precisión, 88.84% para el *Recall*, y 88.49% para el *F1-score.* Finalmente, la técnica LR con una media de 86.89% para la Precisión, 86.05% para el *Recall*, y 85.94% para el *F1-score.*

Figura 29. Reporte de clasificación para el experimento DPBO en SVM.



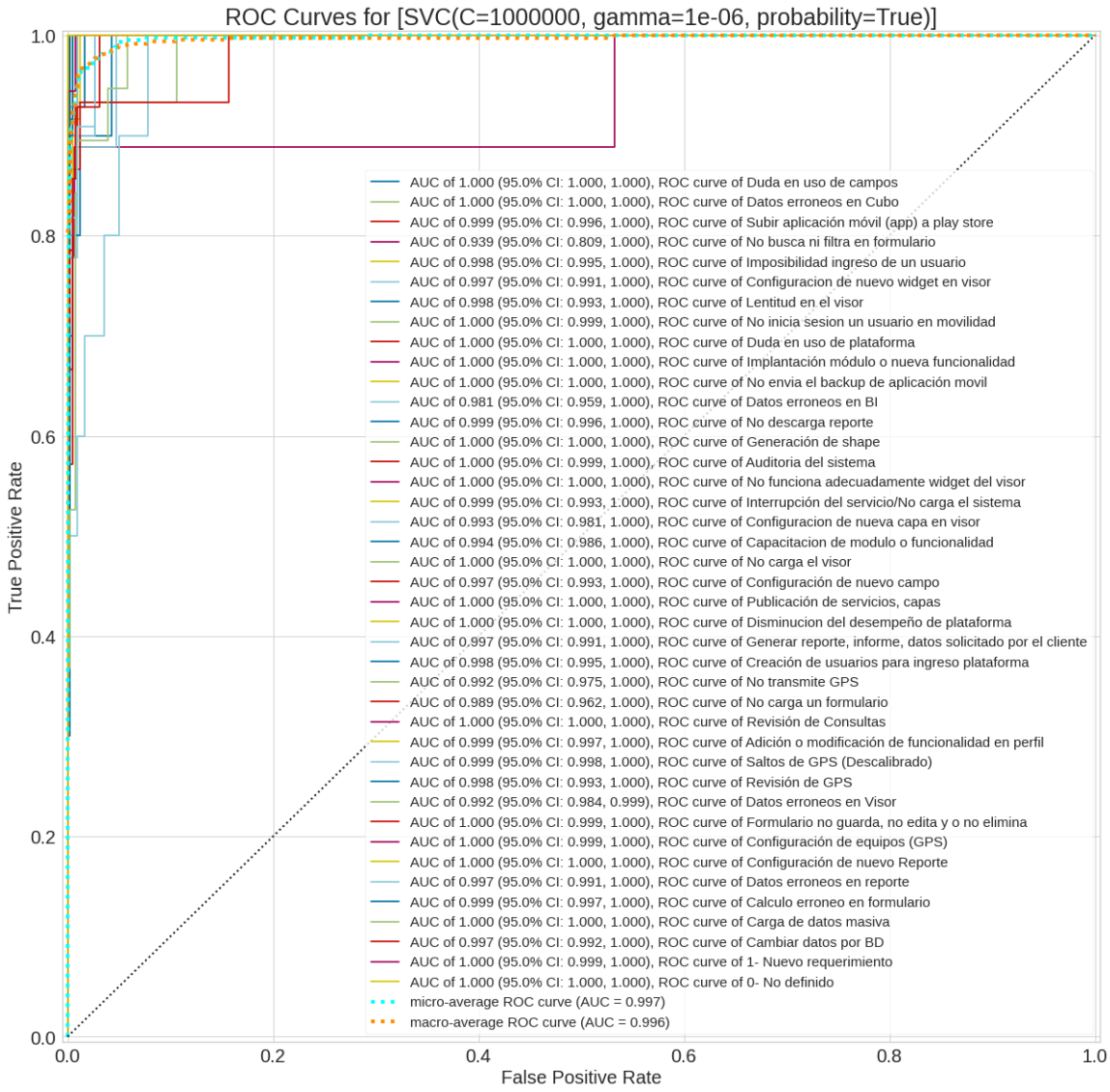
Elaboración propia

Figura 30. Matriz de confusión para el experimento DPBO en SVM.



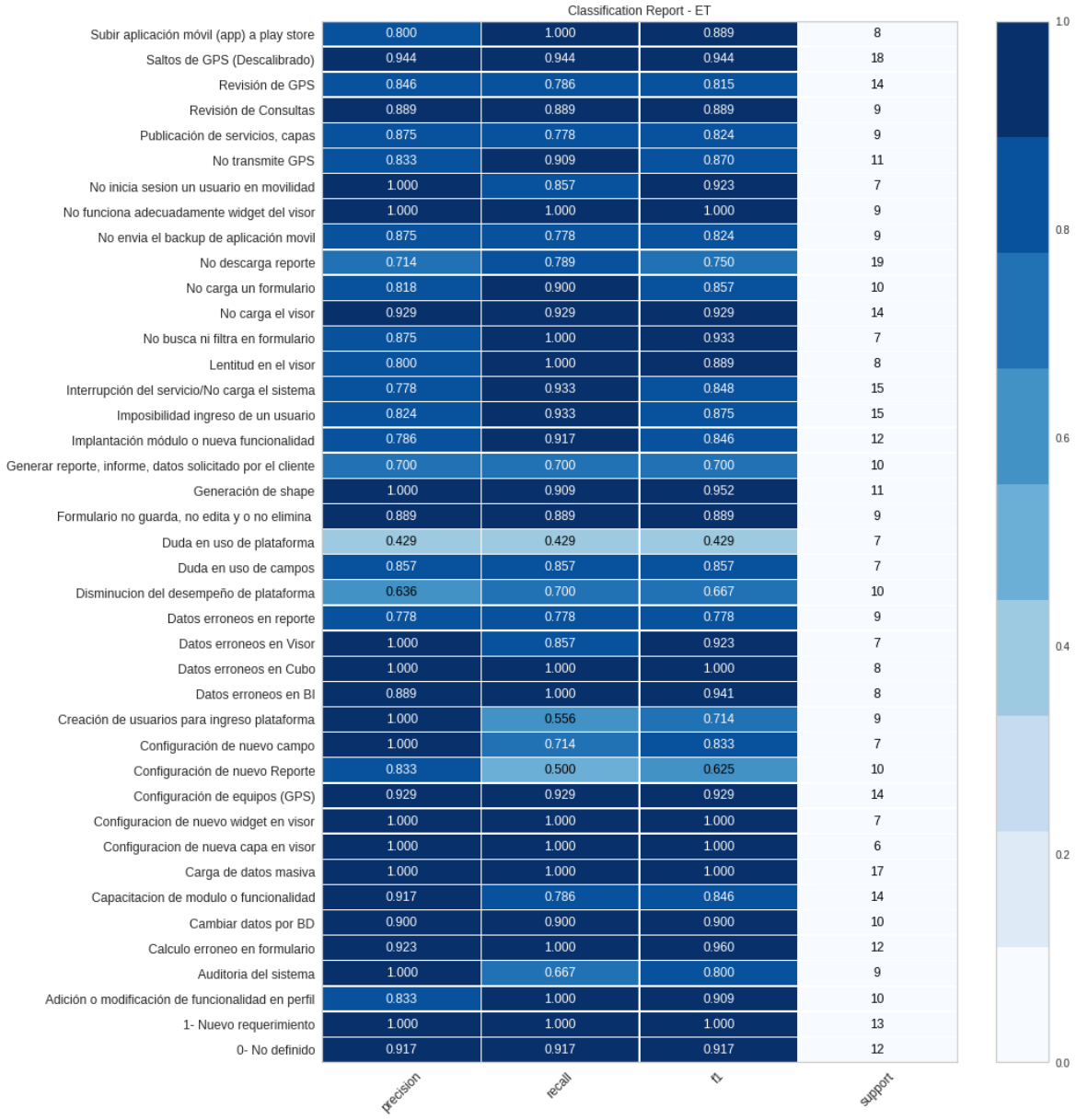
Elaboración propia

Figura 31. Curva ROC para el experimento DPBO en SVM.



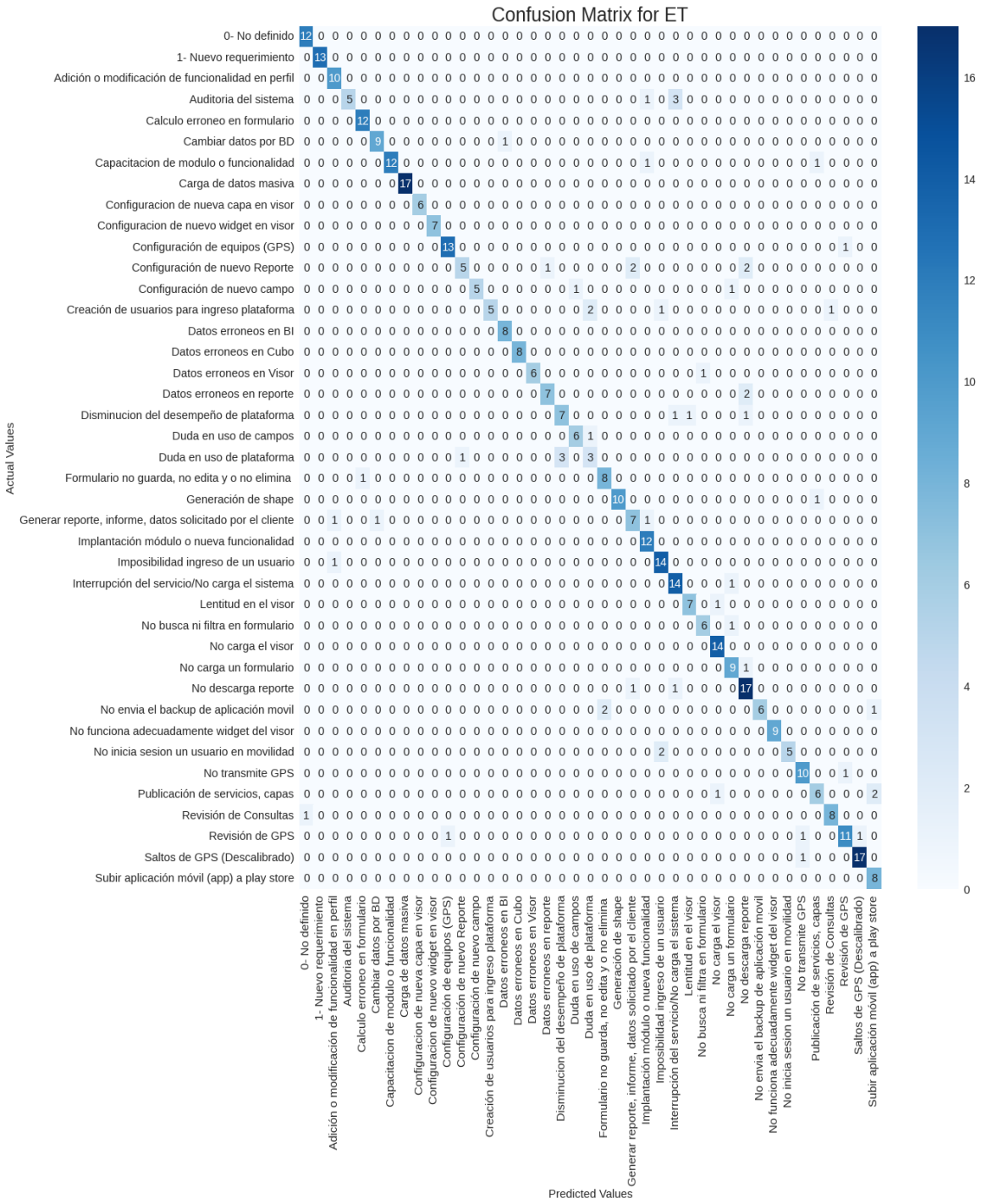
Elaboración propia

Figura 32. Reporte de clasificación para el experimento DPBO en ET.



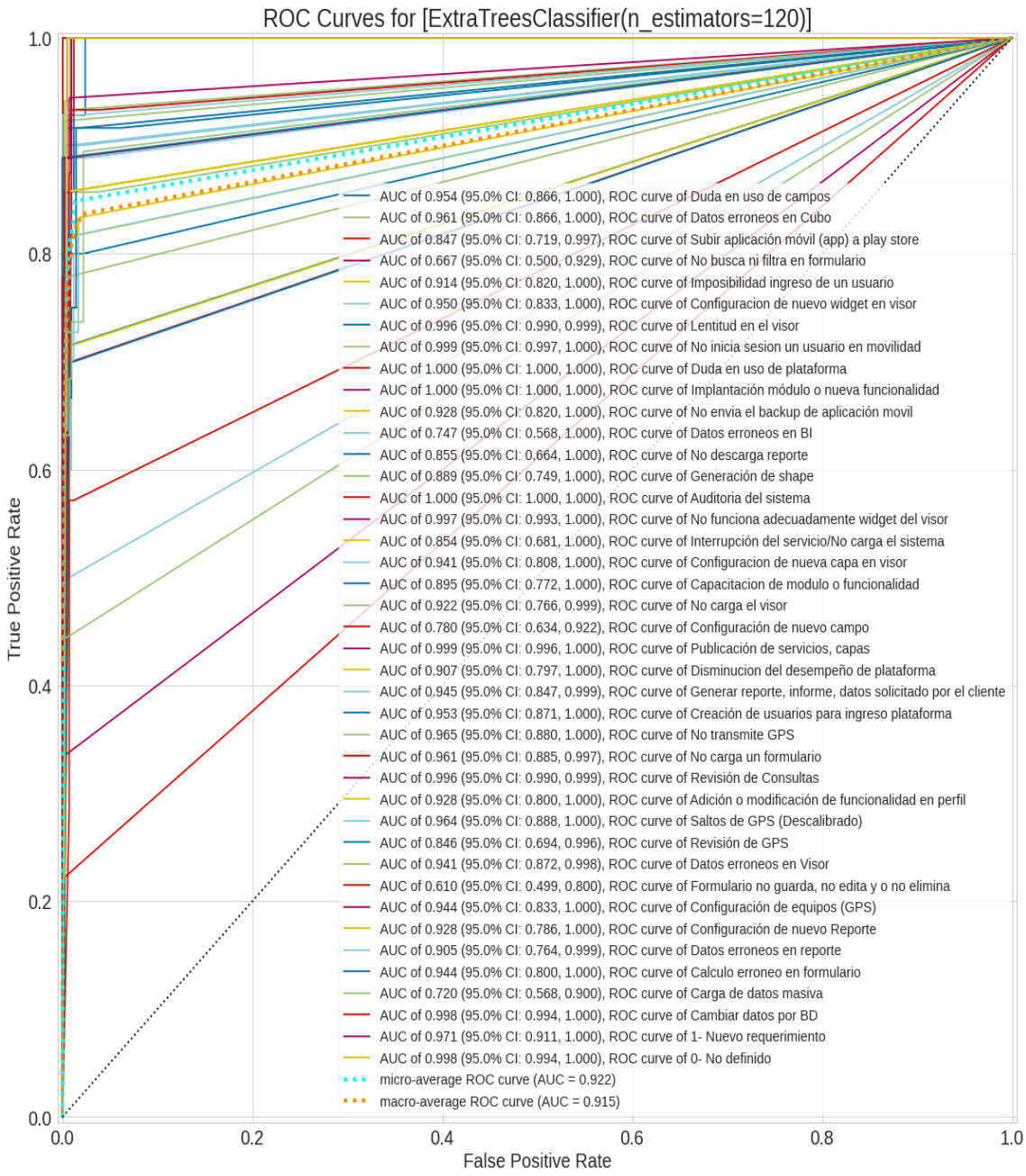
Elaboración propia

Figura 33. Matriz de confusión para el experimento DPBO en ET.



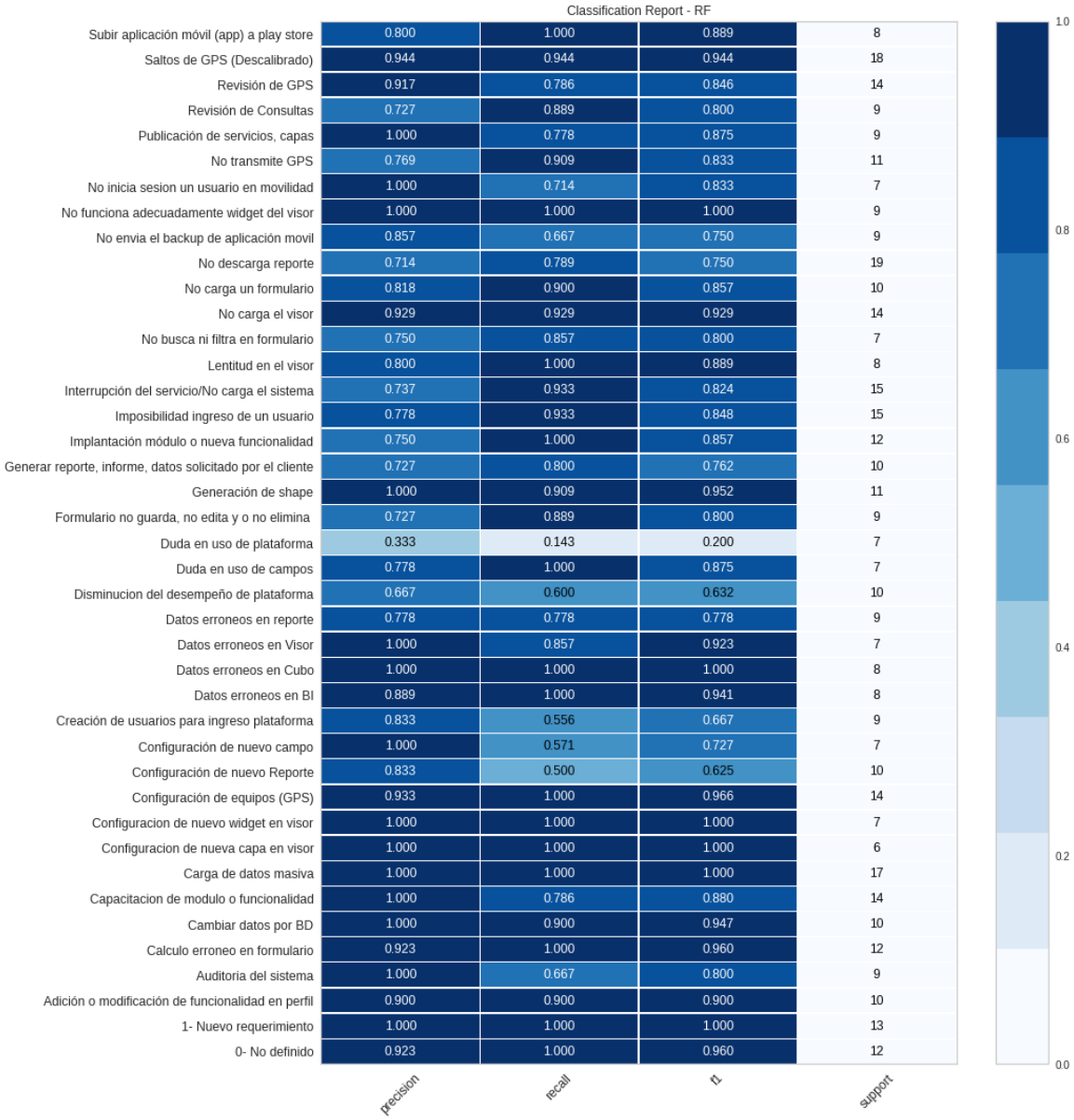
Elaboración propia

Figura 34. Curva ROC para el experimento DPBO en ET.



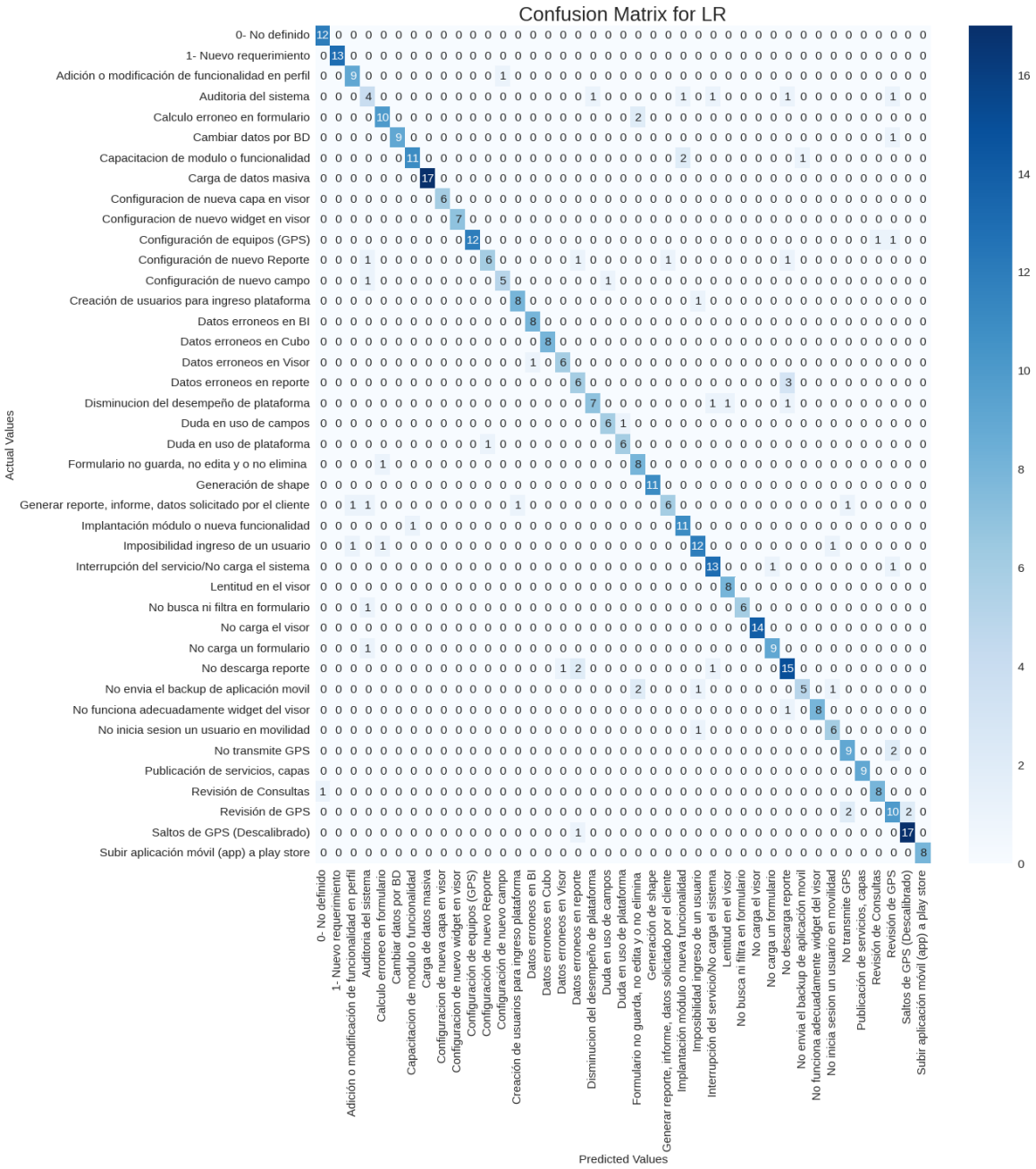
Elaboración propia

Figura 35. Reporte de clasificación para el experimento DPBO en LR.



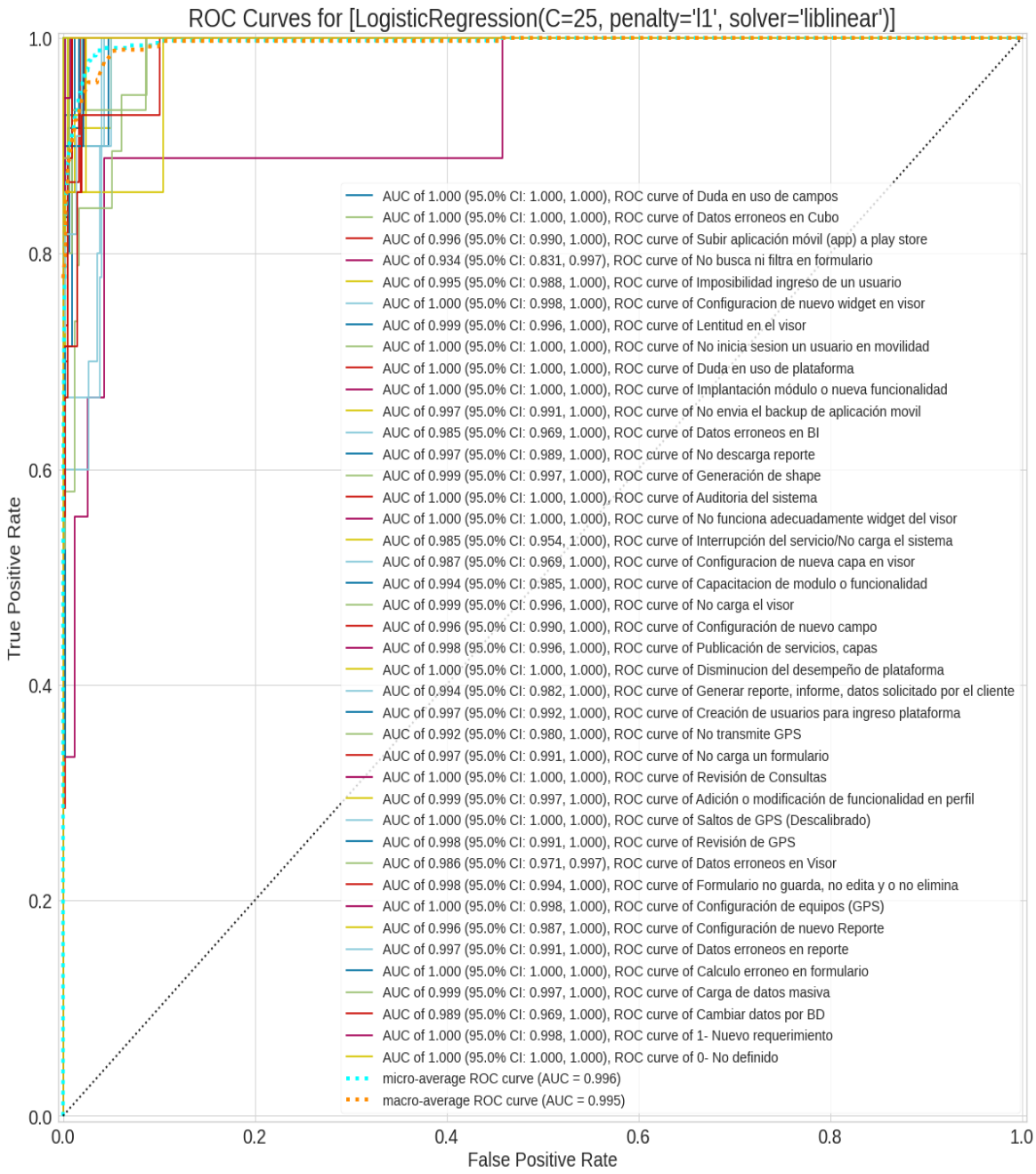
Elaboración propia

Figura 36. Matriz de confusión para el experimento DPBO en LR.



Elaboración propia

Figura 37. Curva ROC para el experimento DPBO en LR.



Elaboración propia

Adicionalmente, se presentan los resultados obtenidos mediante CV con 10-folds, este proceso también se lleva a cabo para cada experimento (OD, DP, DPB, DPBO), los resultados se muestran en la Tabla 12 con la precisión obtenida y la desviación estándar para cada experimento.

Tabla 12. Comparación de los resultados obtenidos mediante CV entre los ocho algoritmos ML en cada una de las condiciones establecidas. Las entradas en negrilla indican los resultados de los tres mejores algoritmos para cada experimento.

Experimento	Técnica de ML	Validación Cruzada [%]	CV Time [s]
OD	**SVM**	**87.00±2.25**	1.558
	ET	**87.14±02.28**	1.182
	RF	85.33±02.44	0.708
	LR	**86.54±02.09**	0.127
	DT	77.82±02.6	0.095
	LDA	82.25±1.223	3.288
	NB	69.01±3.48	0.725
	KNN	74.75±1.83	0.059
DP	**SVM**	**88.21±1.82**	0.629
	ET	**87.79±1.66**	0.825
	RF	86.16±1.46	0.700
	LR	**87.75+1.63**	0.141
	DT	78.75±1.93	0.092
	LDA	83.04±1.9	3.380
	NB	67.20±2.77	0.526
	KNN	76.75±2.76	0.062
DPB	**SVM**	**96.92±3.39**	1.178
	ET	**95.32±3.94**	1.228
	RF	**93.97±4.48**	1.106
	LR	93.32±2.58	0.264
	DT	86.86±5.72	0.175
	LDA	93.01±6.43	7.078
	NB	88.66±8.52	1.113
	KNN	86.35±5.17	0.052
DPBO	**SVM**	**96.11±3.2**	0.8602
	ET	**95.43±4.02**	1.442
	RF	94.42±4.51	8.364
	LR	**95.07±4.37**	0.3731
	DT	84.49±3.83	0.191
	LDA	94.28±5.31	8.614
	NB	88.97±8.40	1.170
	KNN	91.51±7.06	0.207

Elaboración propia

Nuevamente es la técnica SVM la que cuenta con un desempeño mayor mediante CV, esto indica la alta generalización del modelo y seguramente un correcto funcionamiento en clasificación que tendrá aplicado al momento de ser usado en producción.

A partir de estos resultados, la técnica SVM es seleccionada para ser implementada en el modelo de clasificación de incidencias para Timework.

El acceso al código fuente y al proceso realizado para las ocho técnicas seleccionadas en cada uno de los experimentos se pueden encontrar en los repositorios de Github y Zenodo:

- **Github:** https://github.com/BioAITeam/Modelo-de-clasificacion-de-incidencias-mediante-ML-y-NLP
- **Zenodo**: https://zenodo.org/badge/latestdoi/501489594

8.3 RESULTADO OBJETIVO 3

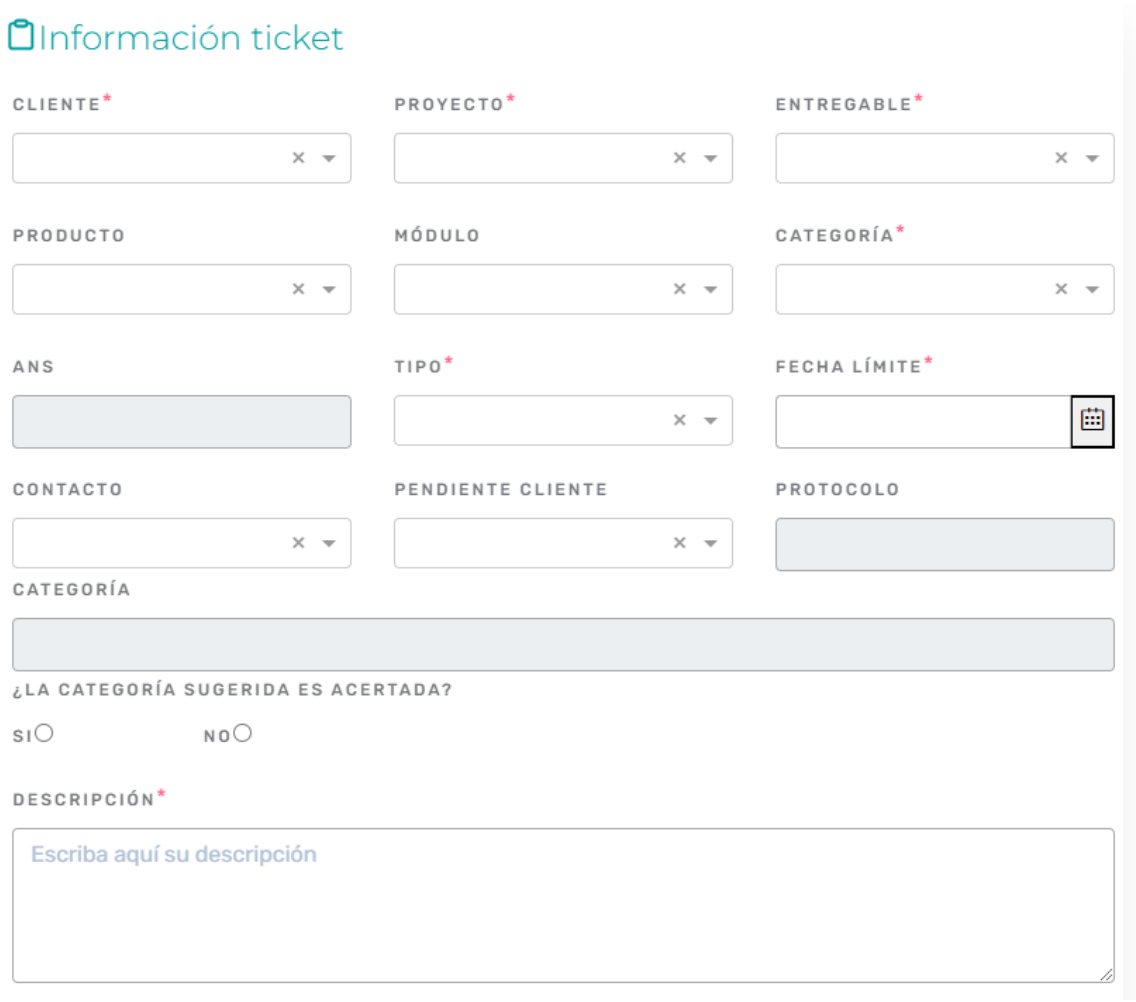
Implementar un modelo computacional de ML para la clasificación de los tickets de servicio brindadas por Timework y la entrega de protocolos de solución basados en la técnica de aprendizaje de máquina resultante del objetivo específico 2 y NLP.

El código para el desarrollo del modelo de clasificación fue desarrollado en Python, el **¡Error! No se encuentra el origen de la referencia.**; muestra el paso a paso de la instalación del modelo para llegar a la predicción final dada por la técnica seleccionada junto con su protocolo.

El **¡Error! No se encuentra el origen de la referencia.**; contienen las instrucciones de uso del módulo de NLP una vez implantado en la plataforma Timework de SIGMA Ingeniería S.A.

El módulo de clasificación de requerimientos y sus campos se muestra en la Figura 38.

Figura 38. Módulo NLP en plataforma Timework.



Tomado de plataforma Timework

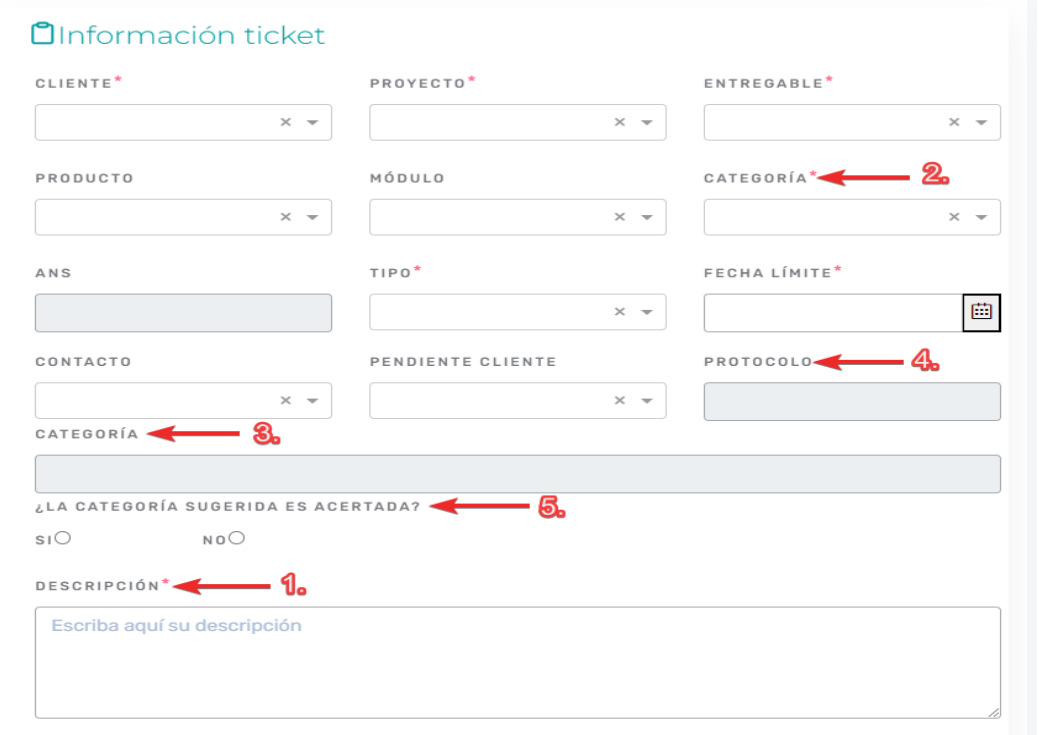
Cada uno de los campos mostrados en la Figura 38 deben ser completados por el área de servicio al cliente de la empresa, una vez ingresado el campo de descripción tanto la categoría como el protocolo son mostrados automáticamente.

8.4 RESULTADO OBJETIVO 4

Validar el funcionamiento y desempeño del modelo computacional mediante análisis estadísticos y juicio de expertos de la organización.

Una vez implantado el módulo en la plataforma Timework, el proceso se realizó con las siguientes condiciones. Todos los datos solicitados en el módulo de registro de Tickets de servicio deben ser completados incluyendo el campo 1, 2 y 5 (ver Figura 39).

Figura 39. Prueba piloto del modelo computacional implantado en Timework.



Adaptado de plataforma Timework

En el campo 1. DESCRIPCIÓN se ingresa el contenido del ticket o requerimiento entregado por el cliente, posterior a esto, en el campo 2. CATEGORÍA se ingresa la categoría asignada por el experto de la organización del área de soporte técnico de la empresa que corresponda con el ticket de servicio indicado, en este punto y después de haber llenado todos los campos se visualiza la predicción o categoría sugerida entregada por el modelo computacional en el campo 3. CATEGORÍA, junto con el protocolo asignado al requerimiento ingresado en el campo 4. PROTOCOLO. Finalmente, se debe marcar en el campo 5. ¿LA CATEGORÍA SUGERIDA ES ACERTADA? “SI”, si la predicción hecha por el modelo computacional corresponde con la categoría asignada por el

experto, de lo contario marcar “NO” si la predicción hecha por el modelo no corresponde con la categoría asignada por el experto.

Posterior a este paso se crea y se guarda el ticket de servicio, este queda almacenado en las bases de datos con todos los campos encontrados en la Tabla 13, estos datos permiten visualizar las predicciones tanto correctas como incorrectas. En el caso de ser incorrectas es posible dirigirse al campo de la categoría entregada por los expertos de la organización para saber cuál era la categoría que el modelo debió haber predicho y a partir de las etiquetas asignadas por los expertos se va realizar procesos de reentrenamiento para mejorar la generalización de los modelos.

Tabla 13. Ejemplo de datos de la prueba piloto entregados por Timework

tik_descripcion	nombre_categoria	categoria_sugerida	categoria_sugerida_correcta
No funciona el sistema	No carga el sistema	No carga el sistema	SI
Por favor poner per_codigo OOCQ-00318-12 en la RES-014472	Cambiar datos por BD	Cambiar datos por BD	SI
El sistema de Seguimiento Personas Barrido de Serviciudad no carga.	No carga el sistema	No carga el sistema	SI
No carga información en el Sistema georreferenciación Porsche	No funciona adecuadamente widget del visor	No funciona adecuadamente widget del visor	SI
Implantación función jsonciclorelleno para todos los clientes de Geoaseo	Implantación módulo o nueva funcionalidad	Implantación módulo o nueva funcionalidad	SI
Informe CARDER V	Realizar informes del área	0- No definido	NO
Buenos días, reporto el vehículo 21701 de Aseo urbano que dejó de transmitir Gps.	No transmite GPS	No transmite GPS	SI

Implantación Excesos de Velocidad para todos los clientes de Geoaseo.	Implantación módulo o nueva funcionalidad	Implantación módulo o nueva funcionalidad	SI

Elaboración propia

Después de tener los campos a analizar definidos, se procede a realizar la prueba piloto del modelo computacional implantado en Timework durante un tiempo de 6 meses que van desde noviembre 20 del 2021 a mayo 20 del 2022. Los campos entregados por Timework (ver Tabla 13) se muestran a continuación:

1. **Descripición del ticket (tik_descripcion):** descripción del ticket de servicio y requerimiento brindado por el cliente.
2. **Nombre de la categoría (nombre_categoria):** categoría asignada por los expertos de la organización
3. **Categoría Sugerida (categoria_sugerida):** categoría predicha o sugerida por el modelo computacional.
4. **Categoría sugerida acertada (categoria_sugerida_correcta)**: campo de decisión de acierto o desacierto entre la categoría ingresada por el área de soporte técnico y la predicción del modelo computacional.

Con base a los datos obtenidos durante el periodo de tiempo de la prueba piloto se procede a analizar los resultados obtenidos a partir del campo “categoria_sugerida_correcta”:

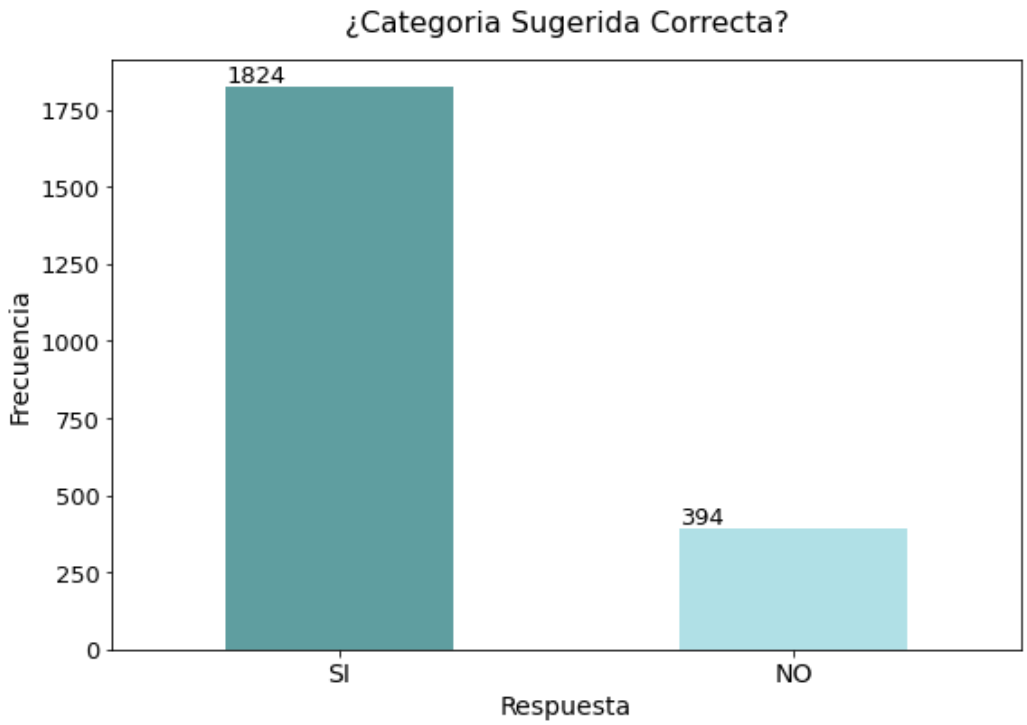
El estudio de los resultados se hizo con Python 3.8 por medio de EDA (Análisis exploratorio de los Datos) que faciliten el análisis estadístico de bases de datos, los resultados se muestran a continuación:

Conteo de tickets registrados durante la prueba piloto

- **Conteo total de tickets:** 2218
- **Conteo total de “SI”:** 1824
- **Conteo total de “NO”:** 394

Los resultados obtenidos a partir de los aciertos y desaciertos encontrados durante la prueba piloto se muestran en el gráfico de barras de la Figura 40:

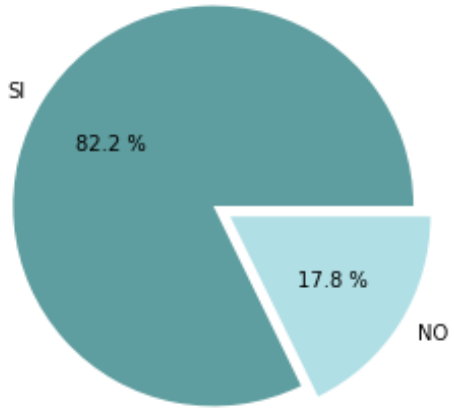
Figura 40. Tabla de frecuencias de aciertos y desaciertos de la prueba piloto.



Elaboración propia

Donde cada una representa los porcentajes mostrados en la Figura 41.

Figura 41. Porcentajes de aciertos y desaciertos de la prueba piloto en Timework.



Elaboración propia

Lo que indica que durante la prueba piloto los errores en predicción por parte del modelo representan un 17.8% de las predicciones totales y el restante 82.2% representa los aciertos en las predicciones del modelo.

Al comparar el desempeño obtenido por medio del modelo durante el entrenamiento y el resultado obtenido durante la prueba piloto se muestra en la Tabla 14:

Tabla 14. Comparación de desempeño del modelo computacional implantado en Timework VS la prueba piloto.

Desempeño del modelo SVM implantado en Timework	**Desempeño del modelo implantado en Timework durante prueba piloto**
90.47%	82.2 %

Elaboración propia

La diferencia en resultados entre ambas pruebas es del 8.27% que están representados en los siguientes puntos:

El 35.53% de los errores ocurren debido a que, en el transcurso de los 6 meses de la prueba piloto, alrededor de 15 categorías fueron creadas en la empresa que han sido adicionadas a las 41 categorías existentes en el momento del entrenamiento e implantación del modelo. Lo que implica que el modelo implementado en Timework no está entrenado para predecir dichas categorías, lo cual hace que el modelo tenga un desacierto en la predicción. Para resolver el problema se debe reentrenar el sistema con las nuevas categorías generadas.

El restante 64.46% se encuentra representado en categorías como 1- Nuevo requerimiento, 0- No definido y Cambiar datos por BD. Para mitigar el aumento de errores en categorías como estas se ha recomendado implantar ciertas condiciones al momento de escribir tickets, usando palabras claves y reentrenando el modelo computacional ya implantado.

9 DISCUSIÓN

El diseño del sistema de clasificación consta de seis pasos: limpieza de datos, aplicación de técnicas de NLP, extracción de características, partición de conjuntos de datos, balanceo y evaluación de métricas.

En primer lugar, la limpieza del conjunto de datos, la aplicación de técnicas de NLP y la extracción de características son pasos esenciales del preprocesamiento de datos cuando se trabaja con datos textuales. El balanceo de los datos es adecuado cuando el conjunto de datos está desequilibrado en grandes proporciones, de lo contrario alteraría la partición de los datos entre el entrenamiento y la prueba y no proporcionaría resultados muy fiables. Es esencial encontrar tamaños de partición de datos adecuados que permitan al modelo generalizar bien en el proceso de entrenamiento y demostrar fiabilidad cuando se pruebe. El conjunto de entrenamiento representa el 80% del total de las muestras, y el conjunto de prueba el 20% restante.

En cuanto a la Tabla 11, el aumento de las puntuaciones de DPB y DPBO en relación con DO muestra que las técnicas de balanceo proporcionan mejores resultados de precisión debido a que hay una diferencia de 64 unidades entre la frecuencia de la categoría más alta y la más baja que han sido compensados en experimentos como DBP y DPBO [101]. En cuanto a la optimización de los parámetros, esta aumenta para SVM y ET con 90.47% y 88.84% respectivamente, obteniendo SVM como la técnica de mayor rendimiento para problemas como la clasificación de textos como se muestra en la Tabla 11 y la Tabla 112 [102].

Las Figuras 29, 30 y 31 muestran resultados prometedores para todas las categorías, sin embargo, las categorías con desempeños más bajos son “Auditoría del sistema", “Configuración de nuevo reporte” y “Duda en uso de la plataforma”. Estas clases mencionadas representan aproximadamente un 29.54% de los errores totales en las predicciones. Por lo tanto, las predicciones erróneas representan 10.23% de todas las predicciones realizadas (44 de errores en 430 predicciones). Según el equipo de soporte técnico, algunas de estas categorías son muy generales y pueden abarcar temas similares lo

que las vuelve confusas entre ellas. Por lo tanto, se recomienda a SIGMA Ingeniería S.A reemplazar o eliminar categorías que no están directamente relacionadas con un requerimiento o solicitud específica y hacer uso de las palabras claves indicadas en la Tabla 9 para el registro de nuevas incidencias.

Por el contrario, la mayoría de las categorías tienen un alto número de predicciones correctas. Además, según el área de soporte técnico, estas categorías contienen palabras clave adecuadas para las predicciones basadas en el entrenamiento previo. Así, el 92.68% de todas las clases evaluadas tienen una precisión media de 92.36% en la predicción realizada.

Las precisiones y desviaciones estándar de los experimentos realizados se muestran en la Tabla 12, a través de este método, la técnica SVM vuelve a tener el mejor rendimiento. El cambio en el rendimiento de las técnicas mediante la aplicación de la optimización de los parámetros y el equilibrio de los datos es pequeño, siendo la opción “DPBO” 1.63 % mejor que la opción “DPB” [103].

Los resultados obtenidos durante la prueba piloto entregan los resultados un poco menores a los esperados, esto se debe a categorías que han sido adicionadas en Timework durante la prueba piloto y que no se habían agregado y entrenado previamente, por lo que el modelo no logra predecirlas e impacta directamente el resultado obtenido. A partir de estos resultados, se logra evidenciar donde se concentra el mayor número de errores para permitir trabajar en ellos y mejorar el desempeño del modelo computacional a partir de un reentrenamiento y uso de palabras adecuadas al ingresar nuevos tickets de servicio.

10 CONCLUSIONES

Se aplicaron ocho algoritmos tradicionales de ML para automatizar la clasificación de categorías de los requerimientos presentados por los clientes en SIGMA Ingeniería S.A, que actualmente se gestionan de forma manual. La empresa cuenta con 41 categorías que representan un aspecto de calidad individual en el área de soporte técnico. Con este trabajo se logró clasificar automáticamente los tickets del área de soporte, mejorando inmediatamente el tiempo de atención y la solución a los problemas de los clientes. Además, como el conjunto de datos se enfrentaba a un problema de desbalanceo de clases, se evaluó usando métricas como la exactitud, precisión, *recall*, *F1-score*, reporte de clasificación, curva ROC y la CM para evaluar el rendimiento de cada uno de los modelos de clasificación seleccionados. Los resultados obtenidos sugieren que el modelo SVM es el que alcanza el mayor rendimiento con una exactitud de 90.47%, una precisión de 91.21%, un *Recall* de 90.47% y un *F1-Score* de 90.29% aplicando previamente las técnicas del conjunto de datos de NLP, preprocesamiento de datos, balanceo y optimización de hiperparámetros siendo este último el más relevante para alcanzar la precisión mencionada.

Este modelo computacional permite trasladar la categoría predicha al área de soporte técnico, facilitando además el protocolo de solución a realizar para dar respuesta al usuario en el tiempo estimado por el equipo. De esta manera, se incrementa la precisión en la solución del requerimiento, se reducen los tiempos de respuesta y se incrementa la satisfacción del cliente.

El conjunto de datos trabajado en esta tesis se probó con otras técnicas como GPT3 y modelos de Deep Learning mediante CNN, los resultados no fueron tan satisfactorios como las técnicas tradicionales de ML (SVM). Adicionalmente, algunos métodos como: *Bag of Words* con TF-IDF *vectorizer* y *Count vectorizer*, *word Embedding* con *Word2Vec* y, los modelos de lenguaje natural de última generación como BERT fueron previamente probados con el conjunto de datos, los mejores resultados se obtuvieron con las técnicas TF-IDF debido a la reducción de tiempo computacional, la reducción de problemas de dimensionalidad y la simplicidad de implementación del mismo. En la prueba realizada con

el método *Bag of Words*, TF-IDF es mejor para este trabajo porque *Count Vectorizer* sólo se centra en la frecuencia de las palabras que se presentan en el conjunto de datos, en cambio TF-IDF además de proporcionar la frecuencia, también brinda la importancia de la palabra en el texto lo cual es muy útil para este estudio. Estos resultados fueron reportados en el artículo aceptado para publicación en la revista *PeerJ computer Science*

Como trabajo futuro, se propone implementar un modelo computacional basado en NLP que a partir del requerimiento ingresado por el área de soporte técnico pueda fabricar el protocolo de solución de forma automática basado en el ticket de servicio registrado. En este sentido, el modelo proporciona la categoría a la que pertenece el requerimiento y proporciona el protocolo de solución a partir de NLP. Adicionalmente, los datos generados durante el tiempo de prueba piloto en la empresa SIGMA Ingeniería S.A pueden ser utilizados para reentrenar el modelo y mejorar el desempeño del algoritmo SVM actualmente desarrollado.

11 RECOMENDACIONES

- Realizar siempre un EDA a profundidad en aplicaciones como estas y analizar las diferentes clases y aplicar la mayor cantidad de técnicas de preprocesamiento que permitan mejorar la extracción o representación de características.

- Realizar una revisión profunda del estado del arte de la aplicación a trabajar, lo cual permite que se experimente inicialmente con los mejores algoritmos de clasificación para una aplicación en particular como por ejemplo la clasificación de texto que han usado otros investigadores como se hizo en este trabajo.

- Aplicar balanceo de clases sólo al conjunto de entrenamiento, para asegurar que las métricas obtenidas en datos de *Testing*, son sobre datos reales, y así al llevar el modelo generado a producción no haya mayor diferencia en las métricas al evitar el *overffiting* sobre los modelos.

- Al experimentar tener en cuenta múltiples métricas, dado que sólo usar la común (Accuracy), en datos desbalanceados, podría dar una impresión errada de los resultados que se están alcanzando.

- En la investigación en general, buscar llevar los proyectos a un *end-to-end*, con aplicación en el

campo, dado que esto genera un impacto empresarial de forma más directa.

- Con respecto al uso y funcionamiento apropiado del modelo computacional implantado en Timework, se recomienda hacer uso de las palabras claves y adecuadas al momento de registrar nuevos tickets de servicio, ya que al estar trabajando con un problema de clasificación multi-clase, la exactitud requerida por el modelo para la diferenciación entre categorías debe ser muy precisa y esto se puede lograr con el uso de las palabras recomendadas para cada categorías entregadas en la sección 8.1 después del previo estudio y análisis del contenido de las bases de datos existentes en SIGMA Ingeniería S.A.
- Finalmente, se recomienda realizar un reentrenamiento cada seis meses para incluir categorías nuevas registradas si es el caso y poder volver a entrenar dichas categorías a partir del estudio de palabras claves o tickets de servicio nuevos ya clasificados.

12 CONTRIBUCIONES

Se realizó el registro de software utilizando modelos de ML y técnicas de NLP aplicados a texto en Timework. Este software junto con su manual técnico y de usuario fueron registrados en la dirección Nacional de Derecho de Autor (DNDA) bajo los registros número 13-91-96 y 13-91-97 respectivamente. El código fuente se puede encontrar en los repositorios de Github y Zenodo:

- **Github:** https://github.com/BioAITeam/Modelo-de-clasificacion-de-incidencias-mediante-ML-y-NLP
- **Zenodo**: https://zenodo.org/badge/latestdoi/501489594

Durante el proceso de maestría se generó un artículo científico de los resultados de la investigación obtenidos a partir del objetivo 2 del trabajo de grado. Este artículo ha sido aceptado y se publicará en la revista *Peerj Computer Science*, la cual es una revista internacional de libre acceso con factor de impacto de 0.81 (cuartil 1) según *SCImago Journal Rank* y homologada por *publindex* como A1 en Colombia.

13 REFERENCIAS

[1] "Inteligencia artificial – Qué es y por qué es importante | SAS." https://www.sas.com/es_co/insights/analytics/what-is-artificial-intelligence.html (accessed Mar. 16, 2021).

[2] "Aprendizaje automático: Qué es y por qué es importante | SAS." https://www.sas.com/es_co/insights/analytics/machine-learning.html (accessed Mar. 16, 2021).

[3] "Qué es el Procesamiento de Lenguaje Natural - Natural Language Processing? | SAS." https://www.sas.com/es_co/insights/analytics/what-is-natural-language-processing-nlp.html (accessed Mar. 16, 2021).

[4] M. Hossin, & M. S.-I. journal of data mining, and U. 2015, "A review on evaluation metrics for data classification evaluations," *academia.edu*, 2015, Accessed: Feb. 03, 2022. [Online]. Available: https://www.academia.edu/download/37219940/5215ijdkp01.pdf.

[5] H. Arteaga-Arteaga, A. Mora-Rubio, … F. F.-P. C., and undefined 2021, "Machine learning applications to predict two-phase flow patterns," *peerj.com*, Accessed: Feb. 03, 2022. [Online]. Available: https://peerj.com/articles/cs-798/.

[6] "Metrics to Evaluate your Machine Learning Algorithm | by Aditya Mishra | Towards Data Science." https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234 (accessed Jul. 14, 2021).

[7] J. Cerda and L. Cifuentes, "Uso de curvas ROC en investigación clínica: Aspectos teórico-prácticos," *Rev. Chil. infectología*, vol. 29, no. 2, pp. 138–141, Apr. 2012, doi: 10.4067/S0716-10182012000200003.

[8] B. W. Yap, K. A. Rani, H. A. Abd Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets," *Lect. Notes Electr. Eng.*, vol. 285 LNEE, pp. 13–22, 2014, doi: 10.1007/978-981-4585-18-7_2.

[9] "Stemming vs Lematización | Baeldung en Ciencias de la Computación." https://www.baeldung.com/cs/stemming-vs-lemmatization (accessed Jul. 01, 2021).

[10] R. Elsborg Madsen, S. Sigurdsson, L. Kai Hansen, and J. Larsen, "Pruning The Vocabulary For Better Context Recognition." Accessed: Jun. 17, 2021. [Online]. Available: www.imm.dtu.dk,.

[11] M. Decuyper *et al.*, "An overview of overfitting and its solutions,"

iopscience.iop.org, p. 22022, 2019, doi: 10.1088/1742-6596/1168/2/022022.

[12] J. Mccarthy, “What is Artificial Intelligence?,” 2007, Accessed: Aug. 24, 2021. [Online]. Available: http://www-formal.stanford.edu/jmc/.

[13] “Qué puede hacer Machine Learning por tu empresa - Zemsania.” https://zemsaniaglobalgroup.com/machine-learning-en-la-empresa/ (accessed Mar. 17, 2021).

[14] I. Lee and Y. J. Shin, “Machine learning for enterprises: Applications, algorithm selection, and challenges,” *Bus. Horiz.*, vol. 63, no. 2, pp. 157–170, Mar. 2020, doi: 10.1016/J.BUSHOR.2019.10.005.

[15] B. Marr and M. Ward, “Artificial Intelligence in Practice,” 2019. https://books.google.com.co/books?hl=es&lr=&id=UbaIDwAAQBAJ&oi=fnd&pg=PA1&dq=machine+learning+in+companies&ots=rOQKLGYHEZ&sig=BZfz_XYYzDtG-PvXl5ESmRt08fg&redir_esc=y#v=onepage&q=machine learning in companies&f=false (accessed Jun. 01, 2022).

[16] S. Paz, “Economía digital : el futuro ya llegó,” 2021, Accessed: Feb. 02, 2022. [Online]. Available: http://ridaa.unq.edu.ar/handle/20.500.11807/2990.

[17] J. McCarthy, “Artificial Intelligence, Logic and Formalizing Common Sense,” *Philos. Log. Artif. Intell.*, pp. 161–190, 1989, doi: 10.1007/978-94-009-2448-2_6.

[18] Priyanka, S. Azad, and R. Chakravarty, “Artificial intelligence (AI) literature in patents: a global landscape,” *Libr. Hi Tech News*, vol. 38, no. 7, pp. 24–28, 2021, doi: 10.1108/LHTN-09-2021-0062/FULL/XML.

[19] M. Razno, “Machine Learning Text Classification Model with NLP Approach,” 2019.

[20] L. Deng and Y. Liu, “A Joint Introduction to Natural Language Processing and to Deep Learning,” *Deep Learn. Nat. Lang. Process.*, pp. 1–22, Jan. 2018, doi: 10.1007/978-981-10-5209-5_1.

[21] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural Language Processing: State of The Art, Current Trends and Challenges,” Aug. 2017, Accessed: Feb. 02, 2022. [Online]. Available: https://arxiv.org/abs/1708.05148v1.

[22] A. Masood Khan and K. Rahat Afreen, “An approach to text analytics and text mining in multilingual natural language processing,” *Mater. Today Proc.*, Jan. 2021, doi: 10.1016/j.matpr.2020.10.861.

[23] M. Sivakami, “Text classification techniques: A literature review,” *Interdiscip. J. Information, Knowledge, Manag.*, vol. 13, pp. 117–135, 2018, doi: 10.28945/4066.

[24] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text Classification Algorithms: A Survey," *Inf. 2019, Vol. 10, Page 150*, vol. 10, no. 4, p. 150, Apr. 2019, doi: 10.3390/INFO10040150.

[25] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimed. Tools Appl.*, vol. 78, no. 3, pp. 3797–3816, Feb. 2019, doi: 10.1007/S11042-018-6083-5/TABLES/2.

[26] E. Alpaydin, "Introduction to machine learning," p. 682, 2020, Accessed: Feb. 02, 2022. [Online]. Available: https://books.google.com/books/about/Introduction_to_Machine_Learning_fourth.html?hl=es&id=tZnSDwAAQBAJ.

[27] J. Alzubi, A. Nayyar, and A. Kumar, "Machine Learning from Theory to Algorithms: An Overview," *J. Phys. Conf. Ser.*, vol. 1142, no. 1, p. 012012, Nov. 2018, doi: 10.1088/1742-6596/1142/1/012012.

[28] "¿Clasificación o Regresión? - IArtificial.net," 2020. https://www.iartificial.net/clasificacion-o-regresion/ (accessed Jun. 15, 2021).

[29] A. I. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 273–292, Jun. 2019, doi: 10.1007/s10462-018-09677-1.

[30] C. Sharpe, T. Wiest, P. Wang, and C. C. Seepersad, "A comparative evaluation of supervised machine learning classification techniques for engineering design applications," *J. Mech. Des. Trans. ASME*, vol. 141, no. 12, Dec. 2019, doi: 10.1115/1.4044524/958463.

[31] A. Ng, M. J.-A. in neural Information, and U. 2001, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," *proceedings.neurips.cc*, Accessed: Feb. 03, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2001/hash/7b7a53e239400a13bd6be6c91c4f6c4e-Abstract.html.

[32] L. Hong Lee, V. P. Kallimani, R. Rajkumar, D. Isa, V. Kallimani, and R. RajKumar, "Text Document Preprocessing with the Bayes Formula for Classification Using the Support Vector Machine," *ieeexplore.ieee.org*, 2008, doi: 10.1109/TKDE.2008.76.

[33] V. K. Vijayan, K. R. Bindu, and L. Parameswaran, "A comprehensive study of text classification algorithms," *2017 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2017*, vol. 2017-January, pp. 1109–1113, Nov. 2017, doi: 10.1109/ICACCI.2017.8125990.

[34] M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, "A Novel Active Learning Method Using SVM for Text Classification," *Int. J. Autom. Comput. 2016 153*, vol.

15, no. 3, pp. 290–298, Jul. 2016, doi: 10.1007/S11633-015-0912-Z.

[35] A. Wibowo Haryanto, E. Kholid Mawardi, and Muljono, “Influence of Word Normalization and Chi-Squared Feature Selection on Support Vector Machine (SVM) Text Classification,” *Proc. - 2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018*, pp. 229–233, Nov. 2018, doi: 10.1109/ISEMANTIC.2018.8549748.

[36] M. Drewnik and Z. Pasternak-Winiarski, “SVM kernel configuration and optimization for the handwritten digit recognition,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10244 LNCS, pp. 87–98, 2017, doi: 10.1007/978-3-319-59105-6_8.

[37] R. Wang, R. Ridley, W. Qu, X. D.-I. P. & Management, and U. 2021, “A novel reasoning mechanism for multi-label text classification,” *Elsevier*, 2021, Accessed: Feb. 03, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457320309341.

[38] Y. Xu, C. H. Shieh, P. van Esch, and I. L. Ling, “AI customer service: Task complexity, problem-solving ability, and usage intention,” *Australas. Mark. J.*, vol. 28, no. 4, pp. 189–199, Nov. 2020, doi: 10.1016/j.ausmj.2020.03.005.

[39] M. Raza, F. K. Hussain, O. K. Hussain, M. Zhao, and Z. ur Rehman, “A comparative analysis of machine learning models for quality pillar assessment of SaaS services by multi-class text classification of users’ reviews,” *Futur. Gener. Comput. Syst.*, vol. 101, pp. 341–371, Dec. 2019, doi: 10.1016/j.future.2019.06.022.

[40] S. A. Pérez, V. Profesor Guía, R. Alfaro, A. Profesor Co-Referente, : Héctor, and A. Cid, “‘Análisis y Clasificación de Textos con Técnicas Semi Supervisadas Aplicado a Área Atención al Cliente,’” 2017.

[41] E. K. Ikonomakis, M. Ikonomakis, S. Kotsiantis, and V. Tampakas, “Text Classification Using Machine Learning Techniques,” 2005. Accessed: Jun. 15, 2021. [Online]. Available: https://www.researchgate.net/publication/228084521.

[42] A. Guío Español, E. Tamayo Uribe, P. Gómez Ayerbe, and M. P. Mujica, *Marco ético para la inteligencia artificial en Colombia*. 2021.

[43] A. S. Miguel Antonio and P. R. Wilmer Darío, “Natural lenguaje processing para la predicción de series de tiempo en el mercado de commodities energético con el uso de modelos en inteligencia artificial,” *Nat. Leng. Process. para la predicción Ser. tiempo en el Merc. Commod. energético con el uso Model. en Intel. Artif.*, 2020, doi: 10.15332/DT.INV.2020.01355.

[44] Q. Z. R., “Bases de datos y su importancia dentro de una Organización.” https://www.gestiopolis.com/bases-datos-importancia-dentro-una-organizacion/

(accessed Mar. 16, 2021).

[45] "¿Qué importancia tienen las bases de datos a nivel empresarial?" https://www.datacentric.es/blog/bases-datos/importancia-bases-de-datos-2/ (accessed May 04, 2021).

[46] "Machine learning y las ventajas para los negocios | Negocios | Portafolio." https://www.portafolio.co/negocios/machine-learning-y-las-ventajas-para-los-528996 (accessed Mar. 16, 2021).

[47] "La Inteligencia Artificial como herramienta para acelerar el progreso de los ODS – Desarrollo Sostenible." https://www.un.org/sustainabledevelopment/es/2017/10/la-inteligencia-artificial-como-herramienta-para-acelerar-el-progreso-de-los-ods/ (accessed Mar. 23, 2021).

[48] "La Inteligencia Artificial, un aliada para alcanzar los Objetivos de Desarrollo Sostenible - Gaceta Médica." https://gacetamedica.com/mas/rsc/la-inteligencia-artificial-un-aliada-para-alcanzar-los-objetivos-de-desarrollo-sostenible/ (accessed May 05, 2021).

[49] I. C. Vélez Agudelo, M. I., Chavarro Bohórquez, D. A., Hernández Tasco, A., Niño Mendieta, Á. M., Tovar Narváez, G. E., & Montenegro Trujillo, "Libro Verde 2030: Política Nacional de Ciencia e Innovación para el Desarrollo Sostenible.," 2018.

[50] "¿Qué es la minería de datos? | SAS." https://www.sas.com/es_co/insights/analytics/data-mining.html (accessed Mar. 16, 2021).

[51] E. Ribas, "¿Qué es el Data Mining o minería de datos?," *Think. Innov.*, Jan. 2018, Accessed: Mar. 16, 2021. [Online]. Available: https://www.iebschool.com/blog/data-mining-mineria-datos-big-data/.

[52] Minciencias, "Technology Readiness Levels (TRL) Descripción." https://minciencias.gov.co/sites/default/files/upload/convocatoria/anexo5_7.pdf (accessed Jun. 23, 2021).

[53] "¿Cuáles son los 9 niveles de madurez de la Tecnología (TRLs)?" https://www.gestionfondosmexico.mx/single-post/2016/07/22/niveles-de-madurez-de-la-tecnología-trls (accessed Jun. 23, 2021).

[54] J. Miguel Ibañez de Aldecoa Quintana, "Niveles de madurez de la tecnología."

[55] L. Da Xu, Y. Lu, and L. Li, "Embedding Blockchain Technology into IoT for Security: A Survey," *IEEE Internet Things J.*, 2021, doi: 10.1109/JIOT.2021.3060508.

[56] "Machine learning: conoce qué es y las diferencias entre sus tipos." https://empresas.blogthinkbig.com/que-algoritmo-elegir-en-ml-aprendizaje/ (accessed Jun. 15, 2021).

[57] O. Simeone, "A Very Brief Introduction to Machine Learning With Applications to Communication Systems."

[58] R. Tabares-Soto, S. Orozco-Arias, V. Romero-Cano, V. S. Bucheli, J. L. Rodríguez-Sotelo, and C. F. Jiménez-Varón, "A comparative study of machine learning and deep learning algorithms to classify cancer types based on microarray gene expression data," *PeerJ Comput. Sci.*, vol. 2020, no. 4, 2020, doi: 10.7717/PEERJ-CS.270/TABLE-1.

[59] S. Kotsiantis, I. Zaharakis, P. P.-A. I. Review, and undefined 2006, "Machine learning: a review of classification and combining techniques," *Springer*, vol. 26, no. 3, pp. 159–190, Nov. 2006, doi: 10.1007/s10462-007-9052-3.

[60] L. C.-M. Nursing and U. 2020, "Logistic regression," *search.proquest.com*, 2020, Accessed: Feb. 03, 2022. [Online]. Available: https://search.proquest.com/openview/e8e7564d6f02ac54d757f3b74422f0ef/1?pq-origsite=gscholar&cbl=30764.

[61] P. Tsangaratos and I. Ilia, "Comparison of a logistic regression and Naïve Bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size," *CATENA*, vol. 145, pp. 164–179, Oct. 2016, doi: 10.1016/J.CATENA.2016.06.004.

[62] F. Nie, Z. Wang, R. Wang, Z. Wang, and X. Li, "Adaptive local linear discriminant analysis," *ACM Trans. Knowl. Discov. Data*, vol. 14, no. 1, Jan. 2020, doi: 10.1145/3369870.

[63] S. Balakrishnama, A. G.-I. for S. and, and undefined 1998, "Linear discriminant analysis-a brief tutorial," *music.mcgill.ca*, Accessed: Feb. 03, 2022. [Online]. Available: http://www.music.mcgill.ca/~ich/classes/mumt611_07/classifiers/lda_theory.pdf.

[64] R. Bernstein, M. Osadchy, D. Keren, and A. Schuster, "LDA classifier monitoring in distributed streaming systems," *J. Parallel Distrib. Comput.*, vol. 123, pp. 156–167, Jan. 2019, doi: 10.1016/J.JPDC.2018.09.017.

[65] E. A. Hidayat Fajrian Nur Dr Azah Kamilah Binti Draman *et al.*, "A comparative study of feature extraction using PCA and LDA for face recognition," *ieeexplore.ieee.org*, pp. 5–8, 2011, Accessed: Feb. 22, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6122779/.

[66] M. Dumont, R. Marée, … L. W.-… C. on C., and U. 2009, "Fast multi-class image

annotation with random subwindows and multiple output randomized trees," *orbi.uliege.be*, 2009, Accessed: Feb. 03, 2022. [Online]. Available: https://orbi.uliege.be/handle/2268/12205.

[67] P. H. Swain and H. Hauska, "Decision Tree Classifier: Design and Potential," *IEEE Trans Geosci Electron*, vol. GE-15, no. 3, pp. 142–147, 1977, doi: 10.1109/TGE.1977.6498972.

[68] "What is a Random Forest? | TIBCO Software." https://www.tibco.com/reference-center/what-is-a-random-forest (accessed Feb. 08, 2022).

[69] A. Géron, "Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems," 2019, Accessed: Feb. 03, 2022. [Online]. Available: https://books.google.com/books?hl=es&lr=&id=HnetDwAAQBAJ&oi=fnd&pg=PT9&dq=Hands-on+machine+learning+with+Scikit-Learn,+Keras,+and+TensorFlow:+Concepts,+tools,+and+techniques+to+build+intelligent+systems&ots=kPUuxFFMB2&sig=-ru73dNE9kacB9mQ1lhoTMMmVJA.

[70] L. Breiman, "Random Forests," *Mach. Learn. 2001 451*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[71] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn. 2006 631*, vol. 63, no. 1, pp. 3–42, Mar. 2006, doi: 10.1007/S10994-006-6226-1.

[72] M. Götz *et al.*, "Extremely randomized trees based brain tumor segmentation," Accessed: Feb. 21, 2022. [Online]. Available: https://www.researchgate.net/publication/267762444.

[73] G. Guo, H. Wang, D. Bell, Y. Bi, … K. G.-O. the M. to M. I., and undefined 2003, "KNN model-based approach in classification," *Springer*, vol. 2888, pp. 986–996, 2003, doi: 10.1007/978-3-540-39964-3_62.

[74] J. Brownlee, "Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch," 2016. https://scholar.google.com/scholar_lookup?title=Master Machine Learning Algorithms%3A Discover how They Work and Implement Them from Scratch&publication_year=2016&author=J. Brownlee (accessed Feb. 07, 2022).

[75] R. Gholami, N. F.-H. of neural computation, and undefined 2017, "Support vector machine: principles, parameters, and applications," *Elsevier*, Accessed: Feb. 03, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128113189000272.

[76] A. A. Arzucan¨ozgür, "Supervised and unsupervised machine learning techniques for text document categorization," 2002. Accessed: Jun. 15, 2021. [Online]. Available:

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.699.7643&rep=rep1&type=pdf.

[77] Z. Wang, X. Sun, D. Z.-2006 I. C. on, and undefined 2006, "An optimal Text categorization algorithm based on SVM," *ieeexplore.ieee.org*, Accessed: Jun. 15, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4064327/.

[78] A. Perez, P. Larranaga, I. I.-I. J. of Approximate, and undefined 2006, "Supervised classification with conditional Gaussian networks: Increasing the structure complexity from naive Bayes," *Elsevier*, Accessed: Feb. 03, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888613X0600003X.

[79] C. C. Aggarwal and C. X. Zhai, "A Survey of Text Classification Algorithms," *Min. Text Data*, vol. 9781461432234, pp. 163–222, Aug. 2012, doi: 10.1007/978-1-4614-3223-4_6.

[80] E. D. Liddy, "Natural Language Processing," *Encycl. Libr. Inf. Sci. 2nd Ed. NY. Marcel Decker, Inc.*, 2001, Accessed: Jun. 23, 2021. [Online]. Available: http://surface.syr.edu/cnlp/11.

[81] A. Chopra, A. Prashar, and C. Sain, "Natural Language Processing," *Int. J. Technol. Enhanc. Emerg. Eng. Res.*, vol. 1, no. 4, 2013, Accessed: Mar. 19, 2021. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.407.6907&rep=rep1&type=pdf.

[82] S. Jusoh, H. A.-F.-2007 I. C. on, and undefined 2007, "Natural language interface for online sales systems," *ieeexplore.ieee.org*, Accessed: Jun. 29, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4658379/.

[83] E. Leopold and J. Kindermann, "Text categorization with support vector machines. How to represent texts in input space?," *Mach. Learn.*, vol. 46, no. 1–3, pp. 423–444, 2002, doi: 10.1023/A:1012491419635.

[84] "Stop words list." https://countwordsfree.com/stopwords/spanish (accessed Jun. 29, 2021).

[85] J. Plisson, N. Lavrac, and D. Mladenic, "A rule based approach to word lemmatization." Accessed: Jun. 17, 2021. [Online]. Available: https://www.researchgate.net/publication/228525639.

[86] "El proceso de implementación de TfidfVectorizer en sklearn.feature_extraction.text - programador clic." https://programmerclick.com/article/7138217210/ (accessed Jul. 01, 2021).

[87] V. Srividhya and R. Anitha, "Evaluating Preprocessing Techniques in Text Categor

ization," *Int. J. Comput. Sci. Appl. Issue*, 2010.

[88] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, pp. 243–248, Apr. 2020, doi: 10.1109/ICICS49469.2020.239556.

[89] "A Gentle Introduction to the Fbeta-Measure for Machine Learning." https://machinelearningmastery.com/fbeta-measure-for-machine-learning/ (accessed Jul. 14, 2021).

[90] "Protección de Datos Personales - Ministerio de Educación Nacional de Colombia." https://www.mineducacion.gov.co/1759/w3-article-387771.html?_noredirect=1 (accessed Aug. 11, 2021).

[91] "El Hábeas Data, protección al derecho a la información y a la autodeterminación informativa." http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S2071-50722016000200002 (accessed Aug. 11, 2021).

[92] "¿Qué es SaaS? Software como servicio | Microsoft Azure." https://azure.microsoft.com/es-es/overview/what-is-saas/ (accessed Mar. 25, 2021).

[93] Y. Zhang, S. Peng, J. L.-C. Engineering, and undefined 2006, "Improvement and application of tfidf method based on text classification," *en.cnki.com.cn*, Accessed: Jun. 17, 2021. [Online]. Available: https://en.cnki.com.cn/Article_en/CJFDTotal-JSJC200619027.htm.

[94] G. Pui Cheong Fung, J. Xu Yu, H. Wang, D. W. Cheung, and H. Liu, "A Balanced Ensemble Approach to Weighting Classifiers for Text Classification." Accessed: Jun. 17, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4053118/.

[95] "Python Software Foundation." https://www.python.org/psf/ (accessed Jun. 02, 2022).

[96] F. . Pedregosa, G. . Varoquaux, A. . Gramfort, V. Michel, B. Thirion, and O. Grisel, "Scikit-learn: Machine Learning in Python," 2011. Accessed: Jun. 16, 2021. [Online]. Available: http://scikit-learn.sourceforge.net.

[97] B. Komer, J. Bergstra, C. E.-I. workshop on AutoML, and undefined 2014, "Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn," *Citeseer*, 2014, Accessed: Aug. 19, 2021. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.924.9697&rep=rep1&type=pdf.

[98] L. Demidova, I. K.-2017 6th M. Conference, and undefined 2017, "SVM

classification: Optimization with the SMOTE algorithm for the class imbalance problem," *ieeexplore.ieee.org*, Accessed: May 17, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7977136/.

[99] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, Jun. 2004, doi: 10.1145/1007730.1007735.

[100] M. Feurer, A. Klein, K. E. Jost, T. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," *proceedings.neurips.cc*, Accessed: May 17, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/5872-efficient-and-robust-automated-machine-learning.

[101] A. Fernández, S. Garcia, F. Herrera, N. C.-J. of artificial, and undefined 2018, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *jair.org*, vol. 61, pp. 863–905, 2018, Accessed: May 18, 2022. [Online]. Available: https://www.jair.org/index.php/jair/article/view/11192.

[102] C. Ramezan, T. Warner, A. M.-R. Sensing, and undefined 2019, "Evaluation of sampling and cross-validation tuning strategies for regional-scale machine learning classification," *mdpi.com*, doi: 10.3390/rs11020185.

[103] B. Komer, J. Bergstra, C. E.-I. workshop on AutoML, and undefined 2014, "Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn," *Citeseer*, 2014, Accessed: May 19, 2022. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.924.9697&rep=rep1&type=pdf.