Modelo de clasificación de peticiones, quejas, reclamos y sugerencias en el sector financiero

Trabajo de grado para optar por el título de Magíster en analítica para inteligencia de negocios

Jaime Enrique Chacón González
Luis Omar García Amador
Rubén Darío Molina Castro

Directores
Juan Pablo Mora
Juan Pablo Pájaro

Facultad de Ingeniería
Maestría en Analítica para Inteligencia de Negocios

23 de mayo de 2022
Bogotá DC

Resumen

Las redes sociales han sido uno de los acontecimientos de mayor importancia e impacto en la historia reciente, estas han evolucionado desde un intercambio electrónico directo de información, hasta convertirse en un lugar de reuniones virtuales, plataformas de negocios y herramientas de marketing esencial del siglo XXI. Esta tendencia ha incrementado el uso de las redes sociales como canal de comunicación de opiniones y comentarios de los clientes sobre productos y servicios de las empresas, por lo tanto, existe una oportunidad enorme en el aprovechamiento de esta cercanía con el usuario y entender sobre qué están hablando los usuarios. Es así como se propone el desarrollo de un modelo que permita clasificar los comentarios de los usuarios dependiendo del contenido lingüístico del mismo.

El desarrollo del presente proyecto se realiza un enfoque en el análisis de comentarios de la red social Twitter en el sector financiero colombiano el cuál será abarcado partir de la metodología CRISP-DM.

Palabras Clave: Peticiones, quejas, reclamos, sugerencias, redes sociales, sector financiero, procesamiento de lenguaje natural, algoritmos de clasificación, Word2Vec

Abstract

Social networks have been one of the most important and impressive events in recent history, they have evolved from a direct electronic exchange of information, to become a place for virtual meetings, business platforms and essential marketing tools of the 21st century. This trend has increased the use of social networks as a channel for communicating opinions and comments from customers about company's products and services, therefore, there is an enormous opportunity in taking advantage of this closeness with the user and understanding what they are talking about. This is how the development of a model that allows classifying user comments depending on their linguistic content is proposed.

This project focuses on the analysis of comments from the social network Twitter in the Colombian financial sector, which will be developed regarding the CRISP-DM methodology.

Keywords: petitions, complaints, claims, suggestions, social networks, financial sector, natural processing language, classification algorithms, Word2Vec

Tabla de Contenido

1. INTRODUCCIÓN 7

1.1. PLANTEAMIENTO DEL PROBLEMA 7

1.2. METODOLOGÍA 7

2. DESARROLLO DEL PROYECTO 10

2.1. ENTENDIMIENTO DEL NEGOCIO 10

2.2. ENTENDIMIENTO DE LOS DATOS 16

2.3. PREPARACIÓN DE LOS DATOS 20

2.4. MODELAMIENTO 29

2.5. EVALUACIÓN DE MODELOS 35

3. RECOMENDACIONES Y CONCLUSIONES 40

3.1. RECOMENDACIONES 40

3.2. CONCLUSIONES 40

BIBLIOGRAFÍA 42

Índice de Tablas

Tabla 1. Cuentas Twitter Sector Financiero. 17

Tabla 2. Descripción de variables Dataset TripAdvisor 17

Tabla 3. Descripción de variables Dataset Complaints 18

Tabla 4. Descripción de variables Dataset Tweets del sector financiero de Colombia 19

Tabla 5. Distribución de frecuencia variable Score 20

Tabla 6. Distribución de frecuencia por Etiqueta 21

Tabla 7. Media y Desviación de palabras usadas por etiqueta de reviews 23

Tabla 8. Top 5 palabras más utilizadas por categoría en reviews 23

Tabla 9. Top 5 palabras más utilizadas por categoría en reviews sin Stopwords 24

Tabla 10. Resultados Accuracy 36

Tabla 11. Resultados Sensitivity. 36

Tabla 12. Resultados Specificity 37

Tabla 13. Resultados AUC. 38

Índice de Ilustraciones

Ilustración 1. Modelo de Referencia CRISP - DM. 8

Ilustración 2. Tareas por etapa del CRISP - DM. 9

Ilustración 3. Pipeline Proyecto 13

Ilustración 4. Palabras totales por clase de reviews. 22

Ilustración 5. Palabras distintas por clase de reviews. 22

Ilustración 6. Nube de palabras reviews 24

Ilustración 7. Análisis sintáctico frase reviews restaurantes. 25

Ilustración 8. Análisis sintáctico de frase de review financiera 26

Ilustración 9. Lista de palabras a excluir por categoría sintáctica 26

Ilustración 10. Distribución de frecuencia por vía de comunicación de complaints financieros . 29

1. Introducción

1.1. Planteamiento del problema

Las redes sociales han sido uno de los acontecimientos de mayor importancia e impacto en la historia reciente, estas han evolucionado desde un intercambio electrónico directo de información, hasta convertirse en un lugar de reuniones virtuales, plataformas de negocios y herramientas de marketing esencial del siglo XXI. De tal manera y con tanta rapidez, que se han convertido en la base y el puente digital entre los consumidores y las empresas (Euromonitor, 2021).

Un estudio desarrollado por la consultora Guidance (2012) concluye que las redes sociales se han convertido en la principal fuente de información, para los consumidores. Hoy en día, antes de efectuar una compra, el 67% de los usuarios, contrastan opiniones en Internet y, de estos, el 89% reconoce que se sienten influenciados o muy influenciados por estos comentarios a la hora de tomar una decisión.

De acuerdo con lo mencionado anteriormente, existe una gran oportunidad de negocio para las empresas en cuanto a la recopilación y entendimiento de todos los comentarios y publicaciones relacionados a sus marcas, productos y servicios. Sin embargo, esto genera un reto importante, debido a que cada comentario es distinto, estos pueden referirse a temas positivos o negativos sobre los productos o servicios y pueden tratar entre peticiones, quejas, reclamos o sugerencias. Debido a esto, se plantea desarrollar un modelo que permita clasificar estos comentarios o publicaciones, con el fin de que las empresas puedan identificar las opiniones de los consumidores sobre sus productos y servicios y de esta manera les permita desarrollar estrategias y proyectos en busca de la mejora de estos, a partir del correcto análisis de las redes sociales.

Para el desarrollo del presente proyecto se realiza un enfoque en el análisis de comentarios de la red social Twitter en el sector financiero colombiano.

1.2. Metodología

El presente trabajo de grado es desarrollado mediante la metodología de minería de datos CRISP-DM (Cross Industry Standard Process for Data Mining). El procedimiento consiste en un ciclo que comprende seis etapas debidamente organizadas, estructuradas y definidas (Azevedo & Santos, 2008), las cuales, de acuerdo con Chapman et al. (2000), se detallan a continuación:

Entendimiento del negocio: Siendo la etapa inicial de la metodología, se enfoca en entender los objetivos y requerimientos del proyecto desde una perspectiva de negocio, con el fin de usar esta información y poder convertirla en la definición de un problema de minería de datos y un plan preliminar diseñado para lograr los objetivos.

Entendimiento de los datos: Inicia con la recopilación y familiarización con los datos, se identifican problemas de calidad, primeros conocimientos y subconjuntos para formar una hipótesis sobre información oculta.

Preparación de los datos: Esta etapa cubre todas las actividades para construir el conjunto de datos final a partir de los datos sin procesar iniciales. Existe la posibilidad de que las tareas de preparación de los datos se realicen varias veces sin ningún orden prescrito. Entre las principales tareas, se encuentran la selección de tablas, registros, y atributos, así como la limpieza de datos para herramientas de modelado.

Modelamiento: En esta etapa se seleccionan y aplican varias técnicas de modelado y se realiza la optimización de sus parámetros. Por lo general, existen varias técnicas para el mismo tipo de problema de minería de datos. Algunas técnicas tienen requisitos específicos sobre la forma de los datos. Por lo tanto, a menudo es necesario retroceder a la etapa de preparación de datos.

Evaluación: Después del desarrollo de los modelos es importante su evaluación y comparación profunda, con el fin de asegurarse que cumplan con los objetivos de negocio. Es indispensable determinar si existe algún objetivo que no se esté cumpliendo o no se haya considerado. Al final de esta etapa se debe llegar a una decisión sobre el uso de los resultados de minería de datos.

Despliegue: En esta etapa final se organiza y presenta el conocimiento obtenido, junto con las recomendaciones finales al cliente.

En las siguientes ilustraciones se detallan el proceso y tareas de acuerdo con sus respectivas etapas:

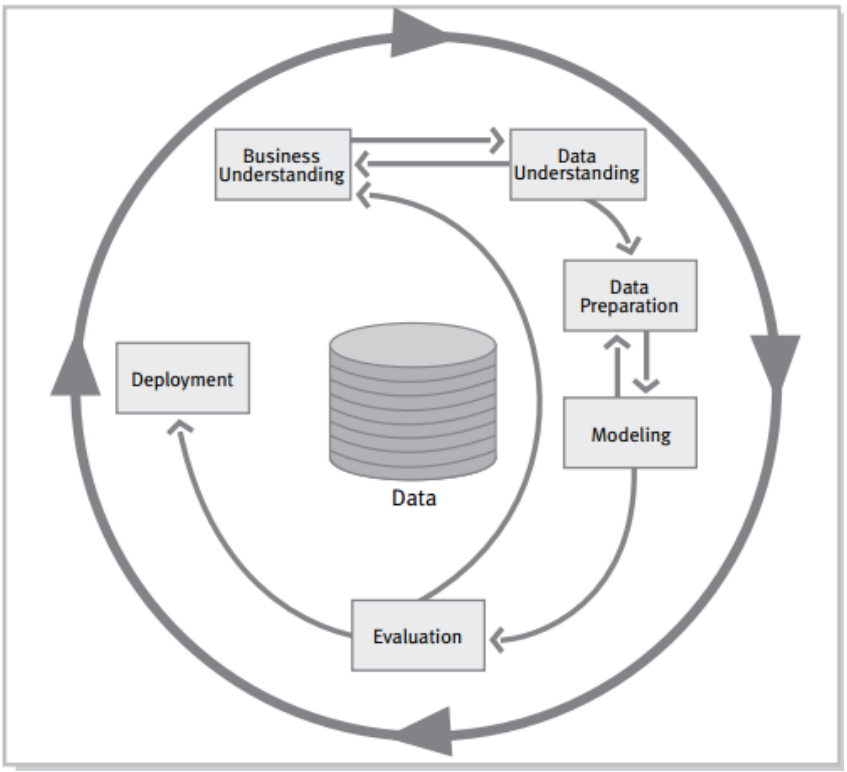

Ilustración 1. Modelo de Referencia CRISP - DM.

Fuente: Chapman et al. (2000)

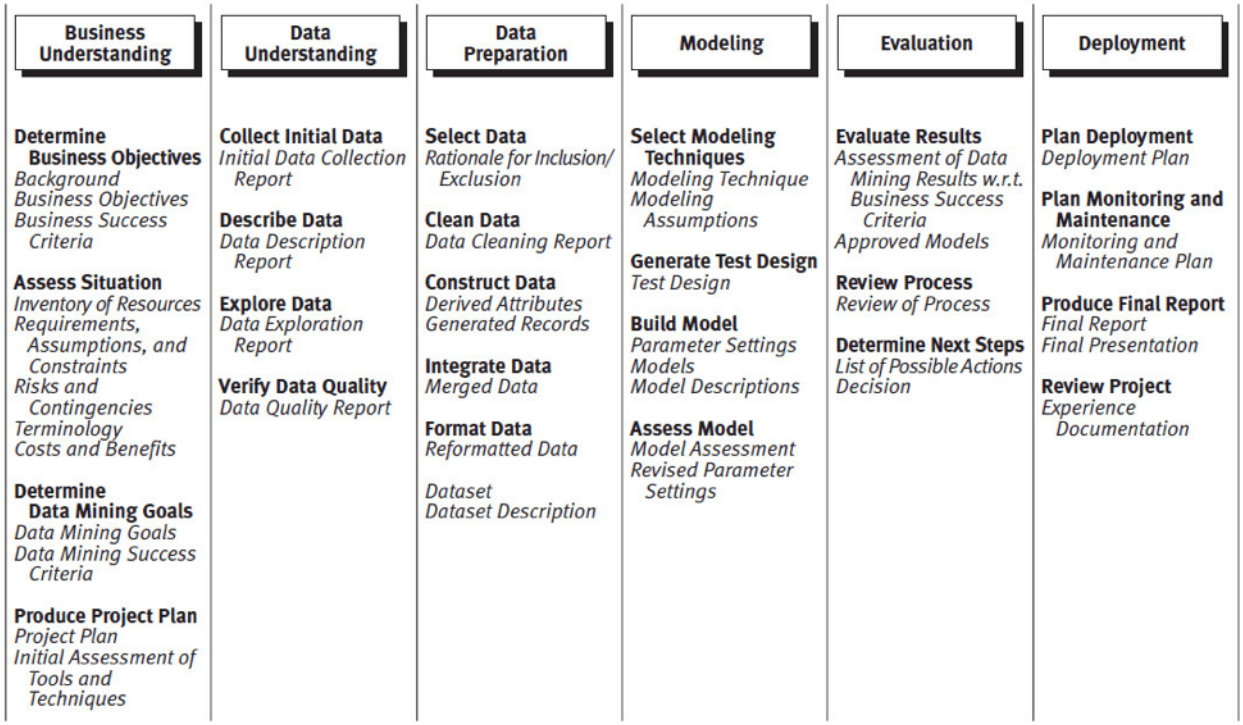

Ilustración 2. Tareas por etapa del CRISP - DM.

Fuente: Chapman et al. (2000)

2. Desarrollo del proyecto

2.1. Entendimiento del negocio

Alianza CAOBA, es el primer centro de excelencia y apropiación que apoya el uso de las tecnologías de Big Data y Data Analytics en Colombia, la cual se encuentra constituida por las empresas Grupo Bancolombia, Grupo Nutresa, IBM de Colombia, SAS Institute Colombia, DELL, Clúster CREATIC y las Universidades ICESI, EAFIT, Los Andes y la Pontificia Universidad Javeriana, que actúa como ejecutor del proyecto. Cuenta con tres líneas de servicios realizadas de la mano de las universidades mencionadas, apoyadas por docentes investigadores siendo las siguientes:

Investigación aplicada: consiste en el desarrollo de soluciones aplicables en diversos contextos del país en cuanto a problemáticas reales surgidas de los sectores público y privado.

Generación de capacidades: apoyo en el desarrollo de grupos de investigación y formación a partir de la transferencia de conocimiento especializado a la industria nacional, dándole gran importancia a la vinculación de oferta de líderes nacionales en Big Data y Data Analytics con la demanda de grandes empresas de sectores estratégicos de la economía.

Consultoría: servicios para empresas colombianas, que faciliten la apropiación de tecnologías, metodologías y herramientas de análisis de datos y procesamiento de grandes volúmenes de información.

Entre sus áreas de experiencia, se encuentran la visualización de datos, detección de anomalías, Machine Learning, Big Data, analítica de datos, gobierno de datos y procesamiento de lenguaje natural. En la última área mencionada, CAOBA requiere desarrollar un modelo que le permita clasificar los distintos comentarios o publicaciones en la red social Twitter, entre peticiones, quejas, reclamos y sugerencias, con el fin de incluirlo en su línea de servicio de consultorías en el sector financiero.

Los tipos de comentarios del modelo presentan las siguientes definiciones (PUJ, 2022):

- **Petición:** escrito presentado con el fin de solicitar información sobre algún asunto de interés o consulta.

- **Queja:** manifestación de protesta, censura, descontento o inconformidad que formula una persona con relación a sus servicios o sus procesos de gestión.

- **Reclamo:** es todo derecho que tiene toda persona de exigir, reivindicar o demandar una solución particular referente a la prestación indebida de un servicio o la falta de atención de una solicitud que se considera por parte del usuario como incompleta, injusta o ausente de lo previamente acordado o esperado.

- **Sugerencia:** manifestación de una idea o propuesta para mejorar el servicio o la gestión de la universidad.

Es de gran importancia entender el impacto que generan los comentarios de productos y servicios, sean positivos o negativos, en redes sociales sobre la imagen de una empresa. En función del sector, entre un 11% y un 27% de las pérdidas de clientes actuales se deben a la influencia que ejercen los comentarios vertidos en la Red. (Guidance, 2012). En un solo comentario, las personas pueden desmeritar las marcas, realizar comparaciones con la competencia, criticar el desempeño de productos y servicios y comentar sobre su experiencia con estos. Toda esa información en las redes sociales tiene un potencial de gran importancia para la toma de decisiones en las empresas, sin embargo, uno de los principales inconvenientes radica en la recolección, clasificación y análisis de esta información.

Adicionalmente, existe cada vez más interacción del cliente con los canales de comunicación con las empresas, en búsqueda de mejoras y soluciones para los inconvenientes que se le presenten, por lo que estar al tanto de esa información en tiempo real incrementa el valor de una marca o empresa percibido por el usuario y de esta manera, su fidelidad y compromiso con esta.

Por lo tanto, el desarrollo de una herramienta que permita identificar los comentarios que afecten la reputación de las empresas puede generar grandes oportunidades para estas con opciones como el desarrollo de estrategias y proyectos en pro de disminuir los comentarios negativos a partir de acciones proactivas y enfocadas en el consumidor con malas experiencias.

De acuerdo con lo mencionado anteriormente, se definen los siguientes objetivos:

2.1.1. Objetivos de Negocio.

Objetivo General

- Desarrollar una herramienta que permita a CAOBA mejorar su consultoría en los sectores financieros, entregando una clasificación de Tweets en categoría positiva, siendo peticiones y sugerencias y negativa en cuanto a quejas y reclamos.

Objetivos Específicos

- Clasificar los tweets de los usuarios sobre el sistema financiero en valoraciones positivas o negativas.

- Proveer al cliente final de CAOBA un mecanismo de información oportuno, que dé cuenta del estado de las conversaciones que se están dando en Twitter respecto a los productos y servicios del sistema financiero.

- Disponer un algoritmo de clasificación de valoraciones positivas o negativas, teniendo como insumo comentarios de redes sociales tales como Twitter, que permita a CAOBA replicarlo en distintos sectores de servicio a demás del financiero.

2.1.2. Objetivos de analítica.

Objetivo general

- Entrenar un algoritmo de clasificación a partir de textos provenientes de un dominio de servicio, junto con el vocabulario de comentarios sobre el sector financiero de Estados Unidos, para ser usado en la clasificación de Tweets en el dominio del sistema financiero colombiano.

Objetivos Específicos

Dada la naturaleza del problema, siendo la clasificación de comentarios positivos y negativos, nuestra variable objetivo, se enfoca en los comentarios con calificaciones negativas, por lo tanto, se definen las siguientes métricas, las cuales fueron definidas a partir de revisión de bibliografía sobre experimentos y estudios realizados en cuanto a clasificación a partir del procesamiento de lenguaje natural (Patil & Kolhe, 2022):

- Obtener métricas de clasificación en los experimentos de entrenamiento como sigue:
    - Accuracy: >85%
    - Sensitivity: >85%
    - Specificity: >75%

- Obtener métricas de clasificación en los conjuntos de datos de prueba como sigue:
    - Accuracy: >80%
    - Sensitivity: >80%
    - Specificity: >70%

- Obtener métricas de clasificación en los conjuntos de datos de validación como sigue:
    - Accuracy: >70%
    - Sensitivity: >70%
    - Specificity: >60%

2.1.3. Restricciones.

- Existencia de bases de datos con clasificaciones de comentarios u opiniones del sector financiero en Colombia limitada.

- El procesamiento de lenguaje natural debe realizarse en bases de datos en español

2.1.4. Pipeline.

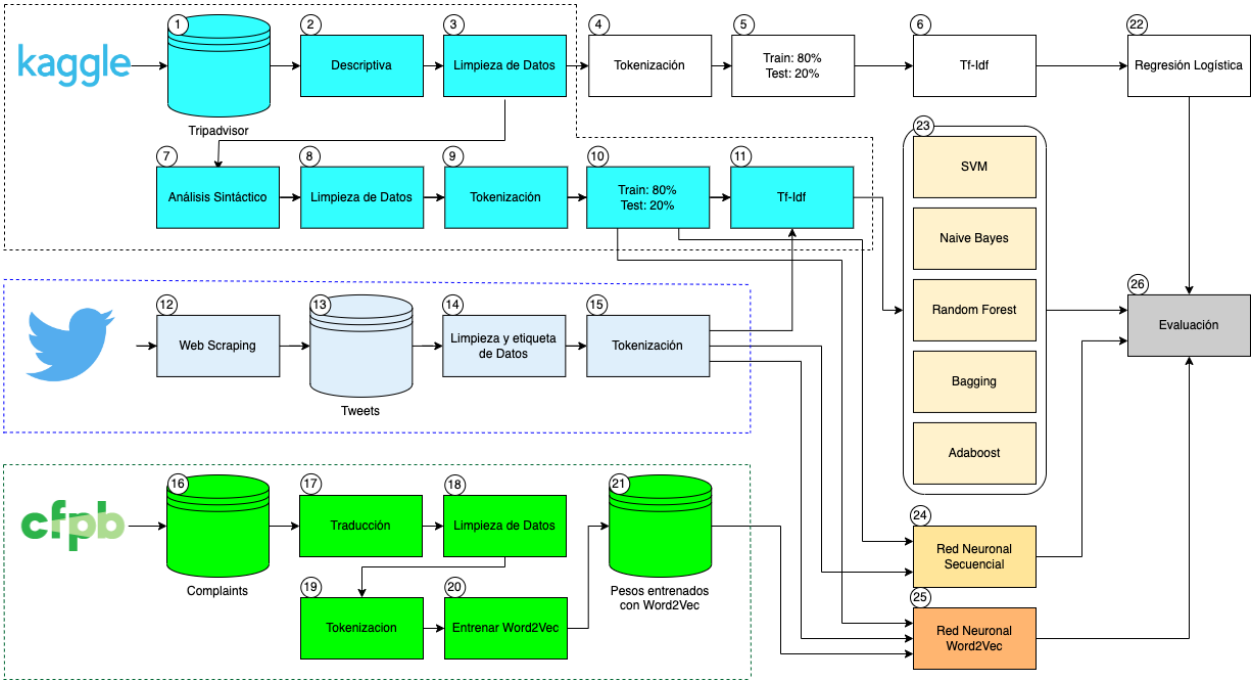

Ilustración 3. Pipeline Proyecto

Fuente: Elaboración propia

2.1.4.1. Descripción Pipeline.

1. **Tripadvisor Dataset:** Se importan datos desde una página web sobre un concurso realizado sobre "Clustering & Sentiment Analysis" en Kaggle (Lázaro, 2021)correspondientes a reviews realizadas sobre el servicio de restaurantes en Perú.

2. **Descriptiva:** Análisis descriptivo de texto de las reviews.

3. **Limpieza de Datos (1):** Sobre los datos cargados se aplica una primera limpieza de datos en la cual se eliminan reviews repetidos, nulos y caracteres especiales, seguido por una conversión a minúsculas de los datos, finalizando con una eliminación de "StopWords" genéricos.

4. **Tokenización (1):** Consiste en dividir las cadenas de texto en tokens, en este caso palabra por palabra, esta tokenización será utilizada para aplicar los datos de los reviews con una limpieza básica con fines de un modelo base.

5. **Train – Test (1):** Se dividió el conjunto de datos tokenizados en un 80% para entrenar el modelo y 20% para la prueba para implementar en un algoritmo base.

6. **Tf-Idf:** Consiste en una medida que pondera el uso de una determinada palabra dentro de un conjunto de documentos y que supone, por lo tanto, un elemento importante y relevante para la clasificación de documentos frente a la consulta de un usuario. Seguido por la construcción de una matriz de vectores asignando un valor numérico a cada una de las palabras a partir de los datos preprocesados el cual será utilizado en los algoritmos de machine learning.

7. **Análisis Sintáctico:** Proceso que permite identificar las características léxicas que aportan información de interés al modelo.

8. **Limpieza de Datos (2):** De acuerdo con el análisis sintáctico, se realiza una segunda limpieza la cual está orientada a remover todas las características léxicas que no agregan información, tales como: verbos, sustantivos y adjetivos propios del sector de restaurantes.

9. **Tokenización (2):** Tokenización de reviews después de la segunda limpieza de datos.

10. **Train – Test (2):** Se dividió el conjunto de datos tokenizados en un 80% para entrenar el modelo y 20% para la prueba, para usarse en modelos diferentes al algoritmo base.

11. **Tf-Idf:** Consiste en una medida que pondera el uso de una determinada palabra dentro de un conjunto de documentos y que supone, por lo tanto, un elemento importante y relevante para la clasificación de documentos frente a la consulta de un usuario. Seguido por la construcción de una matriz de vectores asignando un valor numérico a cada una de las palabras a partir de los datos preprocesados el cual será utilizado en los algoritmos de machine learning.

12. **Web Scraping:** se realiza usando la biblioteca de Pyhon “Twint” y consiste en buscar mensajes de Twitter en un conjunto de cuentas prestablecidas, siguiendo criterios de búsquedas definidos en los scripts realizados para tal fin.

13. **Tweets Dataset:** conjunto de datos obtenido a partir del proceso de Web Scraping en las cuentas de Twitter de instituciones financieras de Colombia generados entre el 18 de marzo y el 19 de abril de 2022.

14. **Limpieza y etiqueta de datos:** se realiza una limpieza del dataset de Tweets y luego se clasifican manualmente en etiquetas “Bueno” (Sugerencias y Peticiones) y “Malo” (Quejas y Reclamos), por tres (3) personas de forma independiente, el protocolo de etiquetado se encuentra explicado en la etapa de “Preparación de los datos”.

15. **Tokenización (3):** Se realiza tokenización en los datos obtenidos de los Tweets.

16. **Complaints Dataset:** Se importan datos desde la Oficina de Protección Financiera del Consumidor de Estados Unidos (CFPB, 2022) correspondientes a reseñas realizadas sobre el sector financiero del mismo país.

17. **Traducción:** Se realiza una traducción de las reseñas del sector financiero de inglés a español a partir de la API de GoogleTranslator.

18. **Limpieza de datos (3):** Sobre los datos de reseñas se eliminan registros repetidos y nulos.

19. **Tokenización (4):** Se realiza tokenización en los datos obtenidos de las reseñas financieras.

20. **Entrenar Word2Vec:** Consiste en una técnica de procesamiento de lenguaje natural, siendo una red neuronal desarrollada por Google en el 2015, que tiene como finalidad aprender de las asociaciones de palabras a partir de un corpus de gran tamaño, una vez entrenado, se produce una matriz de pesos asociados a cada palabra de vocabulario único con el objetivo de transferirlo a la red neuronal como capa de entrada de Embedding.

21. **Pesos entrenados con Word2Vec:** Matriz generada con 31645 filas (palabras únicas) y 100 columnas (tamaño determinado en la optimización de hiper parámetros), que usamos en la capa de Embedding de la red neuronal entrenada.

22. **Regresión Logística:** Se entrena un algoritmo de regresión logística como modelo base con parámetros por defecto.

23. **Modelos Tf-Idf:** Entrenamiento de algoritmos de una máquina de soporte vectorial, Naive Bayes, Random Forest, Bagging y Adaboost tomando como datos de entrada la matriz producida por el proceso de Tf-Idf.

24. **Red neuronal secuencial:** red neuronal entrenada a partir de la vista minable de las reseñas de restaurantes y los Tweets del sector financiero.

25. **Red Neuronal Word2Vec:** red neuronal entrenada a partir de la vista minable de las reseñas de restaurantes y los Tweets del sector financiero junto con la capa de Embedding desarrollada en el entrenamiento del Word2Vec.

26. **Evaluación:** Se aplicaron las métricas de desempeño a los resultados de los modelos de entrenamiento y prueba.

2.2. Entendimiento de los datos

2.2.1. Recolección de Datos Iniciales.

En el presente proyecto se recopilan tres (3) bases de datos iniciales obtenidas de distintas fuentes descritas a continuación[1]:

- **TripAdvisor Dataset:** Base de datos pública correspondiente a reviews realizados sobre el servicio de restaurantes en Perú, descargada desde la página web de Kaggle (Lázaro, 2021) usada en un concurso realizado sobre "Clustering & Sentiment Analysis". Se selecciona esta base debido a que los registros presentan un puntaje en valores entre uno (1) y cinco (5), dado por el usuario al momento del Review, adicionalmente, aunque no tienen información sobre el sector financiero, si presentan vocabulario sobre comentarios u opiniones de los usuarios respecto a la prestación de un servicio. Esto puede generar información importante para modelar otro servicio como el financiero. El dataset será utilizado para la construcción de la base de entrenamiento y prueba del modelo.

- **Complaints Dataset:** Base de datos pública correspondiente a reseñas realizadas por los usuarios sobre el sector financiero de Estados Unidos, descargada desde el portal de la Oficina de Protección Financiera del Consumidor de Estados Unidos, por sus siglas en inglés (CFPB, 2022) agencia que implementa y hace cumplir la ley financiera federal para el consumidor y garantiza que los mercados de productos financieros para el consumidor sean justos, transparentes y competitivos. Se selecciona este dataset con el objetivo de tener vocabulario del sector financiero y poder utilizarlo en la construcción del modelo en los algoritmos de redes neuronales como capa de embedding.

- **Tweets Sector Financiero Colombia:** conjunto de Tweets construido a partir del método Web Scraping, el cual se realiza usando la biblioteca de Pyhon "Twint" y consiste en buscar mensajes de Twitter en un conjunto de cuentas prestablecidas, siguiendo criterios de búsquedas definidos en los scripts realizados para tal fin, descargado a partir del periodo entre 18 de marzo y 19 de abril de 2022. El dataset será utilizado como base de validación para el modelo. Para la búsqueda se seleccionan las cuentas de Twitter de la siguiente tabla:

[1] Datos originales disponibles en el repositorio en el link: https://gitlab.com/CAOBA-Central/productos-caoba/datalab/poc-005-cam-pqr-imagenes-fase1-develop/-/tree/develop/data/raw

Tabla 1. Cuentas Twitter Sector Financiero.

Compañía	Cuenta de Twitter
Banco Agrario	@bancoAgrario
Banco Caja Social	@bancocajasocial
Banco de Bogotá	@bancodebogota
Banco de Occidente	@bco_Occidente
Banco Itaú	@itaucol
Banco Popular	@bco_Popular
Bancolombia	@bancolombia
BBVA	@bbva_colombia
Davivienda	@davivienda
Scoctiabank Colpatria	@scotiaColpatria

Fuente: elaboración propia

2.2.2. Descripción de los datos.

La descripción de las variables que contienen las bases de datos mencionadas anteriormente se detalla a continuación:

TripAdvisor: dataset que presenta 1.258.435 registros en nueve (9) variables de las cuales dos (2) son seleccionadas como variables de interés siendo: "review" y "score"

Tabla 2. Descripción de variables Dataset TripAdvisor

Variable	Tipo	Descripción	De Interés
id_review	String	Identificador único de los review	No
review	String	Texto que describe los cometarios o review	Si
title	String	Titulo de Review	No
score	Float64	Puntaje dado a cada review entre 1 y 5	Si
likes	Int64	Cantidad de puntos por aceptación de otros usuarios de la plataforma	No
id_nick	String	Identificador del usuario que realizo el review	No
service	Float64	Identificador de servicio asociado al restaurante	No
date	String	Fecha del comentario o review	No
platform	String	Plataforma que captura el review	No

Fuente: elaboración propia

Complaints: dataset que presenta 1.657.002 registros en dieciocho (18) variables de las cuales una (1) es seleccionada como variables de interés siendo: "Consumer complaint narrative"

Tabla 3. Descripción de variables Dataset Complaints

Variable	Tipo	Descripción	De Interés
Date received	String	Fecha en que el CFPB recibe el compaint	No
Product	String	Tipo de producto que el consumidor identifica en el complaint	No
Sub-product	String	Tipo de subproducto que el consumidor identifica en el complaint	No
Issue	String	Tema que el consumidor identifica en el complaint	No
Sub-issue	String	Subtema que el consumidor identifica en el complaint	No
Consumer complaint narrative	String	Narrativa del complaint del consumidor	Si
Company public response	String	Respuesta opcional de la compañía al complaint del consumidor	No
Company	String	Compañía sobre la cual están realizando la queja	No
State	String	Estado donde vive el consumidor	No
ZIP code	String	Código postal del consumidor	No
Tags	String	Data que facilita la búsqueda de complaints de acuerdo al consumidor	No
Consumer consent provided?	String	Indentifica si el consumidor dio consentimiento de publicar su complaint	No
Submitted via	String	Como fue cargado el complaint al CFPB	No
Date sent to company	String	Fecha en que el CFPB envió el compaint a la empresa	No
Company response to consumer	String	Como responde la compañía al consumidor	No
Timely response?	Bool	Identifica si la compañía dio una respuesta a tiempo	No
Consumer disputed?	Bool	Identifica si el consumidor discutió la respuesta de la compañía	No
Complaint ID	String	Identificación única del complaint	No

Fuente: elaboración propia

Tweets sector financiero Colombia: dataset que presenta 20.041 registros en 36 variables de las cuales tres (3) son seleccionadas como variables de interés siendo: “id”, “username” y “tweet”

Tabla 4. Descripción de variables Dataset Tweets del sector financiero de Colombia

Variable	Tipo	Descripción	De Interés
id	Int64	Identificador único del Tweet	Si
conversation_id	Int64	Identificador único de la conversación	No
created_at	String	Donde se creó el Tweet	No
date	String	Fecha de Tweet	No
time	String	Hora de Tweey	No
timezone	Int64	Zona Horaria	No
user_id	Int64	Identificador único del usuaio	No
username	String	username	Si
name	String	Nombre de usuario	No
place	String	Lugar donde se realizó el Tweet	No
tweet	String	Texto que contiene el Tweet	Si
language	String	Idioma del Tweet	No
mentions	String	Menciones en el Tweet	No
urls	String	Urls en el Tweet	No
photos	String	Fotos en el Tweet	No
replies_count	Int64	Conteo de respuestas del Tweet	No
retweets_count	Int64	Conteo de retweets del Tweet	No
likes_count	Int64	Conteo de Me gusta del Tweet	No
hashtags	String	Hashtags que contiene el Tweet	No
cashtags	String	Cashtag que contiene el Tweet	No
link	String	Links que contiene el Tweet	No
retweet	Bool	Identifica si es retweet de otro tweet	No
quote_url	String	URL del Tweet	No
video	Int64	Videos en el Tweet	No
thumbnail	String	Miniatura del Tweet	No
near	Float64	Coordenadas de cercanía	No
geo	Float64	Coordenadas geográficas	No
source	Float64	Coordenadas de origen	No
user_rt_id	Float64	Identificador único del usuario que hizo el retweet	No
user_rt	Float64	Cuenta del usuario que hizo el retweet	No
retweet_id	Float64	Identificador único del retweet	No
reply_to	String	Usuario al que le están respondiendo en el Tweet	No
retweet_date	Float64	Fecha del retweet	No
translate	Float64	Traducción del Tweet	No
trans_src	Float64	Idioma Origen del Tweet	No
trans_dest	Float64	Idioma Traducción del Tweet	No

Fuente: elaboración propia

2.2.3. Verificación de calidad de los datos.

De acuerdo la revisión inicial de los datasets, se evidenciaron los siguientes resultados de calidad:

- **TripAdvisor:** el dataset presenta 502.369 registros nulos y 203.767 duplicados en las variables de interés

- **Complaints:** el dataset presenta 1.109.010 registros nulos y 0 duplicados en las variables de interés

- **Tweets sector financiero Colombia:** el dataset presenta 0 registros nulos y 13.239 duplicados en las variables de interés, adicionalmente cuenta con 1.252 Tweets realizados

por las cuentas de los bancos que no son comentarios u opiniones de los usuarios, por lo tanto, deben ser eliminados de la vista minable.

De acuerdo con los resultados anteriores se realizará una limpieza de los registros nulos y duplicados para el desarrollo de una vista minable óptima.

2.3. Preparación de los datos

El proceso de preparación de los datos se realizó de la siguiente manera:

2.3.1. Dataset TripAdvisor.

- Se descarga un dataset con 1.258.435 de datos, correspondientes a reviews dados por consumidores de restaurantes de Perú.

- Se seleccionan únicamente las variables "review" y "score" para el desarrollo del modelo debido a que en "review" obtenemos el texto de los comentarios y en "score" la calificación del usuario correspondiente.

- Se eliminan datos con registros nulos quedando con 756.066 datos.

- Se eliminan duplicados quedando con 552.299 datos.

- De acuerdo con la distribución de la frecuencia de la variable Score se obtiene lo siguiente:

Tabla 5. Distribución de frecuencia variable Score

Score	Frecuencia	%
1	32.136	6%
2	23.764	4%
3	70.129	13%
4	157.806	29%
5	268.464	49%

Fuente: elaboración propia

- Se agruparon como "Bueno" los registros con Score 4 o 5, como "Malo" los registros con score 1 o 2 y como neutros los registros con score 3, a continuación, se muestra la distribución de frecuencia, de las categorías agrupadas:

Tabla 6. Distribución de frecuencia por Etiqueta

Etiqueta	Frecuencia	%
Bueno	426.270	77%
Malo	55.900	10%
Neutro	70.129	13%

Fuente: elaboración propia

- Se excluyen los registros clasificados como "Neutro", para trabajar únicamente con aquellos reviews que son susceptibles de obtener información que permita generar una opinión en algún sentido (Bueno / Malo) respecto a la prestación de un servicio, quedando el conjunto de datos con 482.170.

- Se observa que se tienen un 88.4% de registros con score bueno y 11.6% de registros con score malo, lo que es un gran desbalanceo entre las respuestas para cada categoría.

- Se toma la decisión de modelar con la misma cantidad de datos para cada categoría del score, para ello se selecciona al azar de la base de datos resultante en el paso anterior 40.000 registros con score bueno y 40.000 registros con score malo.

- Sobre los 80.000 registros se realiza una primera limpieza de texto, consistente en:

    - Convertir textos a minúsculas.
    - Eliminar urls.
    - Eliminar signos de puntuación.
    - Eliminar números.
    - Eliminar espacios en blanco múltiples.
    - Tokenizar por palabras individuales.
    - Eliminar tokens con una longitud menor a dos (2).

- A continuación, se muestran los siguientes descriptivos generados una vez terminada la limpieza de los datos:

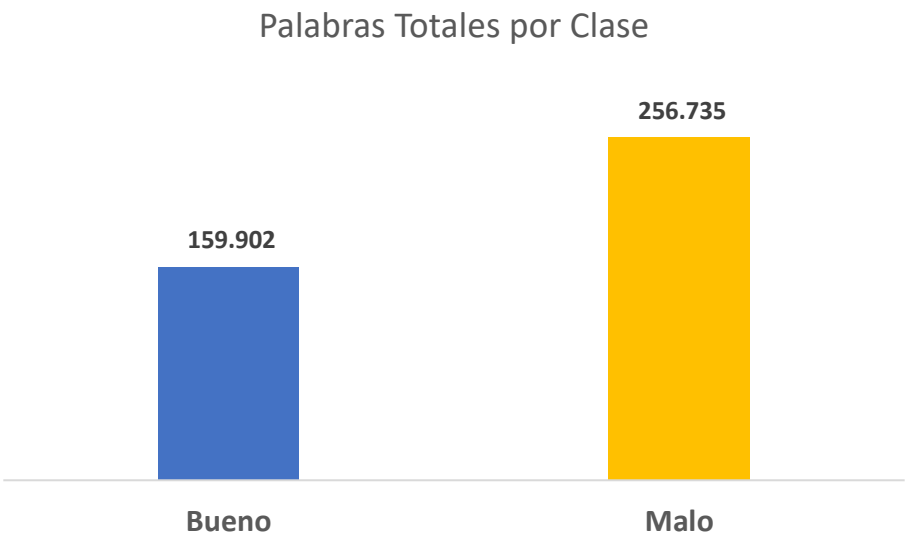

Ilustración 4. Palabras totales por clase de reviews.

Fuente: elaboración propia

- Cantidad de palabras totales por clase: encontramos que los usuarios utilizan alrededor de un 61% más de palabras para describir a un review malo respecto a la cantidad que utilizan en uno bueno.

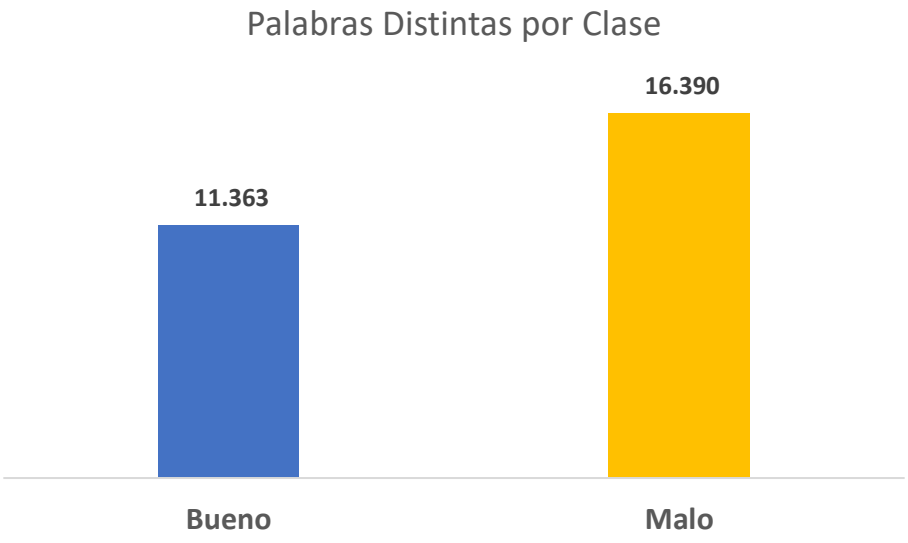

Ilustración 5. Palabras distintas por clase de reviews.

Fuente: elaboración propia

- Cantidad de palabras distintas por clase: similar al caso anterior, se tiene que los usuarios utilizan alrededor de un 44% más de palabras distintas en la redacción de un review malo en comparación a la cantidad que utilizan en uno bueno.

Tabla 7. Media y Desviación de palabras usadas por etiqueta de reviews

Etiqueta	Media	Desviación
Bueno	15.99	21.08
Malo	25.67	34.89

Fuente: elaboración propia

- En promedio, las personas utilizan casi 26 palabras para describir un comentario negativo, mientras que para uno bueno utilizan alrededor de 16 palabras. En ambos casos se observa alta variabilidad en cuanto al número de palabras que usan por comentarios, esto dado que la desviación típica es mayor al promedio de palabras.

Tabla 8. Top 5 palabras más utilizadas por categoría en reviews

Etiqueta	Token	Frecuencia
Bueno	de	6.544
	la	5.969
	el	4.471
	muy	4.453
	es	3.390
Malo	de	9.827
	la	9.762
	que	8.378
	no	8.149
	el	7.774

Fuente: elaboración propia

- Observando el top 5 de palabras utilizadas en los reviews por categoría, se evidencia que es necesaria una limpieza de Stopwords ya que son estas las que predominan.

Tabla 9. Top 5 palabras más utilizadas por categoría en reviews sin Stopwords

Etiqueta	Token	Frecuencia
Bueno	buena	2.815
	comida	2.556
	atención	2.198
	lugar	1.809
	excelente	1.560
Malo	atención	2.153
	comida	2.133
	mala	1.080
	lugar	1.078
	pollo	1.063

Fuente: elaboración propia

- De acuerdo con la nueva limpieza realizada de Stopwords se obtuvieron los resultados de la anterior tabla, observando que aún existen palabras que deben ser eliminadas tales como "comida" o "pollo" debido a que son muy particulares al sector de restaurantes y no agregarían información de utilidad en un modelo para otro tipo de servicios, sin embargo, obtenemos palabras como "buena", "atención" y "excelente" que si pueden aportar al modelo.

Ilustración 6. Nube de palabras reviews

Fuente: elaboración propia

- Nube de palabras generada sobre los 80.000 reviews, se observa que independientemente de la categoría, las palabras más utilizadas están asociadas al sector de restaurantes.

- Se realiza un análisis sintáctico, el cual consiste en la identificación de adjetivos, verbos y palabras relacionadas principalmente con comida que no suman información al modelo del sector financiero con el fin de excluirlas.

    - A continuación, se muestra un ejemplo de las salidas del análisis sintáctico:

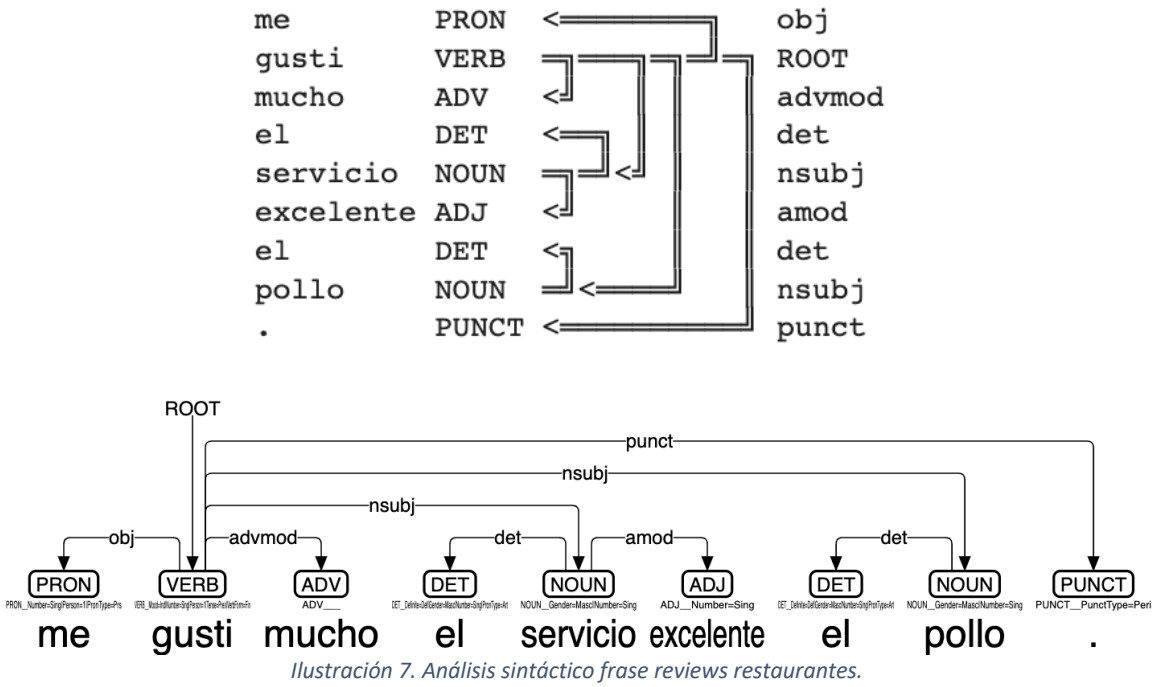

Ilustración 7. Análisis sintáctico frase reviews restaurantes.

Fuente: elaboración propia

Al ver el análisis estructural de la frase se encuentran los distintos componentes sintácticos tales como pronombres, verbos, adverbios, determinantes, sustantivos. De igual manera, es posible identificar la raíz de la frase, siendo esta de gran importancia debido al sentido que le da a la frase.

De este análisis se obtienen varias categorías sintácticas que nos pueden aportar información para el modelo de clasificación tales como: el verbo “gustar”, el adverbio “mucho”, el nombre “servicio” y el adjetivo “excelente”.

- Se continua con la revisión del mismo análisis estructural en una frase sobre el sector financiero siendo “Me gusta mucho el servicio excelentes tasas de interés”:

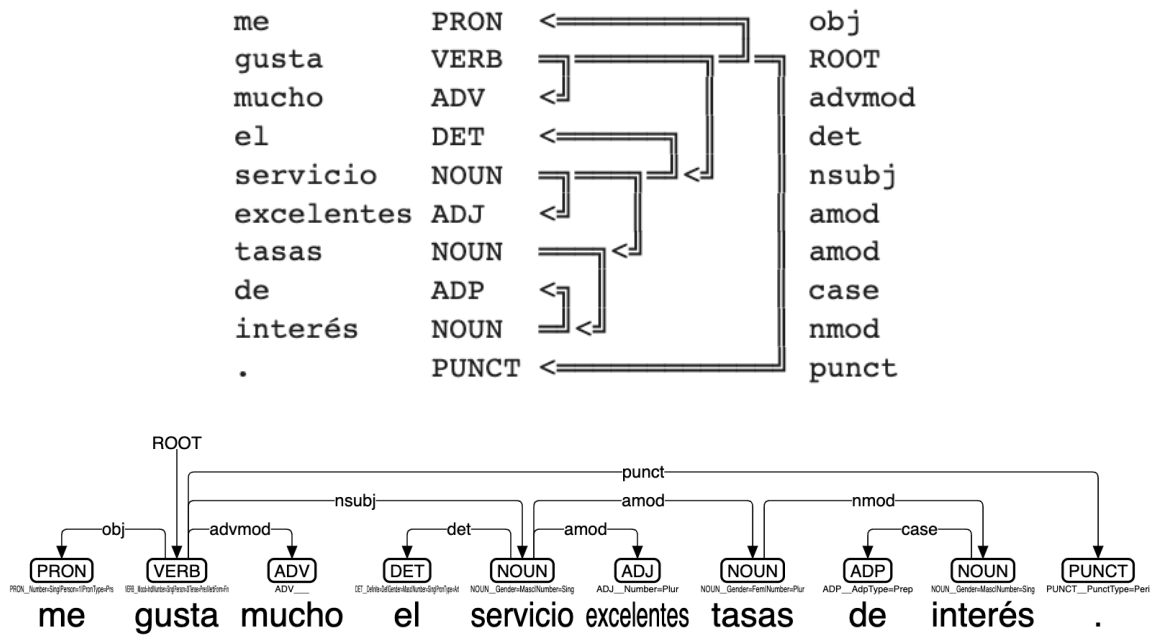

Ilustración 8. Análisis sintáctico de frase de review financiera

Fuente: elaboración propia

A partir de este análisis se puede observar que el algoritmo identifica de igual forma los componentes sintácticos de la frase sin importar el cambio de nombres de acuerdo con la frase anterior.

- Se realiza la exclusión de palabras con categoría sintáctica que no le aportan al modelo las cuales se encuentran en las siguientes listas:

Verbos	'agregar','espolvorear','ahumar','exprimir','rallar','amasar','extender','rebanar','freír','asar','gratinar','rehogar','batir','guisar','remover','colar','hervir','condimentar','hornear','salpimentar','descongelar','saltear','desmenuzar','sazonar','desmoldar','sofreír','dorar','moler','tostar','echar','pelar','trocear','empanar','pesar','untar','picar','verter','escurrir','pinchar','voltear'
Adjetivos	'aceitoso','acido','agridulce','agrio','ahumado','amargo','apetitoso','aromatico','avinagrado','azucarado','blando','caliente','chamuscado','condimentado','correoso','cremoso','crudo','crujiente','delicioso','dulce','duro','empalagoso','exquisito','fresco','grasiento','horrible','insipido','jugoso','maduro','pastoso','picante','quemado',,'rancio','rico','sabroso','salado','sazonado','suculento','verde',
Sustantivos	'brasa','leña','fogon','cocina','tomate','pollo','carne','arroz','sal','pimiena','pizza','cerdo','patata','lechugas','acelga','alcachofa','batata','berenjena','brocoli','brecol','calabacin','calabaza','cardo','cebolla','cebolleta','coles','coliflor','endivia','tomate','zanahoria','escarola','esparrago','espinaca','hinojo','judias','maiz','palmito','pepino','pimiento','puerro','remolacha',

Ilustración 9. Lista de palabras a excluir por categoría sintáctica

Fuente: elaboración propia

- Después de la limpieza de palabras que no aportan al modelo, se realiza la tokenización de los registros.

- Se divide el dataset en datos de entrenamiento (80%) y prueba (20%), de los 80.000 reviews mencionados anteriormente.

- Se realiza la vectorización de los reviews aplicando el algoritmo de TF-IDF.

2.3.2. **Tweets Sector Financiero Colombia.**

- A partir del método de Web Scraping se compilan mensajes de Twitter de un conjunto de cuentas prestablecidas, obteniendo 20.041 registros.

- Se seleccionan únicamente las variables "id", "username" y "tweet" para el desarrollo del modelo debido a que en "tweet" obtenemos el texto de los comentarios y tanto "id" como "username" para la limpieza y etiquetado de los tweets.

- Se eliminan registros duplicados quedando con 6.802 datos.

- Se eliminan Tweets de respuestas de entidades financieras a los usuarios, Tweets de futbol o relacionados con eventos deportivos patrocinados por las entidades y Tweets sin texto, por ejemplo, con únicamente emojis, urls o símbolos obteniendo 5.022 registros.

- A partir de la base de datos generada después de la limpieza, se realiza el etiquetado de los Tweets[2].

Protocolo de etiquetado

Premisas:

- Los tuits son los mismos y se comparten con los miembros del equipo, para el caso que se documenta siendo tres etiquetadores haciendo la clasificación de manera independiente.

Procedimiento:

- Las tres clases de etiquetado son:
  - "Bueno"
  - "Malo"
  - "No aplica"

- Los criterios para segmentar los Tweets en los tres grupos son:

  - **"Bueno"** : comentarios positivos sobre atención, servicios o en general de la entidad que denotara aceptación por parte del cliente.

[2] Datos disponibles en el repositorio en el link: https://gitlab.com/CAOBA-Central/productos-caoba/datalab/poc-005-cam-pqr-imagenes-fase1-develop/-/tree/develop/data/stage

- **“Malo”**: Todo comentario relacionado con una queja, reclamo o insatisfacción con la entidad.
- **“No Aplica”:** Comentarios que no reflejan un elemento de servicio o grado de insatisfacción o aceptación con la entidad financiera.

- Una vez realizada la clasificación por cada uno de los c, se evaluó la concordancia entre los etiquetadores de la siguiente manera:

  - Si los tres resultados son diferentes se elimina el tuit, debido a que habría una discordancia entre etiquetadores.

  - Si dos o más coinciden con la clasificación esta será la etiqueta para utilizar en la evaluación de los modelos.

- Después del etiquetado, consolidación y evaluación de concordancia, las etiquetas quedaron distribuidas de la siguiente forma:

  Bueno: 1.318 - 32,5%

  Malo: 2.738 - 67,5%

- Finalmente, se realiza la tokenización de los Tweets etiquetados, para aplicarlos en los algoritmos de clasificación como datos de validación.

2.3.3. Dataset Complaints.

Se realizaron las siguientes etapas de preparación de datos para el dataset de Complaints (Reseñas Financieras), el cual contiene la siguiente información:

- Se descarga un dataset con 1.657.002 registros.

- Revisamos la distribución de la frecuencia por vía de comunicación obteniendo lo siguiente:

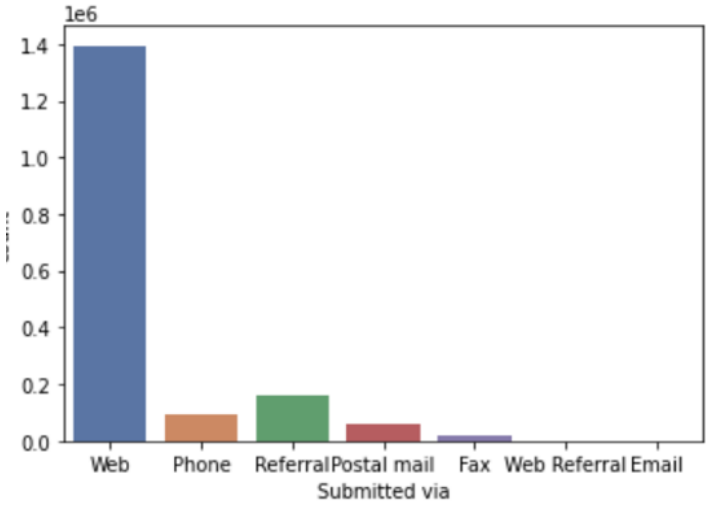

Ilustración 10. Distribución de frecuencia por vía de comunicación de complaints financieros

Fuente: elaboración propia

- En el gráfico anterior podemos identificar que más del 90% de las reseñas del dataset fueron documentadas por vía WEB.

- Se selecciona únicamente la variable "Consumer complaint narrative" para el desarrollo del modelo, debido a que obtenemos el texto de los comentarios para ser aplicados en el entrenamiento de un Word2Vec, que será usado como capa de entrada para entrenar una red neuronal.

- Se eliminan los datos con registros nulos quedando con 547.992 observaciones.

- Se tradujeron las reseñas, usando la API de GoogleTrasnlate.

- Se realiza el entrenamiento del Word2Vec 3 el cual tiene como finalidad aprender de las asociaciones de las palabras a partir de las reseñas traducidas, una vez entrenado, se produce una matriz de pesos asociados a cada palabra de vocabulario único con el objetivo de transferirlo a la red neuronal como capa de entrada de Embedding[3].

2.4. Modelamiento

En la presente etapa se revisan y seleccionan los algoritmos a trabajar en el presente proyecto, de acuerdo con el objetivo de encontrar el clasificador con mejores resultados de acuerdo con los datos de entrenamiento, prueba y validación generados anteriormente.

[3] Proceso documentado en el repositorio en el link: https://gitlab.com/CAOBA-Central/productos-caoba/datalab/poc-005-cam-pqr-imagenes-fase1-develop/-/tree/develop/deploy

2.4.1. Selección de técnicas de modelado.

Se decide implementar varios modelos con el objetivo de realizar comparaciones entre estos y definir cuál es el óptimo de acuerdo con las métricas a evaluar[4].

2.4.1.1. Regresión logística.

Como modelo base se tomó la Regresión Logística, debido a que es un algoritmo de clasificación que posee algunas ventajas tales como su facilidad de implementación, teniendo bajo costo computacional la implementación de este. El entrenamiento del algoritmo se realizó tomando como datos de entrada las codificaciones de palabras producidas por la aplicación del algoritmo Tf-Idf a los reviews después de haber pasados por un proceso de limpieza básicas, es decir eliminación de caracteres extraños, espacios en blanco y palabras vacías.

No se aplicó la segunda etapa de limpieza la cuál amplió el conjunto de "Stop-Words" incorporando la terminología común usada en el dominio del servicio de restaurantes.

2.4.1.2. Algoritmo Naive Bayes.

El primer conjunto de algoritmos que consideramos para compararlo contra el modelo base fueron los Naive Bayes, estos son un conjunto de métodos los cuales son basados en el teorema de Bayes, y que parten del supuesto que las diferentes características del conjunto de datos son independientes, lo que permite la estimación de la función de probabilidad conjunta como el producto de las probabilidades.

Si bien es cierto, el supuesto de independencia entre las distintas variables no es muy plausible, en aplicaciones del mundo real los clasificadores basados en este algoritmo han tenido un buen desempeño. Adicionalmente este es un algoritmo fácil de implementar y con un bajo costo computacional, en comparación con otros algoritmos más complejos. (Learn, 2022)

La biblioteca Sckit Learn de Python, tiene tres implementaciones del algoritmo, que se diferencian principalmente en los supuestos que realizan respecto a la función de probabilidad condicional que utilizan, las implementaciones son las siguientes:

- Naive Bayes Gaussiano, el algoritmo de clasificación asume la distribución condicional normal con media μ y varianza σ.

- Naive Bayes Multinomial, en este caso la función de distribución condicional que se asume es la multinomial.

[4] Documentación en el repositorio en https://gitlab.com/CAOBA-Central/productos-caoba/datalab/poc-005-cam-pqr-imagenes-fase1-develop/-/tree/develop/datalab

- Naive Bayes Bernoulli, con función de distribución condicional de Bernoulli multivariante.

2.4.1.3. Máquina de soporte vectorial (SVM por sus siglas en inglés).

Este es un algoritmo donde la complejidad de implementación de este es mayor que los indicados anteriormente, adicionalmente es necesario implementar estrategias de búsqueda que permitan optimizar hiper parámetros claves como los son: el kernel a usar, el valor del costo y si el kernel seleccionado no es lineal, el grado en caso de ser polinómico y la dispersión en caso de ser gaussiano.

La aproximación que sigue el algoritmo de SVM es, "construir un clasificador, … sobre un espacio de características ampliado que no se corresponde con el espacio de características asociados a los atributos de los que disponemos en el conjunto de entrenamiento…" (Berzal, 2019)

2.4.1.4. Random Forest.

Este es un algoritmo que emplea un conjunto de árboles de decisión para predecir la variable respuesta (de allí viene su nombre), esta aproximación permite reducir la varianza de la estimación, al compensar la varianza que produce cada árbol particular, este algoritmo, al igual que en el caso de SVM, es crítica la obtención de los hiper parámetros, lo que implica que también pueda tener un mayor costo computacional su implementación.

Contrario al algoritmo original en el que cada árbol vota por una clase para predecir la salida, la implementación de Sckit-learn, "combina los clasificadores, promediando las probabilidades de predicción" (Learn, 2022)

Métodos de ensamble

Los algoritmos implementados en este caso fueron Bagging y AdaBoost, ambos algoritmos parten de la premisa de construir un clasificador como combinación de varios modelos buscando con ello mejorar la precisión del resultado final.

En el caso de Bagging este genera aleatoriamente un remuestreo con reemplazamiento de los datos de entrenamiento, entrenándose el modelo para cada remuestreo de forma independiente y combinándose la predicción en función de las predicciones da cada uno de los modelos, esta agregación se puede realizar, bien sea vía el promedio o como voto de cada modelo.

En cuanto al Algoritmo AdaBoost, se basa en construir un modelo en cada iteración que realiza, la estrategia que sigue es la de mejorar la estimación del modelo anterior

en la iteración siguiente, también, "da más relevancia para la predicción de los ejemplos nuevos a los modelos que tienen un mejor comportamiento, en lugar de situarlos al mismo nivel como hace bagging" (Orallo, 2004)

2.4.1.5. Red Neuronal.

Implementamos una red neuronal secuencial usando la facilidad de word embedding, que ofrece Tensorflow por medio de Keras, la misma "transforma los tokens … en sus embedding". (Torres, Python Deep 2Learning. Introducción práctica con Keras y TensorFlow, 2020)

La configuración de la red neuronal es como se muestra a continuación:

```
# Configuración de la red

tf.keras.backend.clear_session()

clasificador_1 = Sequential()
clasificador_1.add(Embedding(vocabLength, 20, input_length=longitud_maximo_review))
clasificador_1.add(Flatten())
clasificador_1.add(Dropout(0.6))
clasificador_1.add(Dense(1, activation='sigmoid'))
```

- La capa de embedding tiene una dimensión de entrada, igual la longitud del vocabulario de los reviews de TripAdvisor, la dimensionalidad del embedding es 20 y como salida la longitud del máximo review.
- Le sigue una capa Flatten que básicamente "aplana" la entrada convirtiéndolo en un vector unidimensional.
- Continúa una capa de Dropout que evita que la red caiga en overfitting.
- Como capa de salida tiene una capa densa con una sola neurona y función de activación sigmoide (estamos en un problema de clasificación binaria)

2.4.1.6. Red neuronal con embedding preentrenado.

El último de los algoritmos que se ha implementado se trata de una red neuronal con embedding preentrenado, el texto que se usó para realizar el entrenamiento fue el proveniente de la base de datos de complaints del sistema financiero estadounidense, y el algoritmo para preentrenarlo fue Word2Vec. A continuación, presentamos la configuración de la red:

```
tf.keras.backend.clear_session()

clasificador_2 = Sequential()
embedding_layer = Embedding(vocabLength, len(vector_dimensions),
                            weights=[embedding_matrix], input_length=longitud_maximo_review,
                            trainable=False)
clasificador_2.add(embedding_layer)
clasificador_2.add(Flatten())
clasificador_2.add(Dense(20, activation='relu'))
clasificador_2.add(Dense(1, activation='sigmoid'))
```

- El argumento input_dim de la capa de embedding es la longitud del vocabulario de las reseñas de TripAdvisor, la dimensión del embedding es igual al número de filas de la matriz generada a por el algoritmo Word2Vec y que contiene las dimensiones de cada palabra que será incrustada, los pesos son los de la matriz de embedding (producidas por Word2Vec) la salida es igual a la longitud máxima de los review, los pesos no son entrenables, es decir van a ser transferidos.

- Luego viene la capa que aplana toda la información multidimensional que está en la capa de embedding para que pueda ser tratada por la red neuronal.

- Le sigue una capa densa oculta, que tiene 20 neuronas de entrada y función de activación RELU.

- De salida esta una capa densa con una sola neurona y función de activación sigmoide.

2.4.2. Construcción de los modelos.

2.4.2.1. Sintonización de hiper parámetros algoritmos.

Algorítmo	Hiperparámetro	Valor	Descripción (1)	Rango de búsqueda	Criterio de selección
Naive Bayes Gaussiano	var_smoothing	0,01	Porción de la varianza más grande de todas las características que se agrega a las varianzas para la estabilidad del cálculo	[1.e+00, 1.e-01, 1.e-02, 1.e-03, 1.e-04, 1.e-05, 1.e-06, 1.e-07, 1.e-08, 1.e-09]	Grid Search con validación cruzada = 3
Naive Bayes Bernoulli	alpha	0,1	Parámetro de suavizado aditivo	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]	Grid Search con validación cruzada = 3
Naive Bayes	alpha	1	Parámetro de suavizado aditivo	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]	Grid Search con validación cruzada = 3
SVM	Kernel	linear	kernel a asar	kernel = ['linear', 'rbf', 'poly']	Grid Search con validación cruzada = 3
	C	1	Costo	C = [1, 5, 10]	Grid Search con validación cruzada = 3
Random Forest	n_estimators	250	Número de árboles en el bosque	[50, 100, 150, 200, 250]	Grid Search con validación cruzada = 3
	max_features	12	El número de características a considerar al buscar la mejor división	[2, 4, 6, 8, 10, 12, 16]	Grid Search con validación cruzada = 3
	max_depth	12	Máxima profundidad del árbol	[4, 6, 8,10, 12, 16]	Grid Search con validación cruzada = 3
Bagging	n_estimators	350	El número de estimadores base en el conjunto.	[50, 100, 150, 200, 250, 300, 350]	Grid Search con validación cruzada = 3
	max_features	65	La cantidad de características que se extraerán de X para entrenar cada estimador base	[2, 4, 6, 8, 10, 12, 16,20, 30, 40, 50, 55, 60, 65]	Grid Search con validación cruzada = 3
AdaBoost	n_estimators		El número máximo de estimadores en los que finaliza el refuerzo	[200, 300, 400]	Grid Search con validación cruzada = 3
	learning_rate		Peso aplicado a cada clasificador en cada iteración de impulso. Una mayor tasa de aprendizaje aumenta la contribución de cada clasificador. Existe una compensación entre los parámetros learning_rate y n_estimators	[1.0, 0.9, 0.8]	Grid Search con validación cruzada = 3

(1) Fuente: (Learn, 2022)

2.4.2.2. Sintonización de hiper parámetros Red Neuronal Embedding.

Los hiper parámetros usados en el entrenamiento de esta red neuronal son los siguientes:

Los hiper parámetros optimizados son:

- Best: 0.900875 using {'batch_size': 56, 'epochs': 5}
- Best: 0.892875 using {'optimizer': 'Adamax'}
- Best: 0.892312 using {'learn_rate': 0.001}
- Best: 0.900250 using {'dropout_rate': 0.6}

Los mismos se encontraron por medio de una grilla en el siguiente espacio de búsqueda:

1. Sintonización del Batch Size y del Número de Epochs.

   batch_size = [56, 128, 256, 512]

   epochs = [5, 10, 20]

2. Sintonización del Algoritmo de Optimización.

   optimizer = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']

3. Sintonización de la Tasa de Aprendizaje.

   learn_rate = [0.001, 0.01, 0.1, 0.2, 0.3]

4. Sintonización de la Regulación por Dropout.

   dropout_rate = [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7]

2.4.2.3. Sintonización de hiper parámetros Red Neuronal Embedding Preentrenado.

Los hiper parámetros usados en el entrenamiento de esta red neuronal son los siguientes:

Los hiperparámetros optimizados son:

- 0.654937 using {'batch_size': 56, 'epochs': 30}
- Best: 0.657562 using {'optimizer': 'Adamax'}
- Best: 0.661625 using {'learn_rate': 0.001}
- Best: 0.730913 using {'dropout_rate': 0.0}
- Best: 0.633375 using {'neurons': 20}

Los hiper parámetros usados en el entrenamiento de esta red neuronal son los siguientes:

1. Sintonización del batch_size y epochs

   batch_size = [56, 128, 256]

   epochs = [30, 35, 40]

2. Sintonización del Algoritmo de Optimización

   optimizer = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']

3. Sintonización de la Tasa de Aprendizaje

   learn_rate = [0.001, 0.01, 0.05, 0.1, 0.15, 0.2]

4. Sintonización de la Regulación por Dropout

   dropout_rate = [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7]

5. Sintonización del número de neuronas de la capa oculta

   neurons = [10, 15, 20]

2.4.2.4. Word2Vec optimización de hiper parámetros.

Los hiper parámetros con los que se entrenó el Wrod2Vect son:

La mejor combinación de parámetros resultantes fue:

- 'alpha': 0.025, '
- min_count': 1,
- 'negative': 5,
- 'size': 100,
- 'window': 3

Se obtuvieron de una Grilla en el siguiente espacio de búsqueda:

- size':(100,150,200,300),
- 'min_count':(1,2,4),
- 'alpha':(0.025,0.015),
- 'window':(3,5,7),
- 'negative':(5,10,15,20)

2.5. Evaluación de modelos

A continuación, mostraremos un compendio de los resultados de las métricas de clasificación, para cada uno de los algoritmos entrenados:

Tabla 10. Resultados Accuracy.

	Accuracy		
Algoritmo	**Train**	**Test**	**Tweets**
Modelo Base	92,3%	90,5%	68,6%
Naive Bayes Gaussiano	85,7%	83,1%	70,0%
Naive Bayes Bernoulli	84,3%	83,0%	65,4%
Naive Bayes Multinomial	89,2%	88,0%	71,3%
SVM	91,6%	88,5%	69,2%
Random Forest	81,5%	80,9%	32,5%
Bagging	92,2%	73,2%	32,5%
AdaBoost	85,9%	80,9%	32,5%
Red Neuronal Sin W2V	91,8%	91,6%	69,6%
Red Neuronal con W2V	73,7%	84,5%	52,4%

Fuente: elaboración propia

De acuerdo con la tabla anterior, podemos evidenciar que los algoritmos con mejores resultados son el Naive Bayes Multinomial y Gaussiano en la base de validación (Tweets), aunque en la red neuronal sin Word2Vec presenta muy buenos resultados en Train y Test, en comparación con los Naive Bayes.

Comparando contra los objetivos de analítica establecidos, todos los algoritmos, incluyendo el modelo base, superan en Train y Test a excepción del Bagging en Test. Mientras que en validación (Tweets) los de mejor rendimiento son el Naive Bayes Gaussiano y el Multinomial.

Tabla 11. Resultados Sensitivity.

	Sensitivity		
Algoritmo	**Train**	**Test**	**Tweets**
Modelo Base	92,6%	90,8%	66,6%
Naive Bayes Gaussiano	76,1%	72,6%	76,3%
Naive Bayes Bernoulli	74,3%	72,6%	73,3%
Naive Bayes Multinomial	88,9%	87,9%	77,9%
SVM	91,6%	88,8%	75,4%
Random Forest	72,4%	71,5%	0,0%
Bagging	84,4%	59,9%	0,0%
AdaBoost	87,9%	71,5%	0,0%
Red Neuronal Sin W2V	91,6%	91,0%	72,0%
Red Neuronal con W2V	67,3%	77,5%	52,4%

Fuente: elaboración propia

En cuanto a sensitivity, los modelos con mejores rendimientos en la base de validación son el Naive Bayes Multinomial y el Gaussiano nuevamente, aunque la red neuronal sin Word2Vec tiene mejores resultados de clasificación en las bases de Train y test y no tiene tanta diferencia vs los Neive Bayes en la de validación.

Los algoritmos de Naive bayes multinomial, SVM y Red Neuronal sin Word2Vec superan los objetivos planteados en Train y test en sensitivity superando de igual manera los objetivos de validación (Tweets).

Los modelos de Random Forest, Bagging y AdaBoost no clasifican ningún valor como positivo en validación y son los de peores resultados en la base de test.

Tabla 12. Resultados Specificity.

	Specificity		
Algoritmo	**Train**	**Test**	**Tweets**
Modelo Base	92,1%	90,2%	74,0%
Naive Bayes Gaussiano	95,3%	93,4%	56,8%
Naive Bayes Bernoulli	94,3%	93,4%	49,0%
Naive Bayes Multinomial	89,6%	88,0%	57,6%
SVM	91,6%	88,6%	56,2%
Random Forest	90,7%	90,3%	100,0%
Bagging	99,9%	86,4%	100,0%
AdaBoost	84,0%	90,3%	100,0%
Red Neuronal Sin W2V	91,9%	92,3%	64,7%
Red Neuronal con W2V	80,0%	91,4%	52,4%

Fuente: elaboración propia

Todos los algoritmos superan los objetivos de analítica planteados en entrenamiento y prueba, pero en validación únicamente la superan el modelo base y la Red Neuronal sin Word2Vec.

El modelo base resulta ser el de mejores resultados en validación, teniendo en cuenta que Random forest, Bagging y Adaboos están clasificando todos los registros como negativos obteniendo una especificidad perfecta.

En especificidad los algoritmos con mejores resultados son la red neuronal sin Word2Vec, logrando mejores resultados en test versus los obtenidos en entrenamiento.

En cuanto a los modelos Random Forest, bagging y Adaboost presentan resultados perfectos en validación, sin embargo, como se evidenció en la métrica de sensibilidad al estar en 0% indica que no hacer nada o adoptar estos algoritmos terminan generando el mismo resultado.

Tabla 13. Resultados AUC.

	AUC		
Algoritmo	**Train**	**Test**	**Tweets**
Modelo Base	92,0%	91,0%	76,0%
Naive Bayes Gaussiano	93,4%	91,0%	66,6%
Naive Bayes Bernoulli	94,4%	93,2%	61,2%
Naive Bayes Multinomial	95,7%	94,8%	67,8%
SVM	96,9%	88,5%	73,6%
Random Forest	91,1%	80,9%	70,1%
Bagging	99,6%	73,1%	62,3%
AdaBoost	94,0%	93,3%	67,2%
Red Neuronal Sin W2V	96,8%	96,9%	74,8%
Red Neuronal con W2V	78,9%	93,7%	55,0%

Fuente: elaboración propia

Los resultados en AUC indican el modelo base como el de mejores resultados en validación, sin embargo, la red neuronal sin Word2Vec, presenta el segundo mejor desempeño en el resultado obteniendo mejores métricas en las bases de entrenamiento y prueba.

En general, los tres 3 mejores modelos son Naive Bayes multinomial, red neuronal sin Word2Vec y Naive Bayes Gussiano, sin embargo, el modelo que cumple con todos los objetivos de analítica propuestos es la red neuronal sin el Word2Vec siendo el modelo con resultados más integrales en cuanto a las métricas.

De acuerdo con los resultados obtenidos y la comparación de los modelos y las métricas de desempeño de los mismos, el mejor modelo seleccionado es el Naive Bayes Multinomial, principalmente por ser el de mejor resultado en la métrica de Sensibilidad, completamos nuestra sugerencia indicando que la Red Neuronal sin Word2Vec y el SVM presentan resultados muy consistentes en las métricas de desempeño versus los objetivos planteados, es importante tener en cuenta que en su implementación el costo computacional es bastante mayor al del Naive Bayes.

Validación de resultados sobre objetivo de negocio

En este apartado, se vinculan los resultados del modelo y los objetivos de negocio; recordando que el objetivo es clasificar Tweets en calificaciones positivas y negativas y a la luz del impacto que tienen en las empresas, valoraciones negativas que afectan la reputación de las mismas, se define como una métrica de mayor prioridad la Sensibilidad, debido a que esta mide que tan bueno es el modelo clasificando la variable objetivo, en este caso los tweets negativos, dado que en efecto fueron etiquetados como negativo. Aunque son importantes en materia analítica, vamos a dar menos relevancia a las métricas de Accuracy y Specificity.

Como indicamos anteriormente, la métrica que, a partir del análisis realizado, genera mayor impacto en el objetivo de negocio planteado es la sensibilidad, dado que los Tweets negativos

normalmente presentan quejas o reclamos, los cuales se deben tratar de manera proactiva para prevenir daño reputacional de la entidad y seguido de esto la pérdida de un cliente. De acuerdo con los resultados obtenidos,

La especificidad, por ser una métrica que da cuenta del porcentaje de valoraciones positivas clasificadas correctamente, pueden tener una menor importancia desde la perspectiva de negocio dado que esta involucra los comentarios positivos, los cuales no generan un riesgo para las entidades financieras o no requieren acción alguna.

3. Recomendaciones y Conclusiones

3.1. Recomendaciones

- Se recomienda generar una base léxica del sistema financiero colombiano, tomando como punto de partida información de PQRS recopilada por organizaciones que hacen parte del sector financiero con la finalidad de pre entrenar un algoritmo Word2Vec que sea la capa de embedding de una red neuronal que genere mejores resultados en próximos experimentos para mejorar los resultados de los clasificadores desarrollados en el presente proyecto.

- Se recomienda diseñar un proceso de etiquetado de los Tweets a partir de un algoritmo de entrenamiento no supervisado que permita generar agrupaciones con el fin de agilizar la obtención de estos datos, de igual manera es recomendable revisar el rendimiento de los algoritmos incluyendo la etiqueta de dato "Neutro" y comparar los resultados de su clasificación.

- De acuerdo con lo revisado en el análisis sintáctico de los comentarios, eliminando las palabras de dominios específicos, se recomienda utilizar el modelo, para clasificar los comentarios de los clientes en distintos sectores de productos y servicios.

3.2. Conclusiones

- Se pudo transferir características sintácticas de reviews de restaurantes a comentarios en Twitter sobre el sector financiero, obteniendo buenos resultados que cumplen con los objetivos de analítica que podrían ser fácilmente implementados para identificar clientes inconformes con productos y servicios de las entidades.

- De acuerdo con los resultados obtenidos se encontraron cuatro (4) algoritmos, los cuales son Naive Bayes Multinomial, Gaussiano, SVM y Redes Neuronales sin Word2Vec, que superaron las métricas analíticas objetivo planteadas al inicio del proyecto.

- El modelo base de clasificación, el cual tiene un bajo costo computacional, presenta muy buenos resultados comparado con modelos más complejos, como los Random Forest, Bagging y Adaboost, que requieren de mayor trabajo en su construcción e implementación.

- Las redes neuronales sin Word2Vec presentan resultados consistentes en todas las métricas y bases en las cuales se les aplican los algoritmos para clasificación a partir de procesamiento natural de lenguaje. Sobre todo, en las bases de entrenamiento y prueba.

- En las redes neuronales con capa de embedding entrenada con el Word2Vec basado en los datos de complaints financieros de Estados Unidos, presentó resultados de las métricas de desempeño inferiores a lo que se podría esperar de una red con ese tipo de características.

- Los algoritmos más sencillos de implementar y con menor costo computacional, presentan consistentemente mejores resultados que los de mayor complejidad y costo computacional.

- Los usuarios utilizan 61% más de palabras para expresar una opinión negativa vs una positiva dado que normalmente describen algún evento desagradable con un producto o servicio.

- A partir de los datos que se extrajeron de Twitter, se evidencia que la personas se manifiestan más para los casos negativos que para los casos positivos y esto afectó en cierto punto la experimentación, dado que se presenta un desbalanceo importante en los tuits, lo anterior redunda en un punto de corte diferente a 0.5 en los clasificadores para obtener mejores resultados.

Bibliografía

Chapman, P. C. (2000). Crisp-dm 1.0: Step-by-step data mining guide. *SPSS Inc*, 14.

Azevedo, A., & Santos, M. (2008). KDD, SEMMA and CRISP-DM: a parallel overview. *Instituto Politécnico do Porto. Instituto Superior de Contabilidade e Administração do Porto*, 4.

Euromonitor. (2021). *Digital Disruptors: The Global Competitive Landscape of Social Media.* Euromonitor Internacional.

Guidance. (2012). Las pérdidas de clientes pueden ascender hasta el 27% por los comentarios en Internet. *PuroMarketing*.

Lázaro. (2021). *Clustering & Sentiment Analysis*. Obtenido de Kaggle: https://www.kaggle.com/code/lazaro97/clustering-sentiment-analysis

CFPB. (2022). *Consumer Complaint Database*. Obtenido de Consumer Financial Protection Bureau: https://www.consumerfinance.gov/data-research/consumer-complaints/#download-the-data

PUJ. (2022). *¿Qué es un sistema de PQRSF?* Obtenido de Pontificia Universidad Javeriana Cali: https://www.javerianacali.edu.co/peticiones-quejas-reclamos-sugerencias-y-felicitaciones

Learn, S. (2022). *Scikit Learn*. Obtenido de Scikit Learn: https://scikit-learn.org/stable/modules/naive_bayes.html

Berzal, F. (2019). Redes Neuronales & Deep Learning. En F. Berzal, *Redes Neuronales & Deep Learning* (pág. 109). Granada: Fernando Berzal.

Orallo, J. H. (2004). Intriducción a la minería de Datos. En J. H. otros, *Intriducción a la minería de Datos* (pág. 489). España: Pearson - Prentice Hall.

Torres, J. (2020). Python Deep 2Learning. Introducción práctica con Keras y TensorFlow. En J. Torre, *Python Deep 2Learning. Introducción práctica con Keras y TensorFlow* (pág. 286). Colomboa: Marcombo.

Patil, R., & Kolhe, S. (2022). *Supervised classifers with TF-IDF features for sentiment analysis of Marathi tweets*. Obtenido de https://link-springer-com.ezproxy.javeriana.edu.co/content/pdf/10.1007/s13278-022-00877-w.pdf

Dehong, M., Sujian, L., Xiaodong, Z., & Houfeng, W. (2017). *Interactive Attention Networks for Aspect-Level Sentiment Classification*. Obtenido de MOE Key Lab of Computational Linguistics: https://arxiv.org/pdf/1709.00893v1.pdf

Torres, J. (2020). *Pyhton Deep Learning. Introducción práctica con Keras y TenserFlow 2.*

Geron, A. (2019). *Hands-On machine Learning with Scikit-Learn, Keras & TensorFlow. Concepts, Tools, and Techniques to Build Intelligent Systems.* O'REILLY.

Goldberg, Y. (2017). *Neural Network Methods for Natural Language Processing.* Greame Hirst, Series Editor.

CAOBA. (2022). *Repositorio Modelo Clasificación de PQRS*. Obtenido de GitLab: https://gitlab.com/CAOBA-Central/productos-caoba/datalab/poc-005-cam-pqr-imagenes-fase1-develop