UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

DISEÑO Y DESARROLLO DE UN PROTOTIPO DE ANALÍTICA DE TEXTO PARA LOS CLIENTES DE RETAIL DE UN HELPDESK ESPECIALIZADO EN SERVICIO AL CLIENTE

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

TOMÁS ANDRÉS GUZMÁN JARAMILLO

PROFESORA GUÍA:
Carolina Segovia Riquelme

MIEMBROS DE LA COMISIÓN:
Blas Duarte Alleuy
Juan Romero Godoy

SANTIAGO DE CHILE
2024

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL
POR: TOMÁS ANDRÉS GUZMÁN JARAMILLO
FECHA: 2024
PROF. GUÍA: CAROLINA SEGOVIA RIQUELME

DISEÑO Y DESARROLLO DE UN PROTOTIPO DE ANALÍTICA DE TEXTO PARA LOS CLIENTES DE RETAIL DE UN HELPDESK ESPECIALIZADO EN SERVICIO AL CLIENTE

En el contexto de un mundo globalizado y saturado de información, las Tecnologías de la Información (TIC) desempeñan un papel crucial para la inserción efectiva de un Estado. Estas herramientas son fundamentales para potenciar el desarrollo económico y social al simplificar procesos y mejorar el acceso a la información. En este contexto, surge la iniciativa de "Adereso", una startup chilena establecida en 2014 que se autodenomina como proveedor de Software como Servicio (SaaS), operando en el sector de las TIC, específicamente en la subindustria de "Application Software". La cual tiene entre sus productos, Adereso Desk, un helpdesk que omnicanaliza los canales de atención al cliente digital de sus empresas usuarias.

Este proyecto tiene como objetivo crear un prototipo de Analítica de Texto mediante técnicas de Procesamiento del Lenguaje Natural. Se enfoca en analizar las interacciones cliente-agente de las empresas usuarias en el sector del retail de Adereso, abordando tres dimensiones clave: conteo de unigramas, bigramas y trigramas frecuentes, tipificación y análisis del Sentimiento de los Tickets. Para guiar el desarrollo, se adoptó la metodología CRISP-DM Lite, prescindiendo de un despliegue de la analítica. Se empleará un modelado descriptivo para la dimensión de conteo, mientras que para las otras dimensiones se implementarán modelos predictivos.

Para la dimensión de conteo de unigramas, bigramas y trigramas frecuentes, se desarrollaron 3 modelamientos para extraer los 20 unigramas, bigramas y trigramas más frecuentes. Los cuales fueron graficados y revelan los valiosos insights que se pueden obtener a través de este análisis. En el caso de la dimensión de tipificación de tickets, se empleó un modelo bifásico. En la primera fase, se utilizó un modelo no supervisado, LDA, para determinar el número óptimo de tópicos en función del corpus, caracterizando y asignando cada uno de ellos a las conversaciones procesadas del helpdesk. La segunda fase consistió en la construcción de dos clasificadores: uno basado en una red neuronal recurrente LSTM y otro en el modelo transformer distilBERT. Este último demostró un rendimiento óptimo al equilibrar las clases, alcanzando un accuracy del 95.16 %, precision del 95.00 %, recall del 95.00 %, y f1-score del 95.00 %. Finalmente, en la dimensión de Análisis de Sentimiento, se empleó el modelo GPT-3.5 Turbo a través de su API. Comparando sus predicciones con un etiquetamiento manual realizado por el estudiante, se observó un rendimiento destacado con un 95.00 % de accuracy, 96.00 % de precision, 73.00 % de recall y 80.00 % de f1-score.

Por último, se ha dejado una propuesta de visualizaciones para integrar en un futuro en la plataforma, con el objetivo de que esto tenga un real impacto en los clientes de retail de Adereso. Junto con ello, se proponen mejoras y expansiones del análisis propuesto en el actual proyecto.

A mi familia y amigos,
simplemente muchas gracias.

Saludos

Agradecimientos

En primer lugar, me gustaría agradecerle a las personas mas importantes de mi vida: Mi abuelita, Tío Jaramillo, Tía Lorenita, Jara, Santiaguito, Mi polola Vannea y por último, a la persona más importante de mi vida, Mi mamá. La persona que siempre ha dado todo por mí, muchas gracias viejita. Espero que estés orgullosa del hijo que has criado. Sin el apoyo de ustedes nada, de las pocas cosas que he logrado en mi vida, sería posible

En segundo lugar, quiero agradecerles a mis amigos de la enseñanza media: Al Toba, Gonzalo, Mauri, Tapia, Carlos, Franco, Johan, Maxi y mi primo Matías. Me gustaría agradecerles por su apoyo y buena onda de siempre, siento que tenemos una amistad entrañable, de las que no se tienen hoy en día.

En tercer lugar, quiero darles una mención a mis ex compañeros de DIM de la Santa María: Martín, Fabián, Pipa y Tata. Son unos secos, para mí son un ejemplo y me gustaría que en base a trabajo personal, lograr parecerme aunque sea un poco a ustedes. Los admiro un montón.

Ya para ir terminando, me gustaría dar un agradecimiento a las personas con las cuales compartí en la u, universidad que siempre soñé estar y que por cosas de la vida no pude entrar de inmediato después de salir de enseñanza media. Juanix, Juan, Exequiel y Diego, quiero agradecerles por la calidad de persona y estudiante que son, sin duda que el rendimiento académico que mostré en la facultad se debe a la gran disciplina que ustedes me inculcaron desde la posición de amigo, muchas gracias por obsequiarme su hermosa amistad.

Y por último, me gustaría hacer una mención a mis mascotitas: Atenea y Julietita, las quiero con toda mi alma, cuando estoy sin ustedes, las extraño un montón y gracias a su amor incondicional han sido un pilar emocional fundamental en estos 6 años en mi paso por la universidad.

A todos los mencionado, los quiero mucho y tienen un lugar en mi corazón.

Tabla de Contenido

Índice de Tablas

Índice de Ilustraciones

Capítulo 1

Antecedentes Generales

En los últimos años, la industria de software en Chile, perteneciente al campo de las Tecnologías de la Información (TIC), ha experimentado un crecimiento constante. Esto ha ampliado las posibilidades de comunicación, especialmente en el ámbito digital. La importancia de esta industria se ha vuelto notable durante el período en el que comienza estallido social y acaba la pandemia en Chile, ya que muchas empresas se vieron obligadas a pasar de un modelo de negocios tradicional a uno cada vez mas digital.

Esta industria, cada vez más indispensable, se centra en el sector comercial con el objetivo de mejorar la competitividad de aquellos que necesitan de sus servicios. Desde finales de los años 90, el Estado chileno ha mostrado preocupación por generar políticas públicas que fomenten el desarrollo, la adopción y el uso de las tecnologías por parte de actores tanto públicos como privados.

En un mundo globalizado y con un exceso de información, las Tecnologías de la Información son clave para que un Estado se inserte de manera efectiva. Estas herramientas son fundamentales para incrementar los niveles de desarrollo económico y social, ya que simplifican procesos y mejoran el acceso a la información. El sector industrial de las Tecnologías de la Información (TIC) en Chile se caracteriza por una serie de atributos sobresalientes. En primer lugar, se caracteriza por su enfoque continuo en la innovación y la creatividad, lo que no solo abre nuevas vías de comunicación, sino que también impulsa el desarrollo de soluciones tecnológicas avanzadas. Además, las TIC desempeñan un papel fundamental en el ámbito educativo, enriqueciendo la experiencia educativa mediante enfoques dinámicos y accesibles, que facilitan la adquisición de conocimientos. Estrechamente vinculado con la era digital, este sector desencadena y lidera la transformación digital en diversas industrias a través de su conexión intrínseca con internet y la informática. Por otro lado, las TIC ejercen una influencia significativa en varias disciplinas de las ciencias humanas, como la sociología y las teorías de organización y gestión, al proporcionar capacidades avanzadas de recopilación, análisis y presentación de datos, transformando así la investigación y la toma de decisiones en estos campos.

En este escenario, surge la iniciativa de **"Adereso"**, una startup chilena que se estableció 2014 y se autodenomina como un proveedor de Software como Servicio (SaaS). Esta empresa opera dentro del sector de las Tecnologías de la Información (TIC) y, específicamente, se enmarca en la subindustria de "Application Software". Actualmente, Adereso cuenta con un equipo de más de 40 personas y su liderazgo recae en Camilo López, quien ejerce el cargo de Director Ejecutivo (CEO).[Adereso]

Desde su fundación en 2014, la empresa ha ampliado su presencia en varios países de América

Latina, brindando un servicio basado en un modelo de negocios B2B2C. Este enfoque permite a las empresas transformar sus procesos de Servicio de Atención al Cliente digital (SAC) en experiencias omnicanal, enfocada en diversos canales digitales, como Facebook, WhatsApp, Twitter, Instagram, entre otros.
Para una mejor comprensión y visualización de los actores involucrados en el modelo de negocios B2B2C propuesto por Adereso, se adjunta un mockup:

Figura 1.1: Actores del modelo de negocios de Adereso

Como se puede apreciar en la figura 1.1, Adereso actúa como una empresa intermediaria entre las empresas usuarias de su plataforma y sus respectivos clientes. Es importante destacar que las empresas usuarias deben disponer de canales digitales dedicados a la atención al cliente, y las interacciones en las que Adereso funge como intermediario son exclusivamente de naturaleza digital.
Para brindar un servicio eficiente, Adereso se respalda en su plataforma en la nube denominada Adereso Desk, un helpdesk omnicanal que consolida la interacción con los usuarios mediante la integración con diversos canales. Cuando un ejecutivo de Adereso Desk se comunica con un cliente o cuando un cliente se dirige a una empresa usuaria a través de los canales de atención digitales, se inicia lo que comúnmente se conoce como un ticket, una suerte de conversación en términos coloquiales. Este enfoque posibilita a las empresas usuarias de Adereso proporcionar un servicio postventa digital eficaz y ágil, adaptándose a las necesidades de sus clientes. Además, Adereso ofrece herramientas de automatización, como respuestas rápidas para abordar reclamos específicos o gestionar mensajes fuera del horario de servicio, entre otras funcionalidades.
En la actualidad, Adereso atiende a aproximadamente 150 empresas.
Entre los clientes destacados se encuentran empresas de retail de renombre, tales como Walmart, Falabella, Ripley y Paris. Sin embargo, también colabora estrechamente con empresas pertenecientes a las industrias de telecomunicaciones y servicios financieros.

Capítulo 2

Descripción del proyecto y justificación

En la actualidad, **Adereso Desk** cuenta con una sólida implementación de analítica que se enfoca en métricas operativas para evaluar la eficiencia del proceso de atención al cliente digital. Sin embargo, esta analítica no profundiza en el análisis del contenido textual de los tickets, un aspecto que Adereso considera crucial para mejorar la comprensión y calidad de las interacciones con los clientes de Adereso Desk.
La analítica existente se centra en métricas como:

- **Tickets Totales:** representa la totalidad de tickets generados.
- **Abordados:** indica la cantidad de tickets que fueron atendidos por algún agente.
- **Ignorados:** refiere a la cantidad de tickets que fueron desestimados por algún agente.
- **Gestionados:** engloba la cantidad total de tickets que fueron tanto abordados como ignorados por algún agente.
- **Abordados en SLA:** se relaciona con la cantidad de tickets atendidos por algún agente dentro del tiempo específico establecido por la empresa usuaria de Adereso Desk. Por ejemplo, en empresas usuarias del sector retail, el tiempo SLA o tiempo de primera respuesta suele fijarse en 5 minutos. En este contexto, los tickets abordados en SLA serían aquellos respondidos antes de 5 minutos después de su creación en la plataforma.
- **Minutos de Abordaje Promedio:** representa el promedio de minutos en los cuales los tickets fueron atendidos.
- **Horas de Atención Promedio:** indica el promedio de horas durante las cuales los agentes estuvieron respondiendo tickets.

Estas métricas son esenciales para evaluar el rendimiento operativo, pero no proporcionan insights detallados sobre el contenido textual de los tickets.

Este enfoque exclusivamente operacional es lo que a Adereso como empresa le gustaría expandir, ya que brindando una capa adicional de análisis textual a los tickets se podría mejorar la comprensión y la calidad de las interacciones con clientes de los usuarios de Adereso Desk. Asimismo, es evidente que la cantidad de datos no estructurados o textuales generados mensualmente por Adereso es considerable, superando los 2 millones de tickets. Este volumen

sustancial respalda la viabilidad del desarrollo de una analítica textual más avanzada.
Remontándonos a los primeros días de Adereso y la concepción de su propio nombre, encontramos una razón adicional para la pertinencia de esta iniciativa. 'Adereso' surge como un acrónimo de la frase 'Análisis DE REdes SOciales', revelando que la idea de análisis de texto ha estado presente desde sus inicios. Sin embargo, esta iniciativa nunca se materializó en un producto o funcionalidad robusta.

En los niveles directivos de Adereso, se sostiene la creencia de que la implementación de una analítica textual en Adereso Desk generaría beneficios sustanciales. Un detallado conteo de las palabras más frecuentes proporcionaría valiosos conocimientos acerca de las principales inquietudes y temas de los usuarios de Adereso. La clasificación automática de texto simplificaría la gestión de tickets al organizarlos en categorías predefinidas como reclamos, consultas y compras, eliminando así la actual tarea manual de clasificación realizada por los agentes usuarios de las licencias de Adereso. Además, al incorporar una dimensión de análisis del sentimiento, se lograría una evaluación cuantitativa de las actitudes expresadas en los tickets, brindando una comprensión profunda de la satisfacción del cliente y facilitando la identificación proactiva de áreas de mejora. Estas dimensiones integradas no solo enriquecerían la comprensión de las interacciones con los usuarios, sino que también abrirían nuevas posibilidades para la mejora continua y la personalización de los servicios ofrecidos por Adereso Desk.

Es en este contexto que surge la demanda del Área de Desarrollo y Producto de Adereso para el proyecto: **"Desarrollar un prototipo de Analítica de Texto con tres dimensiones principales: Conteo de palabras frecuentes, Tipificación general y Análisis de Sentimiento de los Tickets"**.
La propuesta fue aceptada por el estudiante encargado del proyecto, quien acordó implementarlo inicialmente para los clientes de la industria del Retail de Adereso. Esta elección se fundamenta en la ausencia de un trabajo similar previo, lo que podría complicar la tipificación de tickets de industrias distintas sin historial de tipificaciones. Con más del 53 % de la cartera de clientes provenientes de la industria del retail, este enfoque asegura un impacto significativo en la organización. Además, en una decisión estratégica, el estudiante optó por ampliar la dimensión de conteo de palabras frecuentes no solo a palabras individuales (unigramas), sino que también incluirá bigramas y trigramas en el análisis.

En resumen, este proyecto se centrará en el diseño y desarrollo de una analítica textual para los clientes de Retail de Adereso, la cual tendrá 3 dimensiones:

- **Conteo de Unigramas, Bigramas y Trigramas más Frecuentes:** Se explorará exhaustivamente el contenido textual de los tickets, analizando no solo palabras individuales (unigramas) sino también combinaciones de dos y tres palabras (bigramas y trigramas) para obtener una visión completa y detallada de los patrones lingüísticos relevantes.
- **Tipificación de los Tickets:** Se implementará una clasificación automatizada que asignará etiquetas a los tickets, simplificando la gestión y permitiendo una organización eficiente. Esta automatización reducirá significativamente el tiempo y los recursos dedicados a la tipificación manual, eliminando horas hombre de los ejecutivos y mejorando la eficiencia del proceso.
- **Análisis del Sentimiento de los Tickets:** Se llevará a cabo una evaluación cuantitati-

va de las actitudes expresadas en los tickets, categorizándolas en una escala de Negativo, Neutro y Positivo. Este análisis proporcionará información valiosa sobre la percepción general de los clientes y facilitará la identificación proactiva de áreas de mejora.

Capítulo 3

Objetivos

3.1. Objetivo General

"Desarrollar un prototipo de Analítica de Texto para clientes del sector Retail de Adereso mediante técnicas de Procesamiento del Lenguaje Natural. El objetivo es mejorar la eficiencia y calidad del servicio al cliente digital, facilitando decisiones informadas para ejecutivos a través del análisis de las interacciones cliente-agente"

3.2. Objetivos Específicos

- **Gestionar la extracción, carga y transformación de los datos.**
  Implementar un proceso ETL eficiente para obtener y preparar datos de los tickets de Adereso Desk, con el objetivo de asegurar un dataset limpio y apropiado para el análisis textual.

- **Realizar un preprocesamiento de los datos textuales coherente con la realización de tareas de Procesamiento del Lenguaje Natural.**
  Aplicar técnicas de preprocesamiento de texto para mejorar la calidad de los datos de entrada a los futuros modelos del prototipo.

- **Desarrollar los modelos correspondientes para cada una de las dimensiones de este prototipo.**
  Crear modelos efectivos para cada dimensión del prototipo, abordando el conteo de palabras frecuentes, la tipificación automática de tickets y el análisis de sentimiento.

- **Definir correctamente un modelo de datos para cada una de las dimensiones.**
  Establecer modelos de datos robustos y adecuados para cada dimensión del prototipo, garantizando una representación estructurada y comprensible de la información.

- **Diseñar una visualización intuitiva en la cual se muestren los resultados de la analítica.**
  Crear una visualización que presente de manera efectiva los resultados de la analítica de texto, facilitando la interpretación y la futura toma de decisiones por parte de los usuarios finales del sistema.

- **Generar un instructivo para guiar los pasos de la implementación de este prototipo**
  Crear un manual de instrucciones para la correcta implementación de la dimensión de analítica de texto propuesta en este trabajo de título

Capítulo 4

Alcances

El prototipo de analítica de texto que se desarrollará en este trabajo de título es el primero que se ha realizado alguna vez en esta empresa. Es por esta razón, que para la primera iteración de este tipo de analítica sólo se tomará en cuenta los clientes de Retail de Adereso. Dado que este sector industrial es el más numeroso en la cartera de clientes de la empresa.

Por otro lado, el entregable de este proyecto será el conjunto de modelos desarrollados por el estudiante para cada una de las dimensiones propuestas anteriormente y un conjunto de visualizaciones de cada de las dimensiones del prototipo .

El alcance de este proyecto, no incluye la implementación de esta analítica en la plataforma. El Tutor del estudiante, manifestó que esto quedaría en manos de alguno de sus desarrolladores web.

Capítulo 5

Marco Conceptual

En esta sección se proporcionará el marco conceptual de este trabajo de título, con el objetivo de contextualizar tanto conceptos generales como específicos de este proyecto. Los tópicos a abordar son:

5.1. Aprendizaje Automático no supervisado

El Aprendizaje Automático no supervisado o también conocido como Machine Learning(ML) no supervisado, es un método de Aprendizaje Automático que se utiliza para analizar y agrupar en clústeres conjuntos de datos sin etiquetar de manera previa. La tarea de este tipo de algoritmos es descubrir patrones ocultos sin necesidad de la intervención humana. Esta capacidad lo convierte en una solución ideal para tareas como: Análisis de datos exploratorios, segmentación de clientes, etc. Algunas técnicas de este tipo de aprendizaje son: Clustering, Análisis de Temas(Topic Modeling) y Reducción de dimensionalidad.[IBM]

5.1.1. Modelado de Temas (Topic Modeling)

El Modelado de Temas es un técnica de aprendizaje no supervisado, que permite identificar los temas principales que están presentes en un gran conjunto de documentos, sin necesidad de conocer previamente las categorías o etiquetas. Los documentos se representan como una combinación de temas, y cada tema está compuesto por una distribución de palabras que son más relevantes para ese tema en particular.En este contexto, el modelado de temas determina tanto los patrones de palabras como las frecuencias de palabras dentro de un documento para identificar una lista de temas o grupos de temas en ese documento. Es útil para analizar y clasificar una gran colección de documentos o textos en función de los temas extraídos. Por lo expuesto anteriormente, es una técnica que encaja perfectamente con la realización de primer etiquetado en la dimensión de “tipificación de tickets”.[Grisales, 2022]

5.1.1.1. Asignación Latente de Dirichlet (LDA)

La asignación latente de Dirichlet es un modelo generativo probabilístico utilizado en el Procesamiento de Lenguaje Natural (NLP) y análisis de texto. La principal idea detrás de este algoritmo es que cada documento es un corpus en sí, el cual se puede ver como una mezcla de diferentes temas, y cada uno de ellos se representa como una distribución de palabras. El modelo asume que hay un número fijo de tópicos en el corpus, y cada palabra en un documento proviene de uno de estos tópicos. Sin embargo, estos tópicos son desconocidos y se consideran “ocultos”.[Blai, 2003] Este algoritmo utiliza una distribución Dirichlet para

modelar dicha mezcla de temas en cada documento y la distribución de palabras en cada tópico. Los parámetros de esta distribución se estiman a partir de los datos o documentos de entrada, lo que permite a LDA encontrar de forma automática los tópicos y sus distribuciones en el corpus.

5.2. Aprendizaje Profundo

El aprendizaje profundo constituye un conjunto de técnicas en el ámbito del aprendizaje automático, fundamentadas en el uso de redes neuronales artificiales. Estas redes, inspiradas en el cerebro humano, se componen de capas sucesivas que procesan la información de manera secuencial [Oracle]. La construcción de redes neuronales profundas es la piedra angular del aprendizaje profundo, donde múltiples capas de unidades de procesamiento, denominadas neuronas, permiten aprender representaciones jerárquicas de los datos, potenciando la capacidad para capturar características complejas y abstracciones.

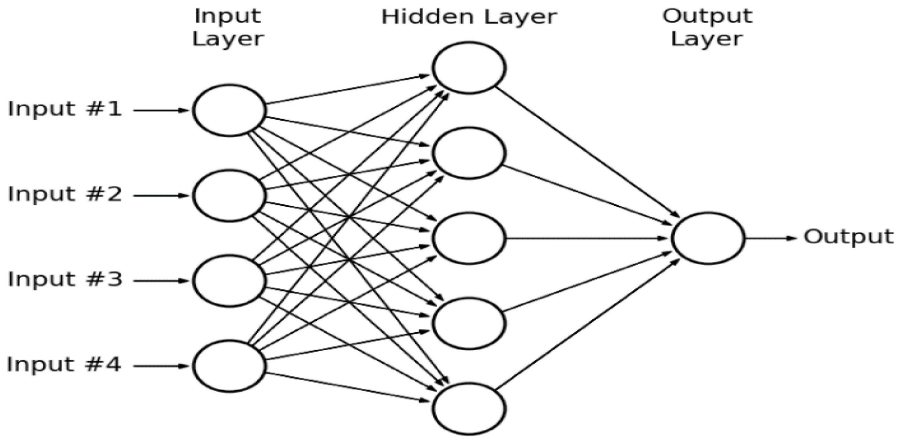

Figura 5.1: Ejemplo de arquitectura de una red neuronal perceptrón multicapa.

Por ejemplo, en la figura 5.1 se representa un Un perceptrón multicapa (MLP) con 4 entradas, 5 neuronas ocultas y una salida. Las entradas se procesan en la capa oculta, que aprende patrones complejos. La salida refleja las predicciones. Los pesos ajustables entre capas se adaptan durante el entrenamiento para mejorar la precisión. La arquitectura jerárquica del MLP se visualiza en la imagen, destacando la conectividad entre capas y su capacidad para representar relaciones complejas.

El aprendizaje profundo ha revolucionado el campo del aprendizaje automático, posibilitando logros que anteriormente se consideraban inalcanzables. Esta técnica se emplea ampliamente en diversas aplicaciones, entre las que destacan el reconocimiento de imágenes, el reconocimiento de voz, el procesamiento del lenguaje natural y la traducción automática.

5.2.1. Redes Neuronales Recurrentes(RNN)

Las Redes Neuronales Recurrentes (RNN) son un tipo de arquitectura de redes neuronales diseñadas para trabajar con datos secuenciales, donde la información tiene una dependencia temporal. A diferencia de las redes feedforward convencionales, las RNN tienen conexiones retroalimentadas que les permiten mantener y utilizar información sobre eventos anteriores en la secuencia.[Schmidt, 2019]

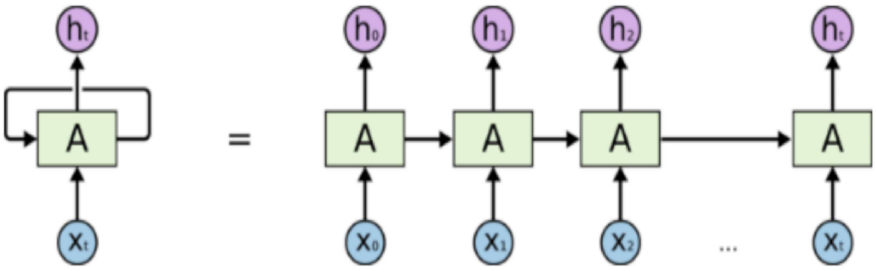

Figura 5.2: Ejemplo de arquitectura de una red neuronal recurrente.

La estructura básica de una RNN consiste en nodos (neuronas) conectados entre sí en una secuencia temporal. Como se puede ver en la figura 5.2, cada nodo toma como entrada la información actual y la información proveniente de nodos anteriores. De esta manera, las RNN pueden capturar patrones y dependencias temporales en datos secuenciales.

Para entrenar RNN, se utiliza el proceso de Retropropagación a Través del Tiempo (BPTT). Este proceso adapta el algoritmo de retropropagación clásico para manejar la estructura de tiempo en la red. Se realiza una propagación hacia adelante a través de la secuencia, y luego se calculan gradientes y se realiza la retropropagación para actualizar los pesos de la red.
Las RNN tienen el desafío del problema de desvanecimiento y explosión del gradiente, donde los gradientes pueden hacerse muy pequeños o muy grandes a medida que se retropropagan a través de la secuencia. Esto puede dificultar el aprendizaje efectivo de dependencias a largo plazo. Para abordar los problemas de las RNN tradicionales, se han propuesto variantes más avanzadas, como las Long Short-Term Memory (LSTM) y las Gated Recurrent Unit (GRU). Estas variantes incorporan mecanismos de puertas y memoria a largo plazo, lo que les permite aprender dependencias a largo plazo de manera más efectiva.
Las RNN se han utilizado en una variedad de aplicaciones, incluyendo el procesamiento del lenguaje natural, la generación de texto, la traducción automática y la predicción de series temporales.

5.2.1.1. Red neuronal recurrente variante LSTM

En la misma línea, La variante LSTM (Long Short-Term Memory) es una RNN que ha demostrado ser muy eficaz para el procesamiento de datos secuenciales de largo alcance. Las LSTM tiene un estado de célula y tres puertas que les permiten controlar cómo la información de entradas anteriores se incorpora a la salida de la red.[Anishnama, 2023]

Estado de Célula: El estado de célula es la memoria de la unidad LSTM. Puede almacenar

y recuperar información a lo largo de secuencias largas. El estado de célula es controlado por un conjunto de compuertas, lo que le permite agregar o eliminar información según sea necesario.

Compuerta de Olvido: La compuerta de olvido determina qué información del estado de célula debe descartarse o conservarse. Toma como entrada el estado de célula anterior y la entrada actual, y produce un valor entre 0 y 1 para cada elemento en el estado de célula, donde 0 significa "olvidar por completo" y 1 significa "conservar por completo".

Compuerta de Entrada: La compuerta de entrada determina qué nueva información debe almacenarse en el estado de célula. También toma el estado de célula anterior y la entrada actual como entrada y produce un valor entre 0 y 1 para cada elemento en el estado de célula, indicando cuánta información nueva debe agregarse.

Compuerta de Salida: La compuerta de salida controla qué información se envía como salida de la unidad LSTM. Toma el estado de célula anterior y la entrada actual, y en función de estos, produce la salida de la unidad y actualiza el estado de célula.

A continuación se muestra la arquitectura de una red neuronal recurrente variante LSTM:

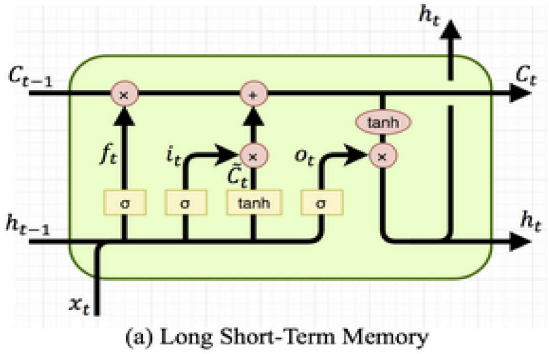

Figura 5.3: Arquitectura de una red neuronal recurrente variante LSTM.

A continuación, se presentan las ecuaciones principales de una unidad LSTM:

$$f_t = \sigma(W_f * [h_{t-1}, x_\dagger] + b_f) \tag{5.1}$$

$$i_t = \sigma(W_i * [h_{t-1}, x_\dagger] + b_i) \tag{5.2}$$

$$\hat{C}_t = tanh(W_c * [h_{t-1}, x_\dagger] + b_c) \tag{5.3}$$

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{5.4}$$

$$o_t = \sigma(W_o * [h_{t-1}, x_\dagger] + b_o) \tag{5.5}$$

$$h_t = o_t * tanh(C_t) \tag{5.6}$$

Interpretando las ecuaciones:

En la **ecuación (5.1)**, la compuerta de olvido decide qué información del estado anterior se debe mantener o descartar.
En la **ecuación (5.2)**, la compuerta de entrada decide qué nueva información se debe almacenar en el estado actual.
En la **ecuación (5.3)**, se calcula un nuevo estado candidato $\hat{C}_t$ que puede agregarse al estado

anterior.
En la **ecuación (5.4)**, aplica la operación de olvido al estado interno anterior de la célula y añadir nuevos valores candidatos, escalados en función de lo que hayamos decidido actualizar.
En la **ecuación (5.5)**, la compuerta de salida decide qué parte del estado oculto se debe usar como salida.
En la **ecuación (5.6)**, finalmente el nuevo estado oculto h(t) se calcula utilizando la compuerta de olvido, la puerta de entrada, el estado candidato y la puerta de salida

5.2.2. Redes Neuronales tipo Transformers

Las redes neuronales tipo Transformer representan una innovadora arquitectura que ha revolucionado el campo del procesamiento del lenguaje natural y otras aplicaciones de aprendizaje profundo. La principal característica distintiva de los Transformers es la atención auto-atentiva, que permite al modelo asignar diferentes niveles de importancia a diferentes partes de la entrada sin depender de la estructura secuencial.[Vaswani, 2017]
La **arquitectura general** de este tipo de redes es la que se visualiza a continuación:

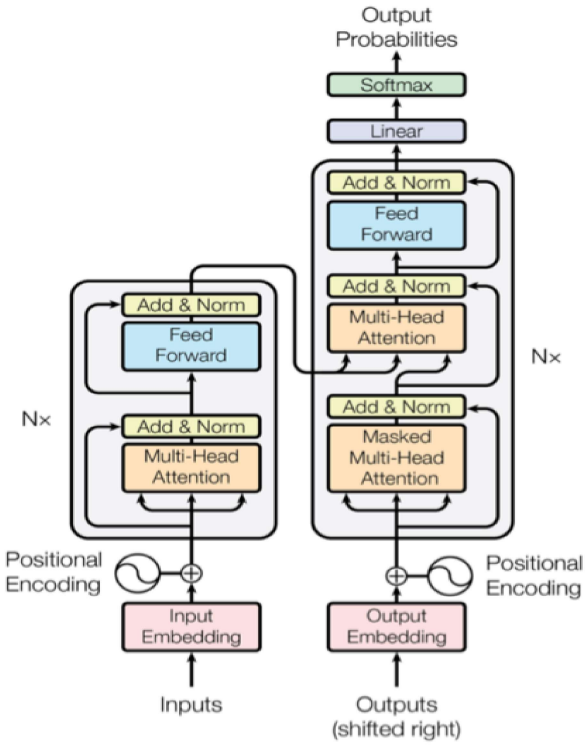

Figura 5.4: Arquitectura de una red neuronal tipo Transformer

Como se puede observar en la figura 5.4 La arquitectura Transformer consta de los siguientes componentes:

1. **Entrada y Embedding:** La entrada se compone de una secuencia de tokens, y cada token se representa mediante un vector de embeddings.

2. **Codificación Posicional:** Se agrega información posicional a los embeddings para capturar la estructura secuencial de la entrada.

3. **Bloque de Atención:** Este bloque es fundamental y se compone de múltiples cabezas de atención. Cada cabeza calcula una atención ponderada sobre todas las posiciones de entrada.

4. **Multi-Head Attention:** La atención se realiza de manera paralela en varias cabezas, y las salidas se concatenan y proyectan linealmente.

5. **Capa de Normalización y Feedforward:** Después de la atención, se aplica una capa de normalización y una red feedforward completamente conectada.

6. **Conexiones Residuales y Normalización Layer:** Se utilizan conexiones residuales y normalización por capas alrededor de cada subcapa del bloque.

7. **Conjunto de Capas:** El modelo completo se compone de conjuntos de capas para codificación (encoder) y decodificación (decoder).

8. **Capa de Salida:** La capa de salida produce las probabilidades de salida para cada token.

9. **Enmascaramiento de Atención:** En el decodificador, se aplica enmascaramiento de atención para garantizar que cada posición futura no tenga acceso a información posterior en la secuencia de entrada.

La capacidad de atención auto-atentiva y procesamiento en paralelo hace que las redes Transformers sean altamente eficientes y escalables para tareas complejas de procesamiento del lenguaje natural, traducción automática y otras aplicaciones.

5.3. Procesamiento de Lenguaje Natural (PLN)

El procesamiento del lenguaje natural (PLN) es una rama de la inteligencia artificial que permite a las computadoras comprender el lenguaje humano, tanto en forma de texto como de voz. Combina técnicas de lingüística computacional, modelos estadísticos, aprendizaje automático y aprendizaje profundo para interpretar el significado completo de los textos y la intención y el sentimiento del hablante o escritor. Algunas de las tareas de esta rama son: Reconocimiento de entidades nombradas, análisis de opinión y análisis de sentimiento.

5.3.1. Conceptos Básicos referentes a PLN

En el ámbito del Procesamiento del Lenguaje Natural (PLN) hay terminos básicos pero a la vez fundamentales que se utilizan para describir, abordar y realizar tareas del PLN de manera efectiva. Estos términos son:

5.3.1.1. Task

En PLN, “task” se refiere a la tarea específica que se lleva a cabo utilizando técnicas de procesamiento de lenguaje natural. Estas tareas pueden variar ampliamente e incluir actividades como clasificación de texto, traducción automática, resumen de texto, análisis de sentimiento, etiquetado de partes del discurso, entre otras. Cada tarea de PLN tiene sus propios objetivos y métodos, y a menudo implica el uso de algoritmos y modelos de aprendizaje automático para procesar y comprender el lenguaje humano.

5.3.1.2. Corpus

Un "corpus" es una colección estructurada de textos o documentos en un formato electrónico que se utiliza para investigaciones y análisis en el campo del PLN. Puede contener una variedad de tipos de textos, como artículos, libros, conversaciones, sitios web, entre otros. Los corpus se utilizan para entrenar modelos de PLN, realizar investigaciones lingüísticas y realizar tareas de procesamiento de texto. Un corpus puede ser específico de un dominio, idioma o propósito particular.

5.3.1.3. Vocabulario

El "vocabulario" en PLN se refiere al conjunto de palabras o términos únicos que se encuentran en un corpus o en un conjunto de datos de texto. El vocabulario puede variar en tamaño y complejidad según el contexto. En tareas de PLN, el análisis del vocabulario puede incluir la identificación y representación de palabras individuales, la frecuencia de aparición de palabras, la eliminación de palabras comunes (stopwords), y la construcción de vectores de palabras para su procesamiento y análisis.

5.3.2. Preprocesamiento en tareas de PLN

El preprocesamiento de datos textuales es una parte fundamental en el Procesamiento de Lenguaje Natural (NLP) ya que ayuda a limpiar y estructura el texto antes de aplicar técnicas de análisis o modelado. A continuación las tareas más comunes de preprocesamiento de datos en NLP:

- **Tokenización:** La tokenización es el proceso de dividir un texto en unidades más pequeñas, por lo general, estas unidades tienen relación palabras o signos de puntuación.
- **Eliminación de stopwords:** Los stopwords son palabras comunes que generalmente carecen de significado en el análisis de texto, como lo son artículos y preposiciones. Estas palabras se eliminan para reducir el ruido y el tamaño del vocabulario.
- **Stemming:** El stemming busca reducir una palabra a su rañiz mediante la eliminación de sufijos y prefijos. Como consecuencia se puede tener una palabra que no existe en el idioma, pero que es una representación truncada de la palabra original.
- **Lematización:** La lematización también busca reducir las palabras a su forma base, pero lo hace de manera mas inteligente al considerar el contexto y gramática del idioma, lo que obtiene una palabra real, la cual es conocida como "lema", la cual puede encontrarse en un diccionario.

5.3.3. Word Embeddings

Los Word Embeddings, o incrustaciones de palabras, son representaciones vectoriales de palabras en espacios semánticos continuos. Este enfoque se ha convertido en un componente fundamental en el procesamiento del lenguaje natural (PLN) debido a su capacidad para capturar relaciones semánticas y sintácticas entre palabras. En lugar de representar palabras como entradas discretas, los Word Embeddings asignan vectores de números reales a cada palabra, lo que permite modelar la similitud y contexto semántico de manera más efectiva.[Briceño, 2021]

El concepto subyacente de Word Embeddings radica en la idea de que palabras con significados similares deberían tener representaciones vectoriales cercanas en el espacio. Esto contrasta con enfoques tradicionales, como Bag of Words, que no capturan la semántica intrínseca de las palabras. Las representaciones vectoriales de Word Embeddings poseen varias propiedades útiles:

- **Semántica Contextual:** Las palabras similares en contexto tienen representaciones similares.
- **Operaciones Vectoriales:** Las operaciones vectoriales en el espacio de Word Embeddings pueden capturar analogías y relaciones semánticas (por ejemplo, "rey" - "hombre" + "mujer" ≈ "reina").
- **Generalización:** Los Word Embeddings entrenados en grandes conjuntos de datos pueden generalizar bien a tareas específicas, incluso con vocabularios limitados.

Los Word Embeddings se utilizan en una variedad de aplicaciones en PLN, incluyendo:

- **Modelado de Lenguaje:** Mejora la precisión en tareas como predicción de palabras siguientes.
- **Clasificación de Documentos:** Facilita la captura de similitudes semánticas entre documentos.
- **Traducción Automática:** Ayuda a manejar la variabilidad léxica entre idiomas.
- **Análisis de Sentimientos:** Permite capturar matices semánticos en la interpretación de expresiones sentimentales.

Los Word Embeddings han revolucionado el campo del procesamiento del lenguaje natural al proporcionar representaciones semánticas densas y contextualizadas para palabras. Su capacidad para capturar significados y relaciones semánticas ha impulsado avances significativos en diversas aplicaciones, convirtiéndolos en una herramienta fundamental en la caja de herramientas de los investigadores y practicantes de PLN.

5.3.4. Bag of Words

El modelo Bag of Words (BoW) es una técnica simple pero efectiva utilizada en procesamiento del lenguaje natural para representar documentos y textos. El modelo BoW se centra en la frecuencia de las palabras sin tener en cuenta su orden o estructura gramatical.
En el modelo Bag of Words, un documento se representa como un conjunto no ordenado de palabras, ignorando completamente la estructura gramatical y la secuencia de las palabras. Cada palabra contribuye a un vector de características, y la frecuencia de cada palabra en el documento determina su valor en el vector.[Nisha, 2020]
Asimismo, el proceso de creación del modelo Bag of Words implica los siguientes pasos:

1. **Tokenización:** El documento se divide en palabras individuales, conocidas como tokens.
2. **Creación del Vocabulario:** Se construye un vocabulario único a partir de todas las palabras únicas en el conjunto de documentos.

3. **Representación del Documento:** Cada documento se representa como un vector, donde cada posición corresponde a una palabra del vocabulario y su valor es la frecuencia de esa palabra en el documento.

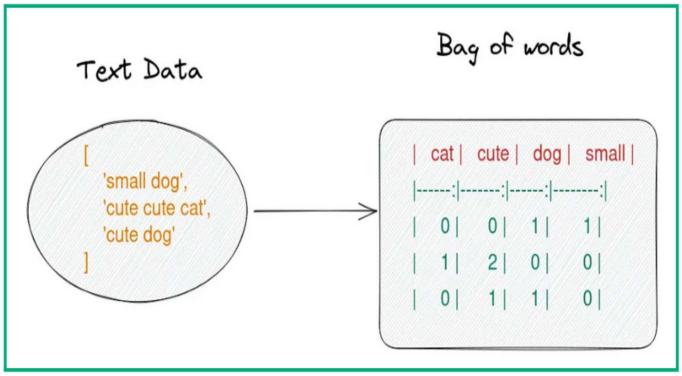

Figura 5.5: Ejemplo de aplicación a un corpus muy acotado del modelo Bag of Words

5.3.5. Matriz TD-IDF

La matriz TF-IDF (Term Frequency-Inverse Document Frequency) es una técnica utilizada en procesamiento del lenguaje natural para representar documentos mediante la ponderación de la importancia de las palabras en un corpus. A diferencia del modelo Bag of Words, la matriz TF-IDF tiene en cuenta tanto la frecuencia de las palabras como su relevancia en el contexto global.

El concepto central de la matriz TF-IDF es calcular un peso para cada palabra en un documento, considerando su frecuencia en ese documento y su rareza en el corpus global. La fórmula general para el cálculo del peso TF-IDF de un término t en un documento d es:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t) \tag{5.7}$$

donde:

- $\text{TF}(t, d)$ es la frecuencia del término t en el documento d.
- $\text{IDF}(t)$ es el logaritmo del inverso de la frecuencia de documentos que contienen el término t en el corpus.

El proceso para construir la matriz TF-IDF implica los siguientes pasos:

1. **Tokenización:** Similar al modelo Bag of Words, el documento se divide en palabras individuales.
2. **Cálculo de TF:** Se calcula la frecuencia de cada palabra en el documento.
3. **Cálculo de IDF:** Se calcula el logaritmo del inverso de la frecuencia de documentos que contienen cada palabra en el corpus.
4. **Multiplicación:** Se multiplica la matriz TF por la matriz IDF para obtener la matriz final TF-IDF.

5.4. Grandes Modelos de Lenguaje(LLM)

Un Gran Modelo de Lenguaje (LLM) es un modelo de lenguaje grande en el ámbito del procesamiento del lenguaje natural (NLP). Se refiere a un modelo de inteligencia artificial que ha sido entrenado en grandes cantidades de datos de texto con el objetivo de comprender y generar lenguaje humano de manera coherente y precisa.[Guo, 2023]
Estos modelos de lenguaje utilizan técnicas de aprendizaje automático, como el aprendizaje profundo, para capturar patrones y estructuras en el lenguaje. A través de su entrenamiento, los LLM's aprenden la distribución estadística de las palabras y las relaciones entre ellas en un corpus de texto extenso. Esto les permite generar texto que es coherente y tiene sentido gramatical.
Un ejemplo conocido de un LLM es GPT-3 (Generative Pre-trained Transformer 3), desarrollado por OpenAI. GPT-3 ha sido entrenado en una amplia variedad de textos, como libros, artículos de noticias y páginas web, lo que le permite responder preguntas, completar oraciones, generar texto y realizar otras tareas relacionadas con el lenguaje.

5.4.1. OpenAI API

La API de OpenAI es una interfaz de programación de aplicaciones (API) proporcionada por OpenAI, una empresa líder en inteligencia artificial. La API de OpenAI permite acceder y utilizar los modelos de lenguaje de OpenAI, como GPT-3, en sus propias aplicaciones y servicios.
Mediante la API de OpenAI, se pueden enviar solicitudes a los servidores de OpenAI para realizar tareas de procesamiento del lenguaje natural (NLP). Pueden enviar texto de entrada y recibir respuestas generadas por el modelo de lenguaje correspondiente.[Brockman, 2020]

5.4.1.1. Formación Adicional: Curso "ChatGPT Prompt Engineering for Developers"

Como parte integral del desarrollo de este proyecto de titulación, el alumno reconoció la importancia de adquirir habilidades específicas en la ingeniería de prompts. Para fortalecer su competencia, completó con éxito el curso gratuito ÇhatGPT Prompt Engineering for Developers", ofrecido por DeepLearning.AI [Deeplearning.Ai, 2023]. El objetivo principal de este curso fue familiarizarse con buenas prácticas en la ingeniería de prompts, explorar nuevas aplicaciones para modelos de lenguaje de última generación y perfeccionar las habilidades de redacción e iteración de prompts mediante el uso de la API de OpenAI.

Este curso proporcionó una valiosa comprensión de estrategias efectivas para guiar y obtener resultados óptimos de LLM's, mejorando así la capacidad del alumno para aprovechar plenamente las capacidades del modelo en el contexto de su proyecto de titulación.

5.5. Métricas

5.5.1. Coherencia

La métrica de coherencia C_v es un indicador fundamental en el campo del Procesamiento del Lenguaje Natural (PLN) que se utiliza para evaluar la calidad y la interpretabilidad de los modelos generativos de tópicos, como el modelo Latent Dirichlet Allocation (LDA). Esta métrica tiene como objetivo cuantificar la cohesión semántica y la interpretabilidad de los tópicos extraídos por un modelo de tópicos, permitiendo así la selección del número óptimo de tópicos en un corpus de texto.

Cuando se aplica el modelado de tópicos a un conjunto de documentos, es esencial determinar el número adecuado de tópicos que mejor representa la estructura subyacente de los datos. La métrica de coherencia C_v aborda esta cuestión al calcular una puntuación que refleja la cohesión de las palabras dentro de un tópico y la separación entre diferentes tópicos. Una puntuación de coherencia más alta indica que las palabras en un tópico están más relacionadas y que los tópicos son más distintos entre sí, lo que mejora la interpretación y utilidad de los tópicos generados.

La métrica C_v opera mediante la comparación de las palabras clave extraídas de los tópicos con respecto a su co-ocurrencia en el corpus de documentos. Al medir la relación entre las palabras dentro de un tópico, se calcula una puntuación que se considera un indicador de la calidad de los tópicos generados. Dicho de otra manera, la métrica C_v ayuda a determinar cuánto sentido tienen los tópicos y si representan de manera coherente aspectos específicos del contenido de los documentos.
Se calcula con la siguiente fórmula:

$$C_v = \frac{2}{N(N-1)} \sum_{i<j} \frac{D(w_i, w_j) + 1}{\log\left(\frac{R(w_i,w_j)}{S(w_i,w_j)}\right) + 1} \tag{5.8}$$

En donde:

- N es el número total de palabras clave en todos los tópicos.
- w_i y w_j son dos palabras clave distintas.
- D(w_i,w_j) es la distancia co-documento entre w_i y w_j (número de documentos en los que aparecen juntas).
- R(w_i,w_j) es el número de documentos que contienen w_i.
- S(w_i,w_j) es el número de documentos que contienen w_j.

5.5.2. Accuracy

La *accuracy* (ACC) mide la proporción de instancias correctamente clasificadas respecto al total de instancias y se define como:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.9}$$

5.5.3. Recall

El *recall* (sensibilidad) mide la proporción de instancias positivas que fueron correctamente identificadas y se calcula como:

$$Recall = \frac{TP}{TP + FN} \tag{5.10}$$

5.5.4. Precision

La *precision* mide la proporción de instancias positivas identificadas correctamente respecto a todas las instancias identificadas como positivas:

$$Precision = \frac{TP}{TP + FP} \tag{5.11}$$

5.5.5. F1-Score

El *F1-score* se define como la media harmónica de precision y recall, y se calcula mediante la fórmula:

$$F1 - Score = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.12}$$

Donde:

TP es el número de verdaderos positivos,
TN es el número de verdaderos negativos,
FP es el número de falsos positivos,
FN es el número de falsos negativos.

5.5.6. Macro Avg

La métrica **Macro Avg** es una herramienta valiosa en la evaluación de modelos de clasificación multiclase, especialmente en situaciones donde existe un desequilibrio en la distribución de las clases con el objetivo de no sesgarse hacia las clases más frecuentes en un conjunto de datos. Su fórmula se define de la siguiente manera:

$$MacroAvg_{Métrica} = \frac{Metrica_{Clase1} + Metrica_{Clase2} + ...Metrica_{ClaseN}}{N} \tag{5.13}$$

En el contexto de este trabajo de título la variable metrica ϵ {Recall, Precision y F1-Score}

Capítulo 6

Metodología

Para llevar a cabo esta tesis, se adoptó la metodología **CRISP-DM LITE**, una variante simplificada de la reconocida CRISP-DM.

Figura 6.1: Diagrama representativo de la metodología CRISP DM LITE

Esta versión omite la fase de implementación, ya que está más allá del alcance de este trabajo. La CRISP-DM LITE es ampliamente utilizada en la minería de datos, un proceso destinado a descubrir patrones y conocimientos útiles a partir de extensos conjuntos de datos. Esta metodología se estructura en cinco fases, que se describen de manera concisa a continuación:

- Comprensión del negocio: Se emprende la tarea de comprender los objetivos y necesidades de Adereso como negocio, estableciendo así la base argumental para el desarrollo del actual proyecto.
- Comprensión de los datos: Se realiza una exploración minuciosa de los datos disponibles provenientes de la mesa de ayuda Adereso Desk, analizando su estructura y calidad, todo esto con el objetivo de detectar posibles problemas que puedan surgir para la correcta realización del Prototipo de Analítica de Texto.
- Preparación de los datos: En este paso, se llevan a cabo labores de limpieza, transformación y manipulación de los datos textuales, asegurando que se encuentren en condiciones óptimas y alineados con los requisitos y desafíos específicos del análisis textual propuesto para el prototipo.

- Modelado: En este punto, se eligen y aplican técnicas de modelado pertinentes, adaptadas específicamente a las características de los usuarios del sector retail de Adereso. La selección entre modelos descriptivos o predictivos se detallará en función de cada dimensión.

- Evaluación: Se lleva a cabo un análisis detallado del rendimiento de los modelos previamente creados en cada dimensión del prototipo, evaluando su conformidad con los requisitos y objetivos establecidos en el proyecto.

Capítulo 7

Desarrollo Metodológico

7.1. Comprensión del negocio

Adereso, fundada en 2014, es una startup chilena que se posiciona como un proveedor de Software as a Service (SaaS). Su enfoque se encuentra en el sector de las Tecnologías de la Información, más específicamente en la subindustria de "Application Software". A lo largo de los años, ha expandido significativamente su presencia en varios países de Latinoamérica, ofreciendo servicios y productos basados en un modelo B2B2C.

Dentro de su oferta, Adereso se destaca por proporcionar soluciones que permiten la automatización de procesos de Servicio de Atención al Cliente (SAC) a través de diversos canales digitales, como Facebook, Whatsapp, Twitter, entre otros. La empresa ha desarrollado dos líneas de productos clave para lograr este objetivo: **Adereso Desk** y **Adereso AI**.

Adereso Desk se presenta como una mesa de ayuda o helpdesk en la nube que omnicanaliza la atención al cliente, focalizándose especialmente en los canales digitales de postventa de sus empresas usuarias. Aquí, los usuarios del software, en su mayoría agentes especializados en atención al cliente contratados por las empresas usuarias de Adereso, utilizan la plataforma para gestionar tickets. El modelo de negocios asociado a este producto se basa en la venta de licencias del software.

Por otro lado, **Adereso AI** se embarca en la implementación de Inteligencia Artificial y el uso de Grandes Modelos de Lenguaje para construir chatbots personalizados para las empresas contratantes. El propósito de estos chatbots es automatizar la atención al cliente en medios digitales, ofreciendo un asistente virtual capaz de resolver tickets de forma autónoma, eliminando la necesidad de intervención humana.
Dado que este trabajo de título se centra exclusivamente en datos relacionados con el helpdesk, a partir de este punto, cualquier referencia a los clientes, rendimiento de la organización, la posición de Adereso en su mercado y los desafíos experimentados en el último año se abordarán exclusivamente desde la perspectiva de Adereso Desk.

En cuanto al desempeño organizacional de Adereso, se debe mencionar que esta startup el año 2023 registro las siguientes cifras respecto de su número de clientes e ingresos mensuales:

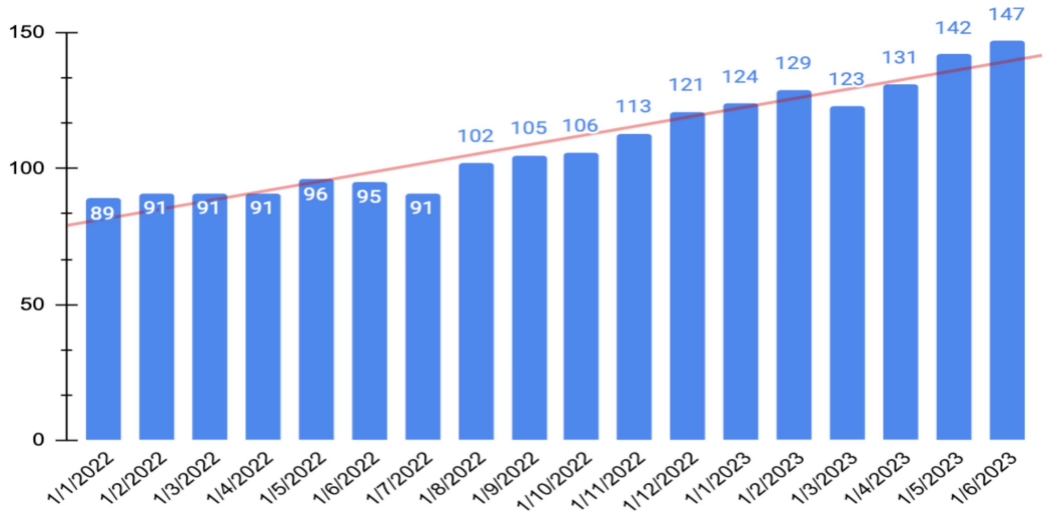

Figura 7.1: Número de clientes desde enero 2022 hasta junio 2023

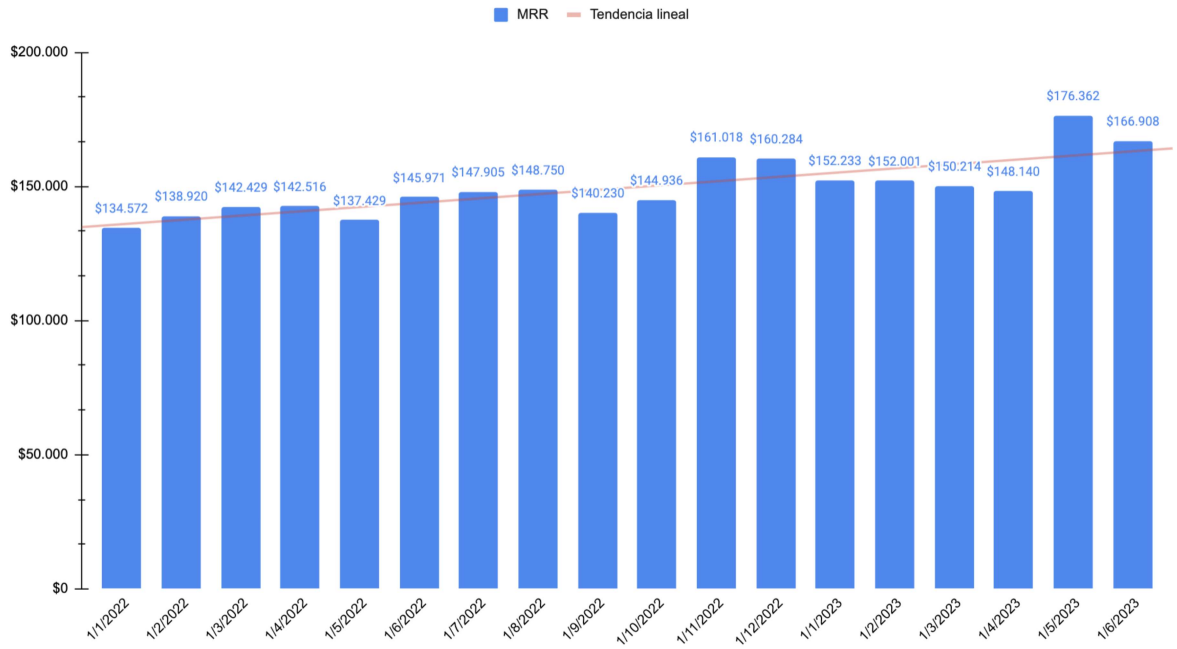

Figura 7.2: MRR's desde enero 2022 hasta junio 2023

Como se puede ver en la figura 7.1 y 7.2, los clientes y los MRR's se han mantenido en un ritmo ascendente en el período Enero 2022 - Junio 2023.
En la misma línea, los tickets de los clientes de Adereso se segmentan según industria de la siguiente forma:

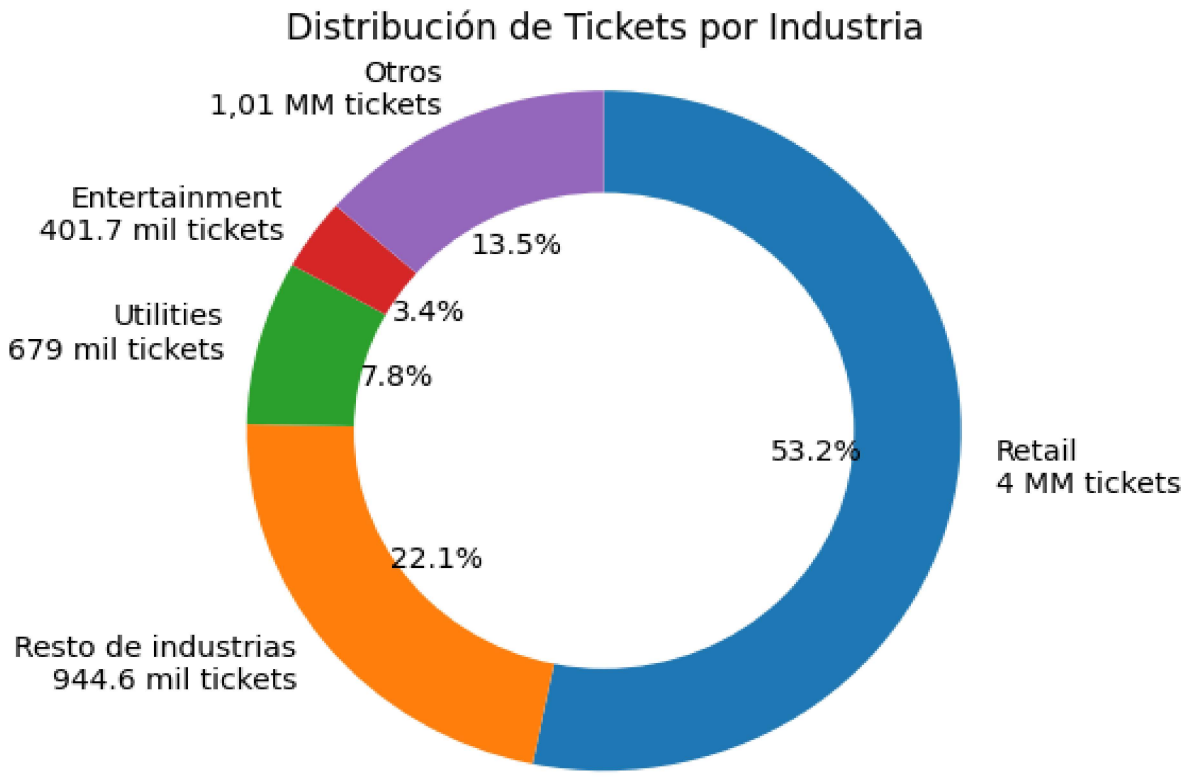

Figura 7.3: Número de tickets por industria desde enero 2022 hasta junio 2023

Como revela la figura 7.3, los clientes de retail son aquellos que mantienen un mayor tráfico a nivel de tickets en el helpdesk.
En el actual mercado de empresas SaaS, abundan numerosas opciones que brindan servicios a clientes a nivel mundial, incluyendo aquellas con presencia en Latinoamérica. Dentro de este panorama, a pesar del notable aumento en sus MRR y en el número de clientes durante el período de enero de 2022 a junio de 2023, Adereso se encuentra en una posición que aún no le permite competir directamente con las grandes empresas líderes de la industria, tales como Zendesk, Salesforce o Onemarketer. Estas compañías han consolidado su presencia y han alcanzado un reconocimiento significativo en el mercado, manteniendo a Adereso en la posición de una empresa seguidora. Este escenario se ve agravado por algunos indicadores específicos de desempeño. En particular, el churn rate experimentó un aumento del 11 % el último año, superando significativamente el objetivo establecido de mantenerlo en el rango del 3 al 5 %, como se había logrado en años anteriores. Además, la actividad de las licencias de los clientes ha experimentado una disminución constante, indicando que, a pesar de pagar por los servicios de la empresa, el interés de los clientes en la herramienta no es óptimo.

Un aspecto crítico revelado por los testimonios de salida de los clientes es que más del 60 % abandonaron Adereso el último año debido a la migración o integración con herramientas y software de competidores directos como Zendesk o Salesforce.

Ante los retos mencionados, el Equipo de Desarrollo y Producto de Adereso consideró imperativo explorar nuevas oportunidades para diferenciar su producto, Adereso Desk. La convicción

dentro del equipo reside en la creencia de que el desarrollo de capacidades analíticas, específicamente enfocadas en el contenido del texto, podría ser un diferenciador clave frente a la competencia, dado que ninguna de esas empresas a desarrollado un producto similar.

La visión estratégica es la de ofrecer una analítica avanzada que proporcione insights valiosos, elevando así el nivel de control y gestión para las empresas usuarias de Adereso. Este enfoque surge en respuesta a la notable influencia que el término "Inteligencia Artificial" ha tenido en el año 2023, reconociendo la importancia de esta tendencia emergente en el ámbito tecnológico. La idea central es que esta diferenciación no solo abordaría los desafíos actuales sino que también posicionaría a Adereso como líder en la implementación de tecnologías innovadoras.

7.2. Comprensión de los datos

En primer lugar, los datos con los cual se trabajará a lo largo de este trabajo de título, guardan relación con exportaciones realizadas desde el Helpdesk para cada uno de los clientes de retail de Adereso. La unificación de dichas exportaciones en un solo dataset es a lo que llamaremos corpus de entrada, cuya temporalidad corresponde al 1er semestre del año 2023, con un tamaño de poco mas de 6.2 millones de registros en donde mas del 90 % de ellos corresponden a grandes retailers del país como lo son: Walmart, Ripley o París. Este dataset tiene 24 columnas pero para efectos explicativos tomaremos la siguiente muestra:

Tabla 7.1: Extracto de los datos utilizados

Ticket	Texto	Fecha	Hora	Canal	Nick del Cliente
8707500	Esta con cinta de embalaje y le faltan cervezas	01-01-2023	14:53:52	Instagram	Sergio Rojas

Respecto de que significa cada una de las columnas de la extracción antes visualizada, se puede decir lo siguiente:

- Ticket: refleja un número único asignado a cada conversación iniciada por el cliente por cualquiera de las redes sociales que el usuario de Adereso haya vinculado al helpdesk.
- Texto: hace referencia al contenido del mensaje enviado por el cliente.
- Fecha: hace referencia a la fecha en la que el cliente envió el mensaje.
- Hora: hace referencia a la hora en la que el cliente envió el mensaje.
- Hora: hace referencia al canal de procedencia por el cual el cliente envió el mensaje.
- Nick del cliente: hace referencia al nombre o número telefónico desde el cual el cliente envió el mensaje.

Como ya se dijo antes esta base de datos tiene poco mas de 6.2 millones de registros y 24 columnas que contienen información de cada uno de los mensajes enviados por clientes de las empresas de retail de Adereso. Como último detalle para agregar, cabe comunicar que cada uno de estos registros no corresponde a una conversación como tal, sino que a un mensaje del cliente enviado a algunas de la cuentas vinculadas al helpdesk. Por lo que, por ejemplo si un cliente envió 10 mensajes durante una interacción con el servicio al cliente online de un retailer cliente Adereso, en la base de datos habrían 10 registros asociados al mismo valor único del Ticket que se crea con el envío del primer mensaje. El manejo de esta naturaleza de los datos se mostrará más adelante en el informe.

7.3. Preparación de los datos

El ítem de preparación de los datos se divide en tres subítems: Consolidación de los datos, Limpieza de la información textual y Preprocesamiento habitual en tareas de Procesamiento

del Lenguaje Natural (PLN). Cada uno de estos subítems se desarrolla y explica a continuación:

7.3.1. Consolidación de los datos

Eliminación de mensajes carentes de contenido semántico:

En este paso de la preparación de los datos, se eliminaron aquellos registros que no tenían un contenido semántico coherente como para formar parte de esta analítica. Estos mensajes eliminados correspondían a registros del tipo:

- "Mensaje de Bot", mensajes que tenían que ver con respuestas ejecutadas por bots de implementados en el servicio al cliente de las empresa de retail usuarias de Adereso.
- "MEDIA_MESSAGE", mensajes que tenían relación, con aquellas situaciones clientes enviaban fotos o audios para comunicarse con la empresa de retail de manera online.

Con la eliminación de este tipo de registros, se pasó de 13,2 millones a 10,3 millones de registros. (disminución de un 21,9 % de registros)

Eliminación de mensajes enviados por los agentes:

En este paso de la preparación de los datos, se eliminaron aquellos mensajes que correspondían a respuestas ejecutadas por lo agentes pertenecientes a estas de retail usuarias de Adereso. Se eliminaron, debido a que este tipo de mensajes no entraban en el análisis pertinente del actual proyecto.
Con la eliminación de este tipo de registros, se pasó de 10,3 millones a 6,2 millones de registros.(disminución de 39,8 % de registros)

Agrupación de registros por conversación:

En este paso de la preparación de los datos, se agruparon todos los mensajes por el número único correspondiente a la columna de "Ticket", y se juntaron todos en un mismo registro. Con respecto al cambio que tuvieron algunas de las columnas del dataset de trabajo, con esta agrupación de registros por conversación, los valores que tomaron las columnas de "Fecha" y "Hora" fueron los del último mensaje enviado por el cliente.
Con la eliminación de este tipo de registros, se pasó de 6,2 millones a 970 mil de registros.(disminución de 84,4 % de registros)

Con la finalización de los hitos, que implicaron la eliminación de mensajes sin contenido semántico, la exclusión de mensajes enviados por agentes y la agrupación de registros por conversaciones, se logró una reducción significativa en el tamaño del conjunto de datos. Este pasó de alrededor de 13.2 millones de registros a aproximadamente 970 mil conversaciones, lo que representa una disminución del 92.7 % en el número total de registros. Sin embargo, es importante destacar que, en términos de contenido efectivo, la reducción es del 53 % (de 13.2 millones a 6.2 millones de registros). Esto se debe a que, en el proceso de agrupación de registros por conversación, se llevó a cabo una reorganización de los datos en lugar de una eliminación directa.

7.3.2. Limpieza de la data textual

Durante esta etapa, se llevan a cabo diversas tareas destinadas a garantizar la calidad y uniformidad de los datos textuales recopilados de las conversaciones antes mencionadas. A continuación, se detallan los principales procedimientos de limpieza de datos que se aplicaron:

Normalización del texto (minúscula):
Para lograr uniformidad y simplificar la comparación de palabras y frases en el análisis, todo el texto de las conversaciones se convirtió a minúsculas. Este paso se implementó de manera consistente en todas las conversaciones registradas.

Eliminación de URL's:
Las URL presentes en las conversaciones se consideraron irrelevantes para nuestros fines analíticos. Por lo tanto, se procedió a la eliminación de todas las referencias a direcciones web, lo que contribuyó a reducir el ruido en los datos.

Eliminación de HTML's:
Con el objetivo de enfocarnos en el contenido textual sustantivo, se detectaron y eliminaron fragmentos de código HTML que a veces se encontraban en las conversaciones. Esto ayudó a garantizar que los datos fueran puramente textuales y aptos para el análisis.

Eliminación de caracteres especiales:
Se identificaron y eliminaron caracteres no alfabéticos o numéricos que no contribuyeron al análisis del contenido textual. Esto incluyó símbolos matemáticos y otros caracteres especiales tales como emojis, que no tenían relevancia para nuestros objetivos.

Eliminación de signos de puntuación:
Para simplificar el texto y facilitar el procesamiento posterior, se llevaron a cabo la eliminación de signos de puntuación, como comas, puntos, exclamaciones y otros caracteres relacionados con la gramática.

Eliminación de saltos de línea y símbolos de desigualdades:
Con el fin de garantizar que el texto estuviera formateado de manera uniforme y libre de símbolos innecesarios, se eliminaron saltos de línea y símbolos de desigualdades que no tenían un propósito específico en nuestro análisis.

7.3.3. Preprocesamientos típicos en tareas del Procesamiento del Lenguaje Natural

La calidad y la adecuación de los datos son elementos cruciales en la fase de preparación de datos en proyectos de procesamiento del lenguaje natural (PLN). Para garantizar que nuestros datos sean apropiados para su análisis y modelado, se llevaron a cabo varios procesos de preprocesamiento. Los siguientes apartados describen las técnicas clave aplicadas:

- **Tokenización:** La tokenización es un paso esencial en el procesamiento de texto que implica dividir el texto en unidades más pequeñas, conocidas como tokens. En este proyecto, se utilizó el tokenizador proporcionado por NLTK (Natural Language Toolkit) para segmentar el texto en palabras y frases, lo que facilitó la representación y el análisis de las conversaciones de los clientes.

- **Eliminación de Stopwords:** Las "stopwords" son palabras comunes y frecuentes en un idioma que generalmente no aportan información relevante en tareas de PLN. Para reducir el ruido en nuestros datos, se eliminaron stopwords utilizando la lista proporcionada por NLTK, que se descargó previamente. Esto permitió enfocarnos en las palabras que realmente contenían información valiosa.

- **Lematización:** La lematización es el proceso de reducir palabras a su forma base o lema. En este proyecto, se empleó el modelo "es_core_news_sm" de spaCy para llevar a cabo la lematización. Esto ayudó a unificar diferentes formas de una palabra y simplificó el análisis al reducir la variabilidad léxica.

- **Aplicación de Stemmers (Snowball y Lancaster):** Los stemmers son herramientas que reducen las palabras a sus raíces, conocidas como "stems". Se aplicaron dos stemmers diferentes, Snowball y Lancaster, para explorar las diferencias en el procesamiento. Esto facilitó la identificación de la variabilidad en la forma de las palabras y permitió una comparación más exhaustiva.

A continuación se mostrará un ejemplo de que hacen específicamente cada uno de estos preprocesamientos:

Tabla 7.2: Extracto para visualizar los preprocesamientos propuestos

Ticket	Texto	Texto Limpio	Texto sin Stopwords	Texto Stemmizado con Snowball	Texto Stemmizado con Lancaster	Texto Lematizado
8707500	Esta con cinta de embalaje y le faltan cervezas	esta con cinta de embalaje y le faltan cervezas	cinta embalaje faltan cervezas	cint embalaj falt cervez	cint embalas falt cervezas	cinta embalaje faltar cervezas

Cabe recalcar que a partir de la columna "Texto" en adelante los preprocesamientos son acumulativos hasta la columna "Texto sin stopwords", es decir, "Texto Limpio" recibe como entrada el contenido de "Texto" para limpiarlo y análogamente para la columna "Texto sin

Stopwords". Desde esta columna en adelante se realizan preprocesamientos de manera independiente, es decir, los preprocesamientos propios de las columnas "Texto Stemmizado con Snowball", "Texto Stemmizado con Lancaster" y "Texto Lematizado" toman como entrada el contenido de la columna "Texto sin Stopwords", no son acumulativos.

Aclarado esto último, analicemos el contenido por columna de la tabla 7.2:
En la **columna "Texto"**, tiene relación con el contenido del mensaje original proveniente de las exportaciones del helpdesk.
En la **columna "Texto Limpio"**, solo se procesó todo el texto anterior y lo reescribió todo en minúscula. Esto debido a que el contenido de texto original no contenía tanto ruido tales como URl's, HTML's, etc.
En la **columna "Texto sin Stopwords"**, se eliminan varias palabras de la columna anterior, en su mayoría correspondientes a preposiciones.
En la **columna "Texto Stemmizado con Snowball"** y **columna "Texto Stemmizado con Lancaster"**, si bien "enraízan" bien las palabras, este preprocesamiento produce tokens o palabras inexistentes en la lengua española. Por el contrario en la **columna "Texto Lematizado"**, este preprocesamiento produce lemas totalmente coherentes y pertenecientes a la lengua española. Es por eso que de ahora en adelante se trabajará y modelará con esta columna, en donde se lematizan los textos originales o bien esta columna pero tokenizada, como se muestra a continuación:

Tabla 7.3: Extracto para visualizar los preprocesamientos propuestos

Ticket	Texto	Texto Lematizado	Texto Lematizado tokenizado
8707500	Esta con cinta de embalaje y le faltan cervezas	cinta embalaje faltar cervezas	['cinta', 'embalaje', 'faltar', 'cervezas']

A modo de conclusión de esta etapa, cabe mencionar que estos procesos de preprocesamiento son esenciales para garantizar que nuestros datos estén listos para análisis avanzados en tareas de PLN. La combinación de tokenización, eliminación de stopwords, lematización y stemmers contribuye significativamente a la calidad y uniformidad de nuestros datos, allanando el camino para futuras etapas de modelado y análisis de texto.

7.4. Modelado y Resultados para cada una de las dimensiones

La sección de Modelado y Resultados se adentra en un análisis minucioso de cada dimensión, proporcionando una explicación exhaustiva de los enfoques y modelos específicos utilizados en cada caso.

En la primera dimensión, enfocada en los Top 20 unigramas, bigramas y trigramas más frecuentes, se empleará un enfoque descriptivo en lugar de predictivo. A través de un procesamiento detallado, que se explicará más adelante, se identificarán y extraerán los 20 unigramas, bigramas y trigramas más frecuentes. Esta elección se fundamenta en el hecho de que estos elementos representan aproximadamente el 20 % de la muestra y proporcionan una visión detallada de los patrones lingüísticos predominantes.

Para la segunda dimensión, centrada en la tipificación general de los tickets, se implementará un modelo en dos partes. En la primera fase, se aplicará el modelo LDA para contabilizar, caracterizar y asignar los tópicos ocultos en el corpus. La segunda parte implicará la construcción de un clasificador automático para las etiquetas descubiertas, con experimentos que involucrarán una red neuronal recurrente, específicamente una variante LSTM, y un modelo transformer llamado DistilBERT.

En la tercera dimensión, se realizará un análisis del sentimiento de los tickets utilizando el modelo GPT-3.5 a través de la API de OpenAI. Mediante la ingeniería de prompts, se inferirá el sentimiento de cada ticket del corpus a analizar.

Es importante destacar que, para las dimensiones 1 y 2, el modelado y los resultados se basarán en un corpus denominado **Muestreo 1**. Este muestreo consiste en una muestra del 20 % del corpus original, aproximadamente 180,000 tickets, distribuidos homogéneamente en el tiempo (mensual) y favoreciendo aquellos con un mayor número de palabras. Este enfoque es crucial para el análisis de tópicos con LDA, ya que más contenido mejora los resultados. La razón del porqué no se utilizará el corpus completo, tiene relación con el elevado gasto computacional que demandaba el procesar alrededor de 970 mil tickets para la ejecución de un modelo LDA y posterior entrenamiento de un clasificador.

Por otro lado, **el Muestreo 2** es una extracción de 9,600 tickets del Muestreo 1, también de manera homogénea temporal, alrededor de 1,600 tickets por mes. Esta selección se realiza considerando la naturaleza de pago de la API de OpenAI y el hecho de que un corpus más extenso no mejora significativamente el rendimiento de un modelo preentrenado.

7.4.1. 1era Dimensión: Top 20 unigramas, bigramas y trigramas más frecuentes

Una parte fundamental del actual proyecto involucra la exploración y comprensión de las conversaciones registradas de los clientes de la empresa de retail en busca de patrones lingüísticos claves. En esta primera dimensión, nos concentramos en el análisis de n-gramas, que comprenden un conjunto de n palabras contiguas en el texto. Los n-gramas, en particular los unigramas (1 palabra), bigramas (2 palabras) y trigramas (3 palabras), se convierten en

una herramienta esencial para desentrañar la estructura y los temas en las conversaciones. El corpus que se ocupará en esta tarea, como se declaró anteriormente, es el muestreo 1, en particular el contenido relacionado con la columna “Texto Lematizado”. A continuación, presentamos cómo se ha llevado a cabo esta tarea:
El proceso de obtener *los 20 unigramas más frecuentes* en el Muestreo 1, se inicia importando las bibliotecas esenciales, como NLTK para el procesamiento del lenguaje natural y Counter para el conteo de frecuencias. A continuación, se descarga el tokenizador denominado "punkt", el cual descompone el corpus previamente concatenado a partir de la columna "Texto Lematizado"del dataframe correspondiente al Muestreo 1. Se establece una función de filtro diseñada para excluir tokens con menos de 3 letras, elementos numéricos y términos específicos como “hola”, “ser”, “menu”, “tarde”, “buen” y “dia”. Estos términos se consideran innecesarios para el análisis debido a su frecuencia elevada en los tickets. Esta función de filtro contribuye a la formación del vocabulario del Muestreo 1. Posteriormente, la librería Counter se utiliza para calcular la frecuencia de cada unigrama en este vocabulario. Finalmente, se seleccionan los 20 unigramas más frecuentes mediante el uso del método ’most_common’ de Counter. Obteniendo el siguiente gráfico de barras:

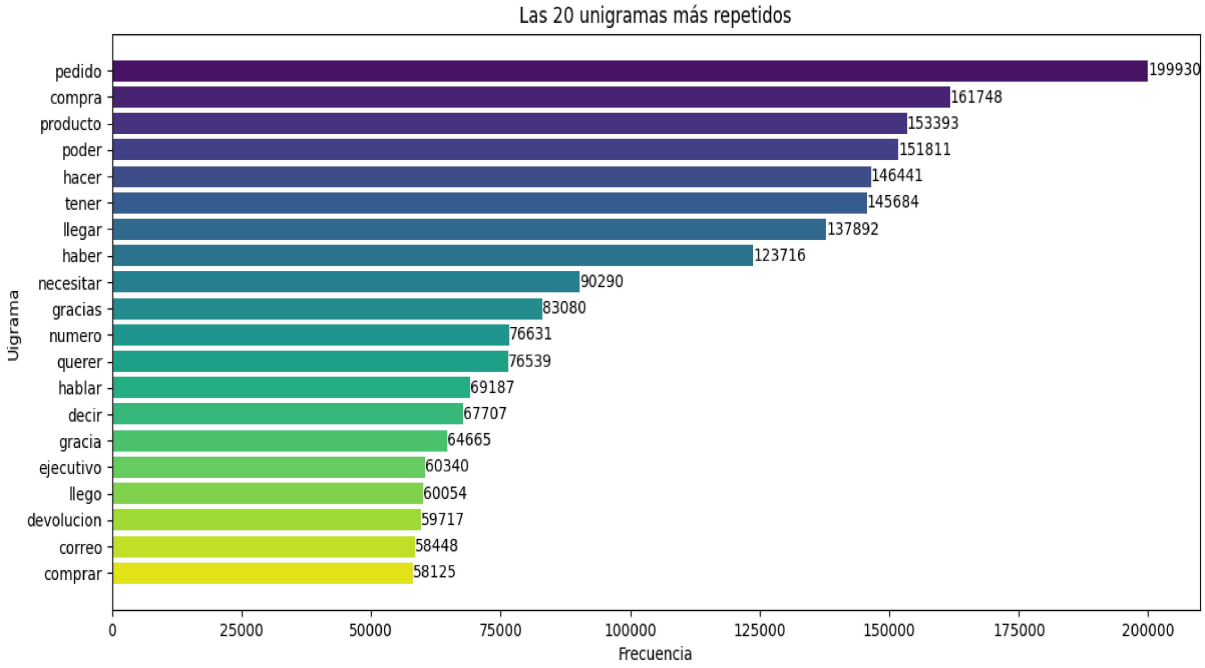

Figura 7.4: Gráfico de barras horizontal de los 20 unigramas mas frecuentes del muestreo 1

En la obtención de *los 20 bigramas y trigramas más frecuentes*, el procedimiento es análogo, con la excepción de que en la función de filtro se incorpora la condición de que, tanto en el par de tokens (bigrama) como en el trío de tokens (trigrama), los tokens deben ser distintos entre sí. Obteniendo los siguientes gráficos:

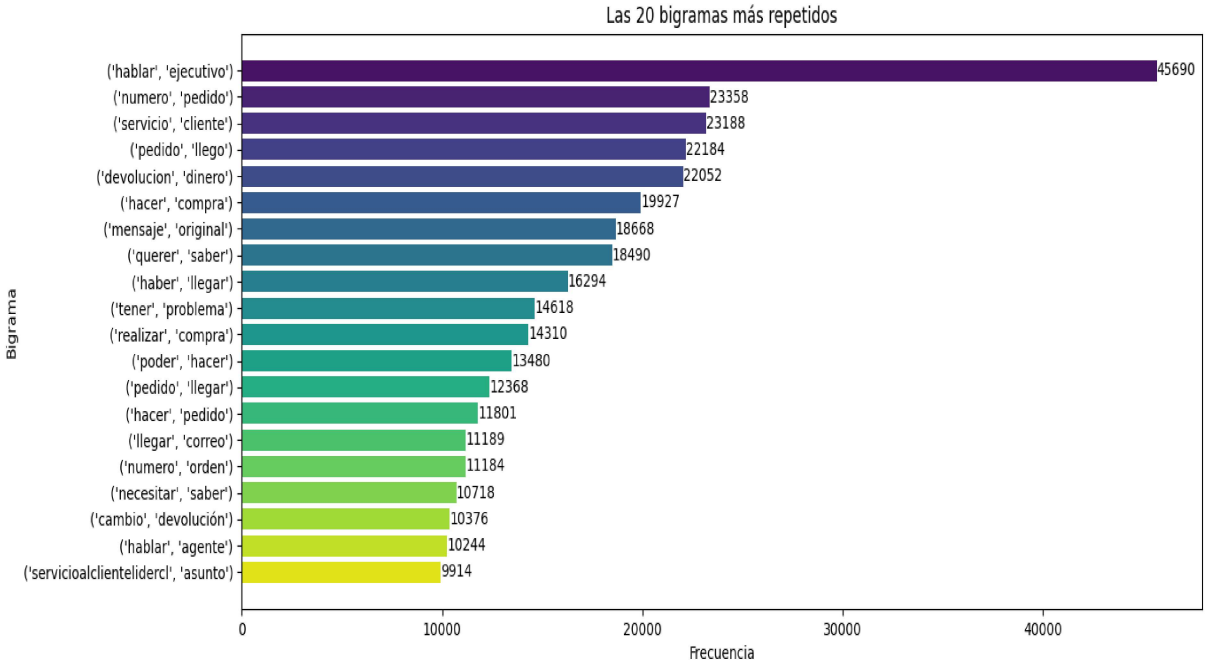

Figura 7.5: Gráfico de barras horizontal de los 20 bigramas mas frecuentes del muestreo 1

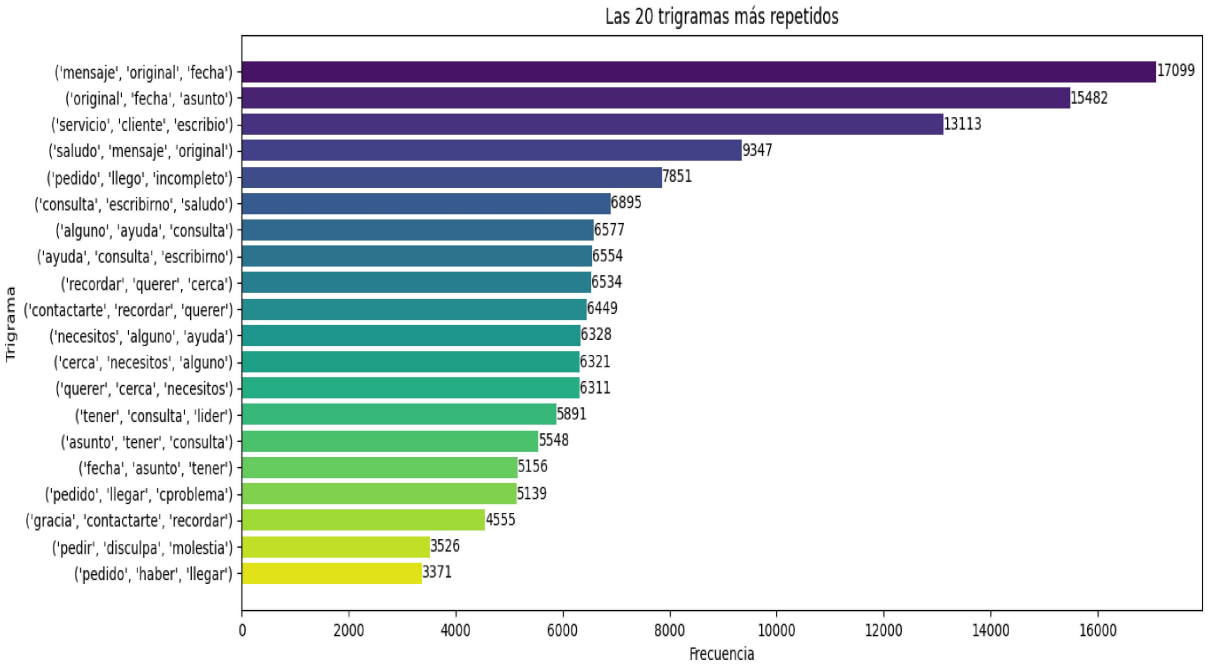

Figura 7.6: Gráfico de barras horizontal de los 20 trigramas mas frecuentes del muestreo 1

En base a la figura 7.4 se puede dar cuenta de una revelación de ciertas tendencias y patrones es ese corpus, el cual se puede expresar con la siguientes observaciones:
Se destacan términos como “pedido”, “compra” y “producto”, sugiriendo una alta frecuencia de referencias a transacciones y adquisiciones. También se identifican verbos comunes como

“poder”, “hacer”, “tener” y otros, indicando una expresión activa por parte de los usuarios. Expresiones de gratitud como “gracias” y “gracia”, junto con “hablar”, podrías sugerirminteracciones positivas. La presencia de “numero” podría asociarse con consultas numéricas, posiblemente relacionadas con números de pedido. Además, términos como “devolución”, “correo” y “comprar” podrían indicar solicitudes específicas. Este análisis proporciona insights útiles sobre los temas recurrentes en las interacciones, permitiendo futuros ajustes eficientes en la atención al cliente digital.

En base a la figura 7.5 se puede dar cuenta de una revelación de ciertas tendencias y patrones es ese corpus, el cual se puede expresar con la siguientes observaciones:
Frases como “hablar ejecutivo”, “numero pedido” y “servicio cliente” sugieren temas recurrentes, posiblemente relacionados con consultas específicas y atención al cliente. Otros bigramas como “pedido llego”, “devolución dinero” y “hacer compra” podrían estar indicado situaciones comunes en transacciones y compras. La presencia de combinaciones como “querer saber” y “haber llegar” podría estar reflejando consultas y estados de los usuarios. Este análisis proporciona una comprensión un poco más detallada de los tópicos más frecuentes, permitiendo una hipotética adaptación más efectiva en la gestión de las interacciones con los usuarios.

En base a la figura 7.6 se puede dar cuenta de una revelación de ciertas tendencias y patrones es ese corpus, el cual se puede expresar con la siguientes observaciones:
Expresiones como “mensaje original fecha”, “original fecha asunto” y “servicio cliente escribió” podría estar indicando la presencia de comunicaciones formales y consultas relacionadas con el servicio al cliente. Trigramas como “pedido llego incompleto” y “consulta escribirno saludo” podría sugerir situaciones específicas en las transacciones y consultas usuales. La recurrencia de combinaciones como “recordar querer cerca” y “cerca necesitos alguno” podría reflejar la necesidad de recordatorios y consultas cercanas. Este análisis proporciona una visión detallada de los tópicos más frecuentes, permitiendo una gestión más eficaz de las interacciones y adaptándose a las necesidades de los usuarios.

7.4.2. 2da Dimensión: Tipificación General de Tickets

7.4.2.1. Primera Parte: Selección del número óptimo, caracterización y asignación de los tópicos ocultos en el corpus

En el contexto de este proyecto, particularmente en el desarrollo de la segunda y tercera dimensión, que implican problemas de clasificación, se enfrenta un desafío significativo: la carencia de etiquetas preexistentes. La falta de etiquetas previas para el entrenamiento de clasificadores específicos en cada dimensión se presenta como un obstáculo. Por este motivo, al abordar la segunda dimensión, centrada en la resolución de una tarea de clasificación de texto, una problemática bien conocida en el ámbito del Procesamiento del Lenguaje Natural, se optó por emplear un modelo de aprendizaje automático no supervisado denominado Latent Dirichlet Allocation. Este modelo, reconocido como un enfoque probabilístico generativo, goza de amplia aceptación en el subcampo del Modelado de Tópicos y se aplicará al corpus mencionado anteriormente, denominado Muestreo 1. El proceso de desarrollo de esta dimensión consta de tres etapas: la selección del número óptimo de tópicos ocultos, la caracterización y la asignación de los mismos.

7.4.2.1.1. Selección del número óptimo de tópicos ocultos dentro del Muestreo 1

Para encontrar el número óptimo de tópicos ocultos dentro de los documentos del corpus en cuestión, se llevó el siguiente pipeline: Primero, se construye un diccionario (**id2word**) que asigna identificadores únicos a cada palabra en el corpus tokenizado. Luego, se prepara el corpus en un formato adecuado para el modelo LDA mediante la representación de bolsa de palabras (BoW) y se aplica la **transformación TF-IDF** para ponderar la importancia de las palabras en los documentos.
Posteriormente, se utiliza la biblioteca **Gensim** para implementar el **modelo LDA multicore**, variante del modelo LDA que aprovecha la capacidad de procesamiento paralelo. La función **'calc_coherence_values'** evalúa el modelo para diferentes cantidades de tópicos, tomando un mínimo de 2 tópicos y un máximo de 12, saltando de uno en uno, calculando la medida de coherencia de los tópicos (C_v) para cada iteración. Este proceso se realiza iterativamente para construir una lista de modelos y una lista de valores de coherencia correspondientes a cada número de tópicos evaluado. Obteniendo los siguientes resultados: A partir

Tabla 7.4: Resultados según número de tópicos y la métrica C_v

Número de Tópicos	Coherencia(C_v)
2	44 %
3	65 %
4	55 %
5	51 %
6	52 %
7	53 %
8	52 %
9	50 %
10	53 %

de la tabla 7.4, se infiere que el número óptimo de tópicos para el corpus correspondiente al dataset Muestreo 1 es 3. Arrojando el modelo los siguientes resultados:

```
[(0,
 '0.010*"saludo" + 0.008*"app" + 0.008*"consulta" + 0.005*"version" + 0.005*"servicio"
 + 0.004*"caso" + 0.004*"asunto" + 0.004*"recordar" + 0.004*"disculpa" + 0.004*"poder"'),
 (1,
 '0.008*"llego" + 0.008*"pedido" + 0.007*"faltar" + 0.007*"pedi" + 0.007*"incompleto"
 + 0.006*"carne" + 0.006*"pollo" + 0.006*"bolsa" + 0.005*"problema" + 0.005*"queso"'),
 (2,
 '0.007*"pedido" + 0.006*"llegar" + 0.006*"compra" + 0.006*"producto" + 0.005*"haber"
 + 0.005*"necesitar" + 0.005*"saber" + 0.005*"hacer" + 0.005*"hablar" + 0.005*"querer"')]
```

Cuya interpretación es la siguiente:

Tópico 0:

- **Palabras claves:** saludo, app, consulta, versión, caso, servicio, disculpa, recordar, poder, asunto.
- **La interpretación de las probabilidades que multiplican a las palabras claves** es la siguiente, si un documento del corpus tiene entre sus palabras:
  "saludo" tiene una probabilidad de aproximadamente 1 % de pertenecer a este tópico.
  "app" tiene una probabilidad de aproximadamente 0.8 % de pertenecer a este tópico.
  "consulta" tiene una probabilidad de aproximadamente 0.8 % de pertenecer a este tópico.
  "version" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
  "servicio" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
  "caso" tiene una probabilidad de aproximadamente 0.4 % de pertenecer a este tópico.
  "asunto" tiene una probabilidad de aproximadamente 0.4 % de pertenecer a este tópico.
  "recordar" tiene una probabilidad de aproximadamente 0.4 % de pertenecer a este tópico.
  "disculpa" tiene una probabilidad de aproximadamente 0.4 % de pertenecer a este tópico.
  "poder" tiene una probabilidad de aproximadamente 0.4 % de pertenecer a este tópico.

Tópico 1:

- **Palabras claves:** llego, pedido, faltar, pedi, incompleto, carne, pollo, bolsa, problema, producto.
- **La interpretación de las probabilidades que multiplican a las palabras claves** es la siguiente, si un documento del corpus tiene entre sus palabras:
  "llego" tiene una probabilidad de aproximadamente 0.8 % de pertenecer a este tópico.
  "pedido" tiene una probabilidad de aproximadamente 0.8 % de pertenecer a este tópico.
  "faltar" tiene una probabilidad de aproximadamente 0.7 % de pertenecer a este tópico.
  "pedi" tiene una probabilidad de aproximadamente 0.7 % de pertenecer a este tópico.
  "incompleto" tiene una probabilidad de aproximadamente 0.7 % de pertenecer a este tópico.
  "carne" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.
  "pollo" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.
  "bolsa" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.

"problema" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"queso" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.

Tópico 2:

- **Palabras claves:** pedido, llegar, compra, producto, necesitar, haber, saber, hacer, hablar, querer.
- **La interpretación de las probabilidades que multiplican a las palabras claves** es la siguiente, si un documento del corpus tiene entre sus palabras:
"pedido" tiene una probabilidad de aproximadamente 0.7 % de pertenecer a este tópico.
"llegar" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.
"compra" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.
"producto" tiene una probabilidad de aproximadamente 0.6 % de pertenecer a este tópico.
"haber" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"necesitar" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"saber" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"hacer" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"hablar" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.
"querer" tiene una probabilidad de aproximadamente 0.5 % de pertenecer a este tópico.

7.4.2.1.2. Caracterización de los tópicos ocultos descubiertos

En esta sección, se revelarán las etiquetas seleccionadas para identificar los tres tópicos descubiertos por el modelo LDA en la sección previa. La elección precisa de estas etiquetas se llevó a cabo mediante una reunión estratégica con el tutor del estudiante en Adereso, Camilo López, y Fernanda Cuéllar, Head of Sales en Adereso. El propósito principal de esta reunión fue proporcionar al estudiante una orientación valiosa para la selección de las etiquetas identificadoras de estos tres tópicos. Después de presentar los resultados obtenidos en la sección anterior y discutir las palabras clave asociadas a cada tópico, se llegó a un consenso sobre las siguientes etiquetas de manera colaborativa:

Tabla 7.5: Asignación de la etiqueta en base a resultados del topic modeling con LDA

Tópico	Top 10 Palabras Frecuentes	Etiqueta Escogida
Tópico 0	saludo, app, consulta, version, servicio, caso, asunto, recordar, disculpa, cliente	Consulta General/Administrativa
Tópico 1	llego, pedido, faltar, pedi, incompleto, carne, pollo, bolsa, problema, queso	Reclamo
Tópico 2	pedido, llegar, compra, producto, haber, necesitar, saber, hacer, querer, hablar	Consulta de Estado de Pedidos/Compras

A modo de conclusión, la asignación de etiquetas a cada tópico, derivada del proceso de modelado de tópicos con LDA, ha permitido identificar y categorizar patrones temáticos clave en los datos analizados. La elección de etiquetas, basada en la frecuencia de palabras en los tópicos, ha demostrado ser coherente con la semántica y el contenido asociado a cada grupo de documentos. Este enfoque proporciona una estructura organizativa valiosa que facilita la comprensión y clasificación eficiente de los tickets.

7.4.2.1.3. Asignación de los tópicos ocultos descubierto a los documentos del corpus

Para la correcta asignación de los tópicos ocultos descubierto a los documentos del corpus, el desarrollo se llevó a cabo de la siguiente manera:

Se importó la biblioteca necesaria, **pandas**, para realizar operaciones eficientes con conjuntos de datos estructurados. Posteriormente, se creó una función llamada **'assign_topic_to_document'** que toma un modelo de tópicos (en este caso, lda_model) y un corpus de documentos y asigna el tópico dominante a cada documento. Esto se logró iterando sobre cada documento en el corpus, utilizando el modelo de tópicos para obtener la distribución de tópicos y *seleccionando el tópico con la probabilidad más alta.*

Finalmente, se aplicó esta función al Muestreo 1, asignando así los tópicos dominantes a cada uno de los documentos en el conjunto de datos. La columna resultante llamada 'Tópico' ahora contiene la información sobre el tópico dominante para cada registro en el Muestreo 1. Teniendo la siguiente distribución:

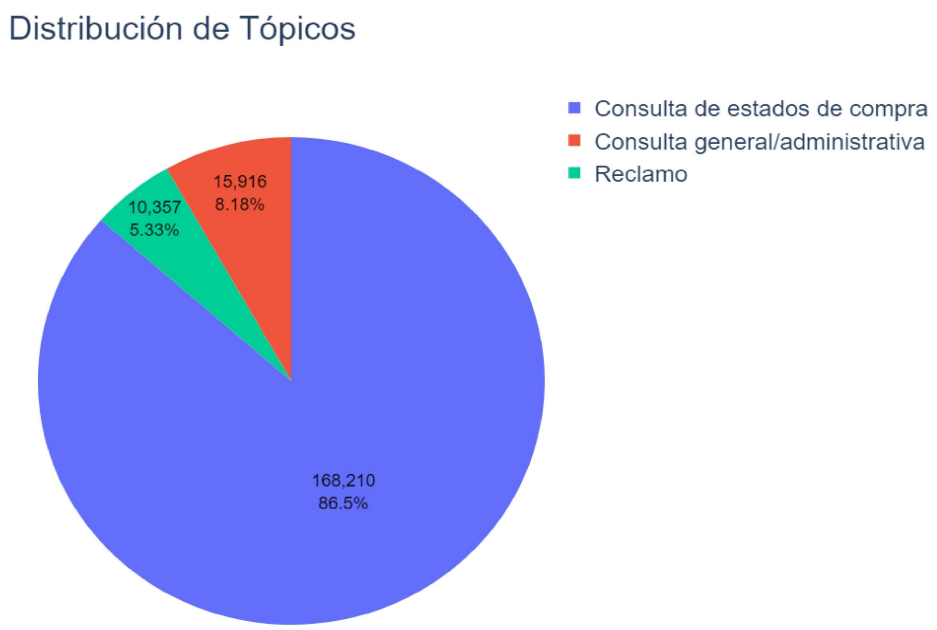

Figura 7.7: Gráfico de torta que muestra la distribución de los tópicos asignados en el conjunto de datos Muestreo 1

Finalmente, se visualizará un extracto de como quedaría la base de datos incorporando la información del tópico para cada uno de los registros del dataset Muestreo 1: De manera

Tabla 7.6: Extracto para visualizar como se guardan los campos referentes a la asignación de tópico de los tickets

Ticket	Texto	Texto Lematizado	Texto Lematizado tokenizado	Tópico	Nombre de Tópico
8707500	Esta con cinta de embalaje y le faltan cervezas	cinta embalaje faltar cervezas	['cinta', 'embalaje', 'faltar', 'cervezas']	1	Reclamo

análoga, para el tópico 0 la columna 'Nombre de Tópico' Consulta General/Administrativa y para el tópico 2 la columna 'Nombre de Tópico' Consulta de Estado de Pedidos/Compras.

7.4.2.2. Segunda Parte: Desarrollo de un clasificador para las etiquetas descubiertas

Con base en los resultados obtenidos en la primera fase del modelado de la segunda dimensión del prototipo propuesto, se confirma la disponibilidad de un conjunto de datos (Muestreo 1) que ha sido etiquetado y presenta alta calidad. Este muestreo abarca información relacionada con tickets procesados durante el período de enero de 2023 a junio de 2023. En este contexto, surge la idea natural de desarrollar un clasificador de etiquetas para automatizar la tipificación de los tickets procesados a partir de julio de 2023.

Con el objetivo de llevar a cabo este proceso, el estudiante consideró relevante realizar experimentos con dos enfoques distintos: un clasificador basado en una red neuronal LSTM (Long Short-Term Memory) y otro basado en un modelo transformers denominado 'distilbert-base-uncased', obtenido de la plataforma Hugging Face. Los detalles de los procesos de modelado y los resultados correspondientes se presentarán en las secciones siguientes.

7.4.2.3. Desarrollo del clasificador basado en una red neuronal recurrente variante LSTM

En esta sección, abordaremos el desarrollo del clasificador basado en una red neuronal recurrente de tipo LSTM. A continuación, se explicará detalladamente el modelado de este clasificador.

En primer lugar, se importan las bibliotecas necesarias para el procesamiento de datos y la creación del modelo. TensorFlow y Keras se utilizan para la implementación de la red neuronal, Pandas para manipular datos tabulares, y otras bibliotecas estándar como NumPy.[Keras] Luego, se realiza la carga de datos desde un archivo CSV que contiene el Muestreo 1, incorporando las columnas que asignan la etiqueta correspondiente al ticket según los tres tópicos mencionados previamente. Cabe destacar que este conjunto de datos abarca un poco más de 194 mil tickets y corresponde al período comprendido entre enero de 2023 y junio de 2023.
Seguidamente, se utiliza una función para contar las palabras únicas presentes en los textos

lematizados del conjunto de datos, es decir, tener conocimiento del vocabulario del corpus inmerso en el Muestreo 1. Esto proporciona información sobre la distribución de palabras en el corpus.
En cuanto a la preparación de los datos de entrenamiento y validación, primero se determina el tamaño del conjunto de entrenamiento como el 80 % del tamaño total del conjunto de datos original. Luego, se crea el conjunto de entrenamiento tomando las primeras filas hasta el tamaño definido. Posteriormente, se forma el conjunto de validación seleccionando las filas restantes después del conjunto de entrenamiento (20 %). Este enfoque asegura que no haya solapamiento entre los datos de entrenamiento y validación. Luego, se extraen las columnas relevantes ('Texto Lematizado' y 'Tópico') de los conjuntos de entrenamiento y validación para obtener las secuencias de texto y las etiquetas asociadas.
A continuación, se realiza la tokenización y el ajuste de las secuencias de texto para que puedan ser procesadas por el modelo. Para ello, se utiliza la clase 'Tokenizer' de TensorFlow.keras para convertir las secuencias de texto en secuencias de tokens. La propiedad 'num_words' especifica el número máximo de palabras a tener en cuenta, basándose en la frecuencia de ocurrencia en el conjunto de entrenamiento y validación. Esto permite una representación numérica eficiente de las palabras. Luego, el Tokenizer se ajusta utilizando el método 'fit_on_texts' aplicado al conjunto de entrenamiento y validación. Después, se obtiene un índice que mapea cada palabra única en el conjunto de entrenamiento y validación a un número entero, conocido como el índice de palabra (word_index). Finalmente, se transforman las secuencias de texto a secuencias numéricas mediante el método 'text_to_sequences', donde cada palabra en el texto se reemplaza por su correspondiente número entero de acuerdo con el índice de palabras aprendido. En último lugar, se realiza un padding a las secuencias de entrenamiento y validación para asegurar que todas tengan la misma longitud. El padding se realiza al final de cada secuencia utilizando ceros como valores de relleno, y el parámetro 'maxlen' especifica la longitud máxima deseada para las secuencias, establecido en 35. Esto da como resultado los conjuntos de entrenamiento y validación, 'train_padded' y 'validation_padded', respectivamente, que son secuencias numéricas representativas de los tickets pero en formato de texto.
Luego, se procede a la creación del modelo, el cual incluye una capa de embedding que utiliza 32 dimensiones para representar cada palabra. Este valor, junto con otros hiperparámetros, se elige empíricamente, afectando la capacidad del modelo para aprender representaciones semánticas y la optimización del gasto computacional. La segunda capa es una capa LSTM con 64 unidades y un dropout del 0.1, que ayuda a prevenir el sobreajuste al apagar aleatoriamente algunas unidades durante el entrenamiento. La capa densa de salida tiene 3 nodos correspondientes a las tres clases de tópicos y utiliza la función de activación softmax para asignar probabilidades a cada clase.
El modelo se compila con la función de pérdida CategoricalCrossentropy, un optimizador Adam con tasa de aprendizaje 1e-3, y métricas de accuracy.

Finalmente, se entrena el modelo utilizando el conjunto de entrenamiento ('train_padded') con las etiquetas codificadas one-hot. El proceso se repite durante 20 épocas. Luego, se evalúa el modelo utilizando el conjunto de validación ('validation_padded'), y se calcula una matriz de confusión y otras métricas de clasificación como la precisio, recall y F1-score.

El modelo antes descrito, con las clases originalmente desbalanceadas, obtuvo el siguiente rendimiento en el conjunto de entrenamiento:

Tabla 7.7: Resultados de la fase de entrenamiento para el modelo RNN LSTM con las clases desbalanceadas

Modelo	Train Accuracy	Train Loss
RNN LSTM clases desbalanceadas	99.96 %	0.0012

Los resultados en el conjunto de validación de este modelo con las clases desbalanceadas es el siguiente:

Tabla 7.8: Resultados de la fase de validación para el modelo RNN LSTM con las clases desbalanceadas

Modelo	Val Accuracy	Val Loss	Precision	Recall	F1-Score
DistilBERT clases desbalanceadas	92.85 %	0.5510	79.33 %	68.67 %	72.33 %

Obteniendo la siguiente matriz de confusión:

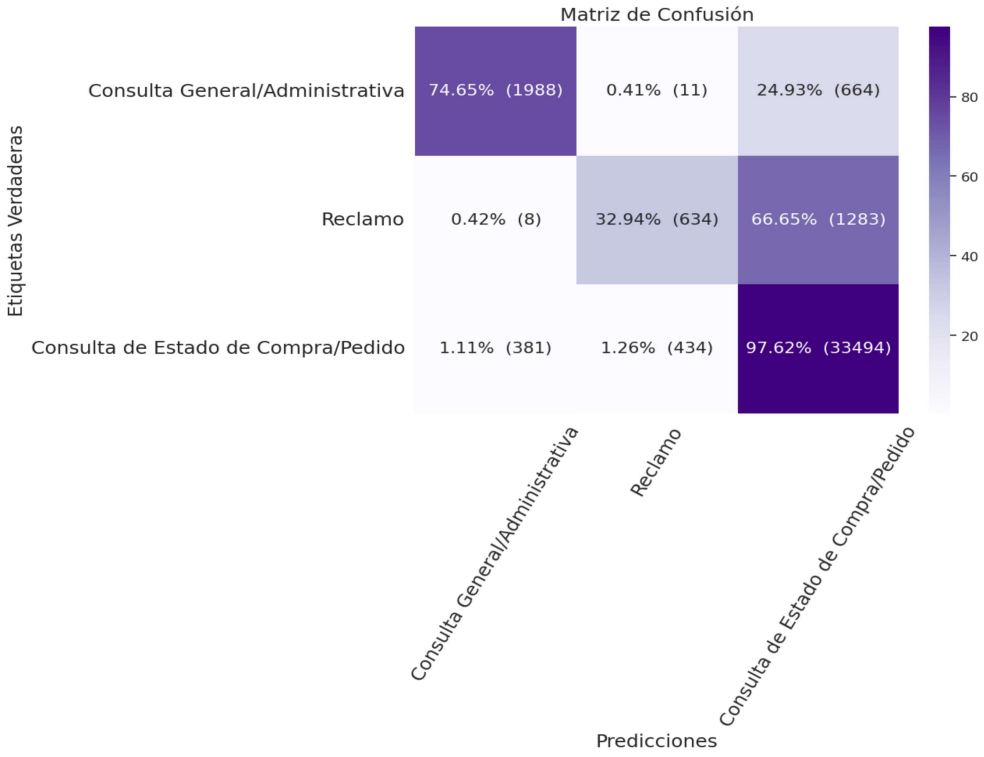

Figura 7.8: Matriz de confusión para el modelo RNN LSTM con clases desbalanceadas

La matriz de confusión presentada en la figura 7.8refleja el desempeño de un modelo de clasificación en tres categorías específicas: 'Consulta General/Administrativa', 'Reclamo' y 'Consulta de Estado de Compra/Pedido'. En el análisis detallado de la matriz, se observan diversas métricas que permiten evaluar la capacidad del modelo para realizar predicciones precisas en cada una de estas categorías.

En relación con la clase 'Consulta General/Administrativa', se registran 1988 verdaderos positivos, indicando que el modelo ha identificado correctamente estas instancias. Sin embargo, se observan 11 falsos negativos, lo que implica que el modelo ha dejado de reconocer algunas instancias reales de esta clase. Asimismo, se han producido 664 falsos positivos, señalando casos en los que el modelo ha clasificado erróneamente instancias como 'Consulta General/Administrativa'.
En cuanto a la categoría 'Reclamo', el modelo ha logrado identificar 634 verdaderos positivos, aunque ha pasado por alto 8 instancias reales de esta clase, como indican los falsos negativos. Por otro lado, se observa un número significativo de falsos positivos, específicamente 1283, revelando situaciones en las que el modelo ha clasificado erróneamente instancias como reclamos.
Respecto a 'Consulta de Estado de Compra/Pedido', se destaca un rendimiento sólido con 33494 verdaderos positivos, lo que indica una capacidad eficiente del modelo para identificar correctamente esta categoría. Sin embargo, se han registrado 815 falsos negativos, señalando instancias no reconocidas, y 815 falsos positivos, indicando casos en los que el modelo ha clasificado erróneamente instancias como 'Consulta de Estado de Compra/Pedido'.

En resumen, la matriz de confusión proporciona una visión detallada del rendimiento del modelo en cada clase. La interpretación se centra en la capacidad del modelo para acertar en las predicciones (verdaderos positivos), así como en los errores cometidos al clasificar incorrectamente instancias (falsos positivos y falsos negativos). En este caso, se observa un buen rendimiento en algunas clases, pero también algunos desafíos, especialmente en la clasificación de reclamos.

Efectivamente, adicionalmente se realizó una variante con las clases balanceadas según la clase minoritaria(Reclamo), obteniendo ahora un conjunto de datos de 31071 tickets, teniendo 10357 de cada tópico. Se realizó el desarrollo de este nuevo clasificador de manera análoga al caso anterior, obteniendo los siguientes resultados en el conjunto de entrenamiento:

Tabla 7.9: Resultados de la fase de entrenamiento para el modelo RNN LSTM con las clases balanceadas

Modelo	Train Accuracy	Train Loss
RNN LSTM clases balanceadas	99.90 %	0.0035

Los resultados en el conjunto de validación de este modelo con las clases balanceadas es el siguiente:

Tabla 7.10: Resultados de la fase de validación para el modelo RNN LSTM con las clases balanceadas

Modelo	Val Accuracy	Val Loss	Precision	Recall	F1-Score
DistilBERT clases balanceadas	89.93 %	0.6260	90.00 %	89.67 %	89.67 %

Obteniendo la siguiente matriz de confusión:

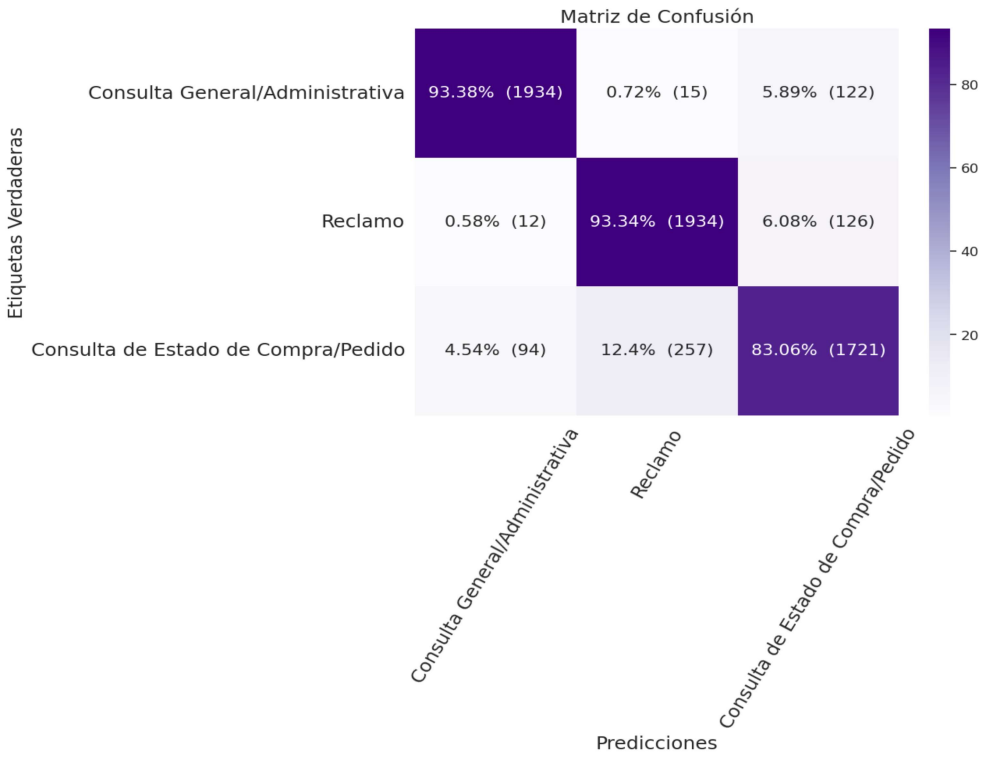

Figura 7.9: Matriz de confusión para el modelo RNN LSTM con clases balanceadas

La matriz de confusión presentada en la figura 7.9, proporciona una visión detallada del rendimiento de un modelo de clasificación en tres categorías específicas: 'Consulta General/Administrativa', 'Reclamo' y 'Consulta de Estado de Compra/Pedido'. Al analizar los elementos de la matriz, se obtienen valiosas métricas de evaluación del modelo.
Para la clase 'Consulta General/Administrativa', se observan 1934 verdaderos positivos, indicando que el modelo ha acertado en clasificar correctamente instancias de esta categoría. Los falsos negativos son bajos (15), lo que sugiere que el modelo rara vez pasa por alto instancias reales de 'Consulta General/Administrativa'. Sin embargo, se identifican 122 falsos positivos, señalando casos en los que el modelo ha clasificado incorrectamente instancias como 'Consulta General/Administrativa'.
En relación con la categoría 'Reclamo', se destaca nuevamente un alto número de verdaderos positivos (1934) y un bajo número de falsos negativos (12), indicando una eficacia en la identificación de instancias reales de 'Reclamo'. Sin embargo, se registran 126 falsos positivos, revelando instancias mal clasificadas como reclamos.
Para la clase 'Consulta de Estado de Compra/Pedido', se observa un rendimiento sólido con 1721 verdaderos positivos, indicando una capacidad eficiente del modelo para identificar correctamente esta categoría. Los falsos negativos (94) y falsos positivos (257) son relativamente bajos, sugiriendo un buen equilibrio en la predicción de esta clase.

En resumen, la matriz de confusión revela un rendimiento general positivo del modelo con énfasis en la identificación precisa de las categorías 'Consulta General/Administrativa' y 'Reclamo'. Se observan desafíos en la minimización de falsos positivos en estas clases, lo que podría ser considerado en ajustes futuros del modelo para mejorar su precisión. Este análisis contribuye a una comprensión más profunda de la capacidad del modelo para clasificar

correctamente las distintas categorías.

7.4.2.4. Desarrollo del clasificador basado en el modelo transformer distilBERT

En esta sección, abordaremos el desarrollo del clasificador basado en el modelo transformer DistilBERT. DistilBERT, derivado de "Distilled BERT", representa una versión optimizada y más ligera del poderoso modelo BERT, desarrollado por Hugging Face. El modelo específico que emplearemos, 'distilbert-base-uncased', es una variante preentrenada que opera sin distinguir entre mayúsculas y minúsculas, siendo designada como "base"debido a su nivel intermedio en términos de parámetros y capacidad. Esta versión ha sido destilada para preservar la eficacia semántica de BERT con un menor número de parámetros, lo que resulta en una mayor eficiencia computacional. En el contexto de nuestras tareas de procesamiento de lenguaje natural, 'distilbert-base-uncased' se revela como un recurso valioso para aplicaciones como clasificación de texto y generación de texto, gracias a su capacidad para capturar representaciones semánticas precisas. A continuación, detallaremos el proceso de desarrollo del clasificador, desde la preparación del conjunto de datos hasta la evaluación del rendimiento del modelo.[HuggingFace, 2020]

Continuando con el desarrollo del clasificador, el proceso avanza con la preparación de datos y la construcción del modelo de clasificación.
Se inicia instalando versiones específicas de las bibliotecas **ktrain** y **transformers** mediante los comandos *pip install*. Posteriormente, se importan las librerías necesarias, incluyendo **pandas** para el manejo de datos, **ktrain** para el aprendizaje profundo de modelos de lenguaje, y **scikit-learn** para funciones auxiliares de procesamiento de datos y evaluación de modelos.
La carga de datos se realiza a partir de un archivo CSV que contiene información con el Muestreo 1 añadiendo las columnas que etiquetan el ticket según los 3 tópicos antes descritos. Recordar que este dataset contiene poco mas de 194 mil tickets, referentes al período Enero 2023 - Junio 2023.
Este conjunto de datos se divide en conjuntos de entrenamiento y prueba utilizando la función **'train_test_split'** de **scikit-learn**. En este caso, se asigna el 80 % de los datos para entrenamiento(155586 tickets) y el 20 % para validación(38897 tickets), con una semilla aleatoria fijada en 42 para garantizar reproducibilidad.
Las categorías a predecir se definen en una lista llamada **categories**, que representa las etiquetas asignadas a cada tópico de los tickets, como *"Consulta General/Administrativa","Reclamo" y "Consulta de Estado de Pedidos/Compra".*
Se utiliza el modelo de lenguaje **'distilbert-base-uncased'**, que es una versión más ligera y eficiente de BERT. La biblioteca ktrain facilita el preprocesamiento de los datos de entrenamiento y prueba a través de la clase **'text.Transformer'**. El cual toma como largo máximo de un ticket, 512 tokens.
El modelo de clasificación se obtiene mediante **'get_classifier()'**, y la fase de entrenamiento se realiza utilizando el método de aprendizaje con un ciclo (fit_onecycle). Este método ajusta el learning rate durante el entrenamiento, partiendo de un valor bajo y aumentándolo gradualmente, lo que puede mejorar la convergencia y evitar problemas como el sobreajuste. En donde el lr de valor bajo se define en $1e-4$ y el lr de valor alto en 1. Y los procesa en batches de a 16 registros del conjunto de entrenamiento.
Finalmente, se evalúa el rendimiento del modelo en el conjunto de prueba mediante el método validate(), permitiendo una evaluación integral de su capacidad predictiva. Evalúandolo

mediante las métricas de accuracy y macro avg de recall, precision y f1-score.
Los resultados del entrenamiento de este modelo con las clases desbalanceadas es el siguiente:

Tabla 7.11: Resultados de la fase de entrenamiento para el modelo distilBERT con las clases desbalanceadas

Modelo	Train Accuracy	Train Loss
DistilBERT clases desbalanceadas	97.74 %	0.1352

Los resultados en el conjunto de validación de este modelo con las clases desbalanceadas es el siguiente:

Tabla 7.12: Resultados de la fase de validación para el modelo distilBERT con las clases desbalanceadas

Modelo	Val Accuracy	Val Loss	Precision	Recall	F1-Score
DistilBERT clases desbalanceadas	96.50 %	0.0849	90.00 %	89.00 %	89.00 %

Y se obtuvo la siguiente matriz de confusión:

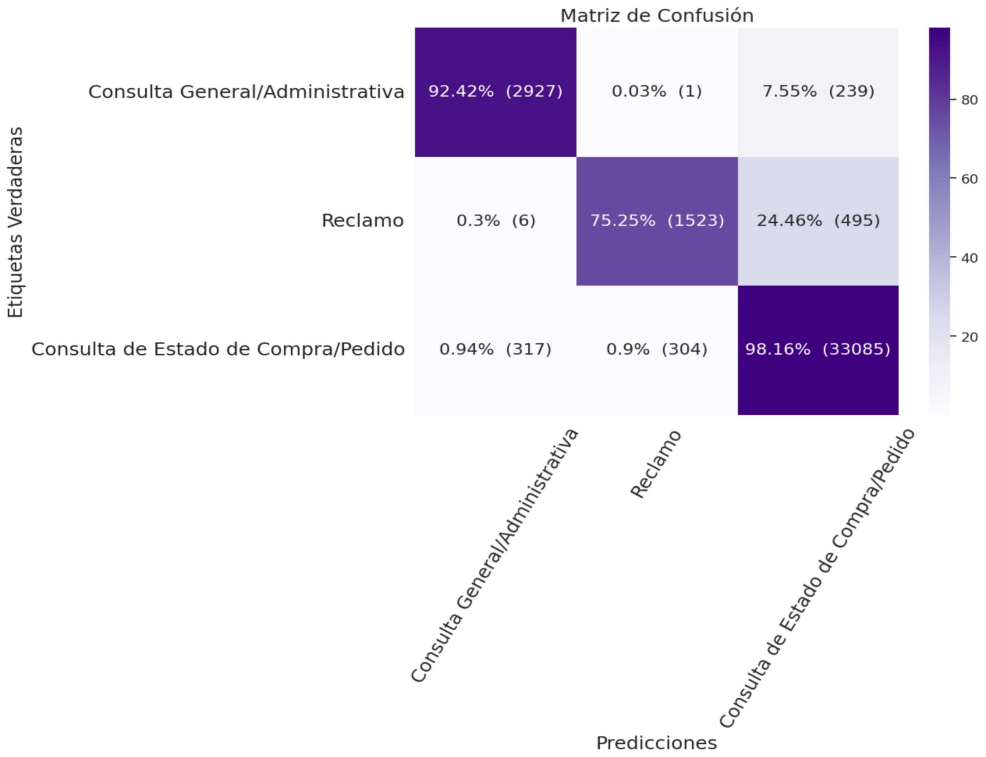

Figura 7.10: Matriz de confusión para el modelo distilBERT con clases desbalanceadas

La interpretación que se puede realizar de la figura 7.10 es la siguiente:
En la evaluación detallada de la matriz de confusión, se destaca el rendimiento del modelo en

las tres clases principales. En el caso de la categoría 'Consulta General/Administrativa', el modelo muestra un desempeño destacado al prever la clase, con un elevado número de verdaderos positivos y una mínima omisión de casos (falsos negativos). Sin embargo, la presencia de algunos falsos positivos sugiere la posibilidad de predicciones incorrectas en esta categoría en ciertas instancias.
Para la categoría de 'Reclamo', se destaca un rendimiento positivo, caracterizado por un bajo número de falsos negativos, indicando que el modelo rara vez pasa por alto instancias de esta clase. Aunque se identifican algunos falsos positivos, señalando cierta confusión con otras clases, la capacidad predictiva general es positiva.
En cuanto a 'Consulta de Estado de Pedido/Compra', la clase presenta un número sustancialmente alto de verdaderos positivos, reflejando una sólida capacidad del modelo para predecir esta categoría. Los falsos positivos y falsos negativos son relativamente bajos, indicando un rendimiento equilibrado y sólido en la predicción de esta clase.
En resumen, la matriz de confusión sugiere un rendimiento positivo del modelo, especialmente en las clases de 'Consulta General/Administrativa' y 'Consulta de Estado de Pedido/Compra'. Se plantea la posibilidad de explorar estrategias para reducir los falsos positivos en las clases correspondientes, mediante ajustes específicos del modelo, como lo son el hecho de balancear el modelo.

Efectivamente, adicionalmente se realizó una variante con las clases balanceadas según la clase minoritaria(Reclamo), obteniendo ahora un conjunto de datos de 31071 tickets, teniendo 10357 de cada tópico. Se realizó el desarrollo de este nuevo clasificador de manera análoga al caso anterior, obteniendo los siguientes resultados en el conjunto de entrenamiento:

Tabla 7.13: Resultados de la fase de entrenamiento para el modelo distilBERT con las clases balanceadas

Modelo	Train Accuracy	Train Loss
DistilBERT clases balanceadas	96.15 %	0.2415

Los resultados en el conjunto de validación de este modelo con las clases balanceadas es el siguiente:

Tabla 7.14: Resultados de la fase de validación para el modelo distilBERT con las clases desbalanceadas

Modelo	Val Accuracy	Val Loss	Precision	Recall	F1-Score
DistilBERT clases desbalanceadas	95.16 %	0.1324	95.00 %	95.00 %	95.00 %

Y se obtuvo la siguiente matriz de confusión:

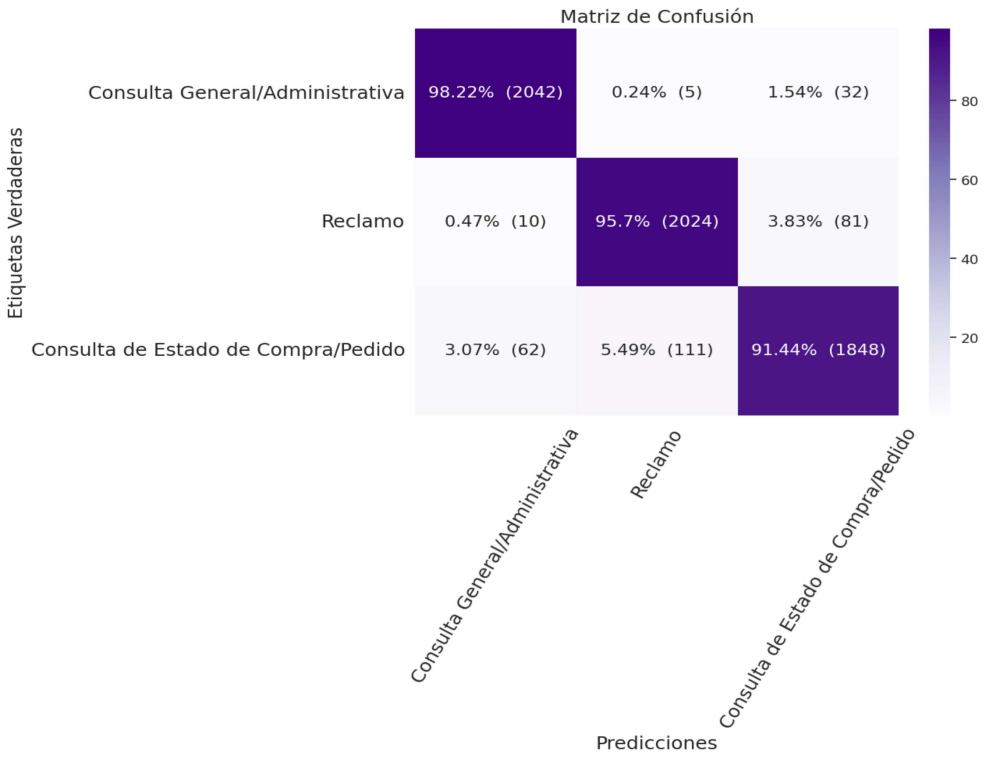

Figura 7.11: Matriz de confusión para el modelo distilBERT con clases balanceadas

La interpretación que se puede realizar de la figura 7.11 es la siguiente:
En la evaluación detallada de la matriz de confusión, se destaca el rendimiento del modelo en las tres clases principales. En la clase 'Consulta General/Administrativa', se observa un número sustancialmente alto de verdaderos positivos (VP), indicando que el modelo logra predecir de manera correcta esta categoría en la mayoría de los casos. La cantidad baja de falsos negativos (FN) sugiere que el modelo rara vez pasa por alto instancias de esta clase. No obstante, se identifican algunos falsos positivos (FP), indicando ocasiones en las que se predice incorrectamente 'Consulta General/Administrativa'.
En la clase 'Reclamo', nuevamente se registra un número elevado de verdaderos positivos, señalando una efectiva capacidad del modelo para predecir esta categoría. Los falsos positivos son ligeramente más prominentes que en la clase anterior, lo que sugiere cierta confusión ocasional con otras clases.
Por último, en la clase 'Consulta de Estado de Pedido/Compra', se destaca un alto número de verdaderos positivos, con bajos valores de falsos positivos y falsos negativos. Este equilibrio indica una sólida capacidad del modelo para predecir con precisión la categoría 'Consulta de Estado de Pedido/Compra'.
En resumen, la matriz de confusión refleja un rendimiento general positivo del modelo, con un énfasis particular en las clases 'Consulta General/Administrativa' y 'Reclamo'. A pesar de ello, se sugiere la exploración de estrategias para reducir los falsos positivos en estas clases, adaptándose a las necesidades específicas del problema que aborda el modelo.

7.4.3. 3era Dimensión: Análisis de sentimiento de los tickets

En el contexto de este proyecto, específicamente en las etapas de desarrollo relacionadas con la segunda y tercera dimensión, que abordan problemas de clasificación, se enfrenta un desafío considerable: la ausencia de etiquetas preexistentes. La carencia de etiquetas previas para entrenar clasificadores específicos en cada dimensión se presenta como un obstáculo significativo. Por esta razón, al abordar la tercera dimensión, centrada en la resolución de una tarea de análisis del sentimiento, una problemática bien conocida en el ámbito del Procesamiento del Lenguaje Natural, se optó por utilizar la API de OpenAI y el modelo GPT-3.5 Turbo para inferir el sentimiento de los tickets en una escala de Negativo, Neutral o Positivo. Este enfoque se aplicará al corpus mencionado anteriormente, denominado Muestreo 2, por las razones antes mencionadas. El proceso de desarrollo de esta dimensión consta de varias etapas, que incluyen el desarrollo del prompt utilizando técnicas de prompt engineering, la implementación de este prompt como indicación para que el modelo GPT-3.5 Turbo infiera el sentimiento de cada documento en el corpus y la evaluación del rendimiento de este modelo en esta tarea específica.

7.4.3.1. Desarrollo del prompt mediante técnicas del prompt engineering

El “prompt engineering” se refiere al proceso de diseñar o formular cuidadosamente las instrucciones o comandos (llamados "prompts") que se utilizan para interactuar con modelos de lenguaje, como GPT-3.5. El objetivo es maximizar el rendimiento del modelo, ajustando los prompts para obtener respuestas específicas y útiles, y así sacar el mayor provecho de las capacidades del modelo. Las técnicas ocupadas fueron Role-Prompting, Contexto, Few-Shot Prompting, Indicación y Pregunta Clara y se definen a continuación:

Role-Prompting (Indicación de Rol):

- *Definición:* Esta técnica implica asignar roles específicos a las partes involucradas en la interacción con el modelo. Al indicar claramente el rol de cada entidad en el prompt, se busca obtener respuestas coherentes y contextualmente relevantes.

Contexto:

- *Definición:* La inclusión de información contextual en el prompt es fundamental en el prompt engineering. Proporcionar antecedentes o detalles relevantes puede ayudar al modelo a comprender mejor la solicitud y generar respuestas más precisas.

Few-Shot Prompting (Indicación con Pocas Ejemplos):

- *Definición:* Esta técnica implica presentar al modelo algunos ejemplos de entrada y salida deseados antes de la consulta principal. Con solo unos pocos ejemplos, se espera que el modelo capture el patrón y proporcione respuestas coherentes en función de esos ejemplos.

Indicación:

- *Definición:* El uso de indicadores claros y específicos en el prompt ayuda a guiar al modelo hacia la generación de respuestas específicas. Puede implicar el uso de palabras clave o instrucciones explícitas para dirigir la atención del modelo hacia ciertos aspectos de la consulta.

Pregunta Clara:

- *Definición:* Formular preguntas de manera clara y precisa es esencial en el prompt engineering. Proporcionar una estructura clara en la pregunta puede ayudar al modelo a comprender mejor la solicitud y producir respuestas más relevantes y útiles.

Tomando como base estas técnicas se diseño el siguiente prompt:
"Imagina que eres un asistente virtual especializado en análisis de sentimientos para interacciones cliente-agente en la industria del retail.(Role Prompting)
Como el asistente especializado, analizarás el siguiente mensaje y clasificarás como negativo, neutral o positivo, considerando el contexto del retail, el mensaje que estará entre comillas.(Contexto)
Aquí hay algunos ejemplos para que aprendas sobre las clasificaciones de sentimientos específicas al retail:
- Mi pedido se ha retrasado mucho: [Mensaje con clasificación: Negativo]
- Quiero cambiar mi clave de la aplicación web: [Mensaje con clasificación: Neutral]
- Mi pedido llegó satisfactoriamente y a tiempo: [Mensaje con clasificación: Positivo] (Few-Shot Prompting)
Analiza el siguiente mensaje: "mensaje". ¿Cómo clasificarías su sentimiento en la escala de negativo, neutral o positivo? (Indicación)
Dame tu respuesta es una sola palabra, ya sea: Negativo, Neutral y Positivo " (Pregunta Clara)

7.4.3.2. Inferencia del sentimiento de los tickets a partir del prompt diseñado

En primer lugar, la función getCompletion se encarga de interactuar con el modelo para obtener la respuesta completa a partir de un prompt dado. Esta función configura adecuadamente el formato del mensaje de entrada y utiliza la API de OpenAI para obtener una respuesta del modelo, estableciendo la temperatura en 0 para asegurar determinismo en la salida del modelo.

La función principal, analizarSentimiento, utiliza la función anterior para evaluar el sentimiento de un mensaje específico en un conjunto de datos. Extrae el mensaje de la fila del conjunto de datos bajo la columna "Texto Limpioz lo trunca a una longitud máxima de 4000 caracteres. Luego, se aplica el prompt antes diseñado, que presenta al modelo GPT-3.5 Turbo, describiendo al modelo que imagine ser un asistente virtual especializado en análisis de sentimientos en la industria del retail. Solicita al modelo clasificar el sentimiento del mensaje como negativo, neutral o positivo en el contexto del retail, y finalmente, captura la respuesta generada por el modelo.

Este proceso de análisis de sentimientos se aplica a cada fila del conjunto de datos utilizando la función apply en el DataFrame Muestreo 2. Los resultados de las clasificaciones de sentimientos son almacenados en una nueva columna llamada "Sentimiento", enriqueciendo así el conjunto de datos con información contextualizada sobre la polaridad emocional de cada mensaje en el contexto del retail.

Obteniendo los siguientes resultados:

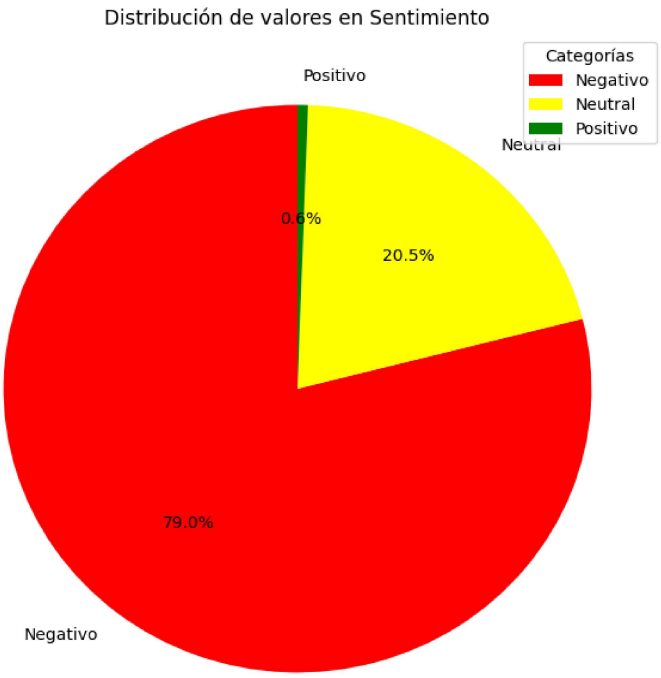

Figura 7.12: Pie chart de la distribución de la columna 'Sentimiento' en el dataframe Muestreo 2

La figura 7.12 muestra una distribución desigual en las etiquetas de sentimiento en las conversaciones entre clientes y agentes de empresas de retail en el corpus analizado. La mayoría de los documentos, con un 79.0 %, están etiquetados con sentimiento negativo, indicando posiblemente la presencia de situaciones insatisfactorias o problemas en las interacciones. Un 20.5 % se clasifica como neutral, sugiriendo un tono más neutro o informativo en algunas conversaciones. En contraste, solo un pequeño porcentaje, el 0.6 %, presenta una etiqueta de sentimiento positivo, lo que podría reflejar experiencias satisfactorias o comentarios positivos en estas interacciones. Este patrón revela la predominancia de tonos negativos en las conversaciones, lo cual podría requerir un análisis más detallado para comprender las dinámicas específicas entre clientes y agentes en el contexto del retail.

7.4.3.3. Evaluación del rendimiento de este modelo en esta tarea específica de análisis del sentimiento

Uno de las mayores dificultades que se tenían para evaluar el rendimiento del modelo GPT 3.5 Turbo en esta tarea específica, era el hecho de no tener etiquetas preexistentes con la cual comparar las inferencias que había hecho el modelo. En respuesta a esto, el estudiante etiquetó manualmente el Muestreo 2, con el fin de poder medir la performance de este LLM en esta tarea específica, incorporando las metricas de accuracy y precision, recall y f1-score macro avg:

Tabla 7.15: Resultados de la performance del modelo GPT 3.5 Turbo para la tarea de análisis del sentimiento comparadas con etiquetas manuales

Modelo	Val Accuracy	Precision	Recall	F1-Score
GPT 3.5 Turbo	95.00 %	96.00 %	73.00 %	80.00 %

7.4.4. Propuesta de Visualización de Analítica

En esta sección, se abordará una proposición en cuanto al tipo de gráficos que se deberían incluir en la visualización que recaerá en el usuario final de esta analítica. La data que se tomará en cuenta para desarrollar esta propuesta es la del Muestreo 2

7.4.4.1. Gráficos Referentes a la Primera Dimensión

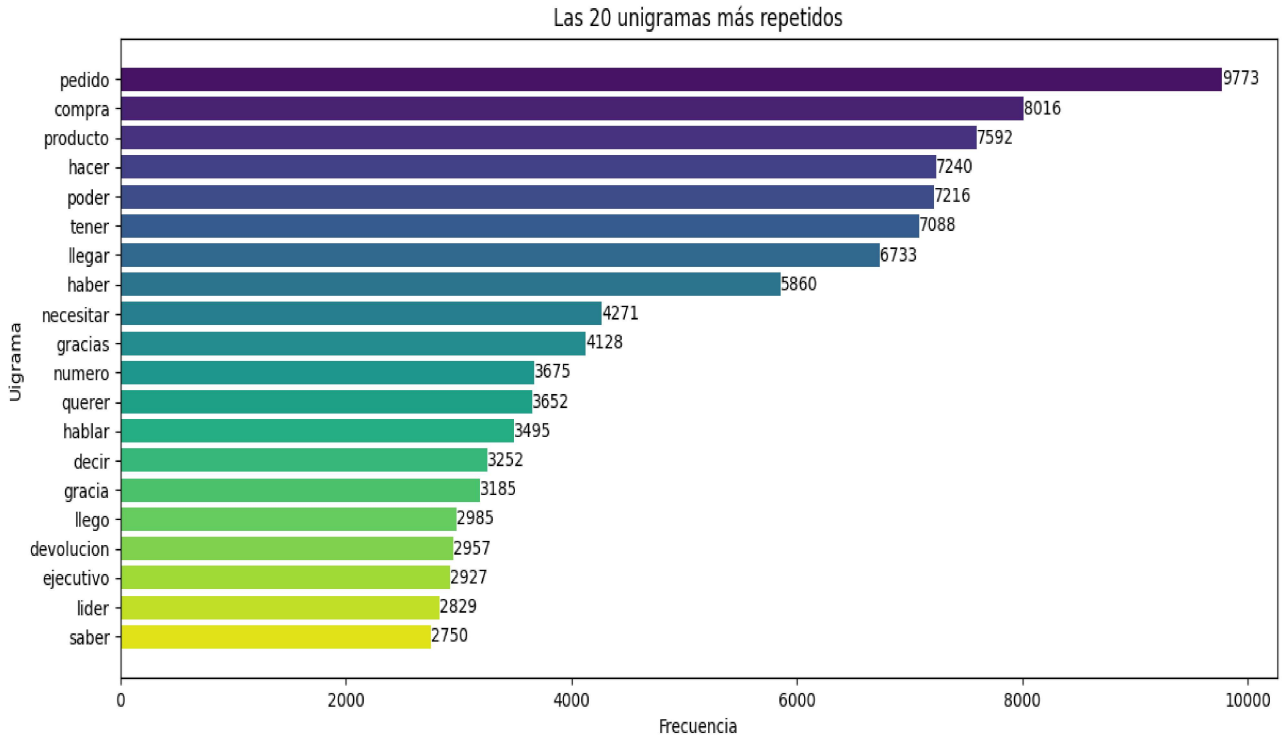

Figura 7.13: Bar Chart de los 20 unigramas más frecuentes en el dataframe de Muestreo 2

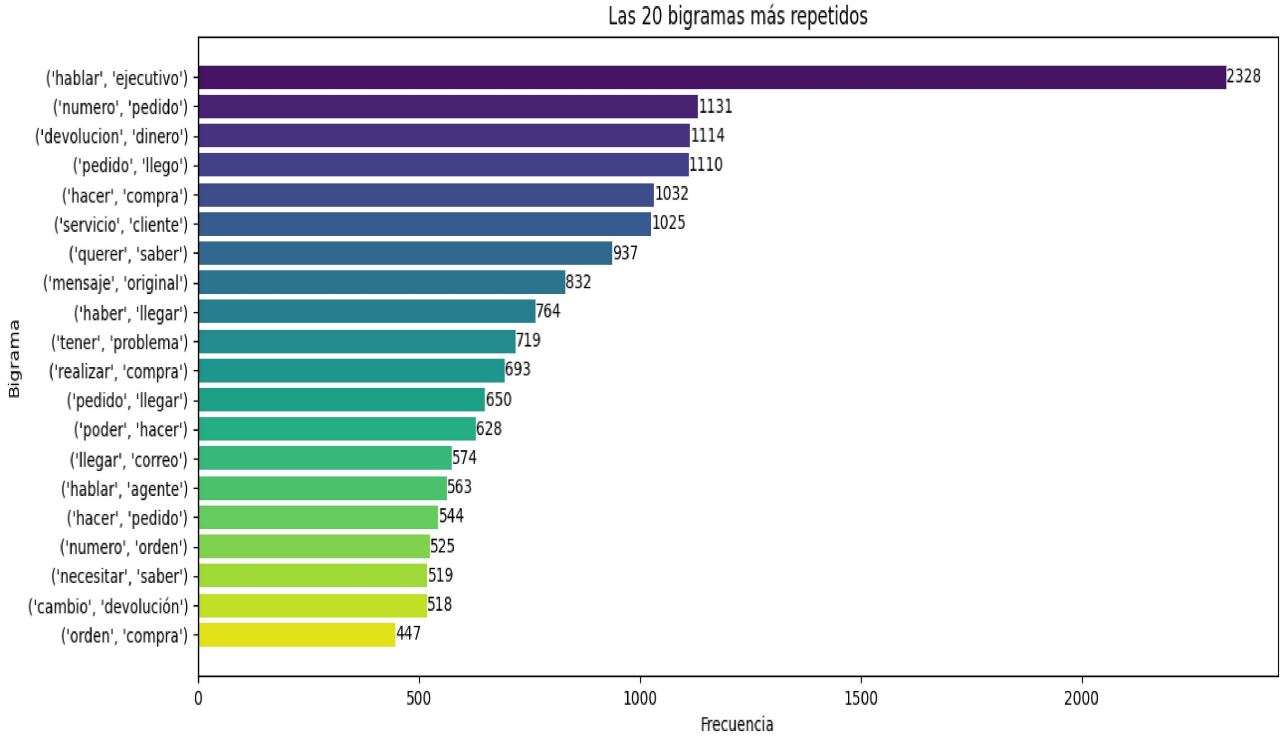

Figura 7.14: Bar Chart de los 20 bigramas más frecuentes en el dataframe de Muestreo 2

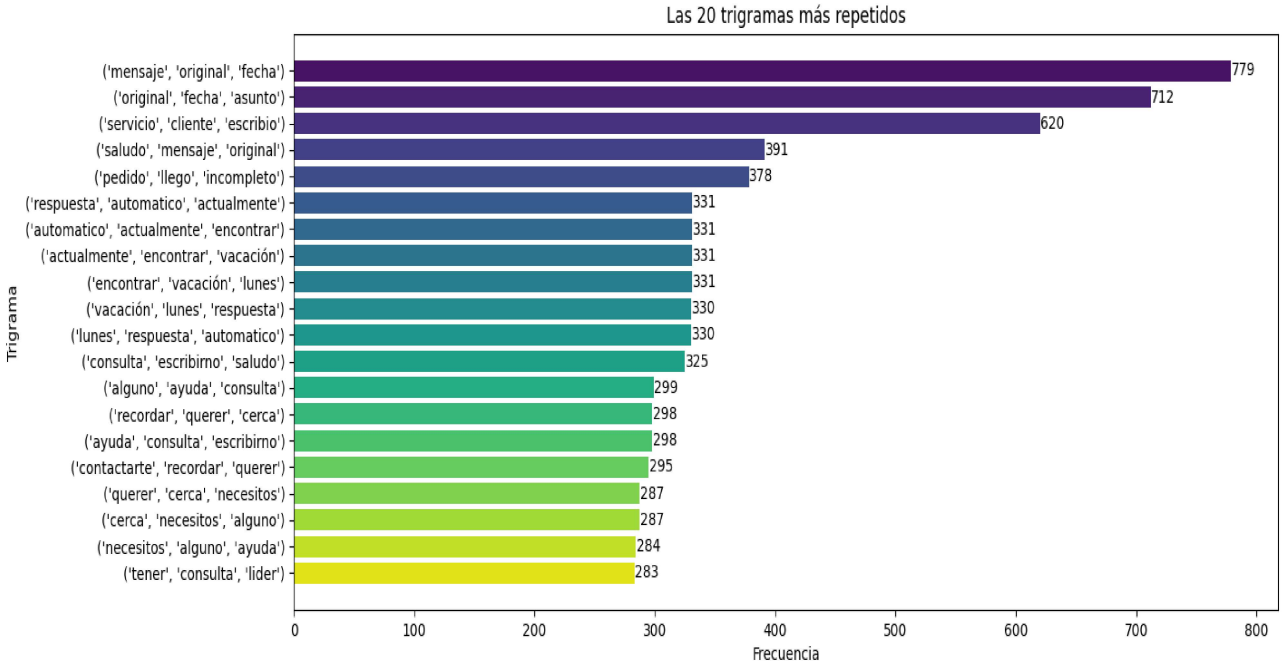

Figura 7.15: Bar Chart de los 20 trigramas más frecuentes en el dataframe de Muestreo 2

7.4.4.2. Gráficos Referentes a la Segunda Dimensión

7.4.4.2.1. Distribución General de Tópicos

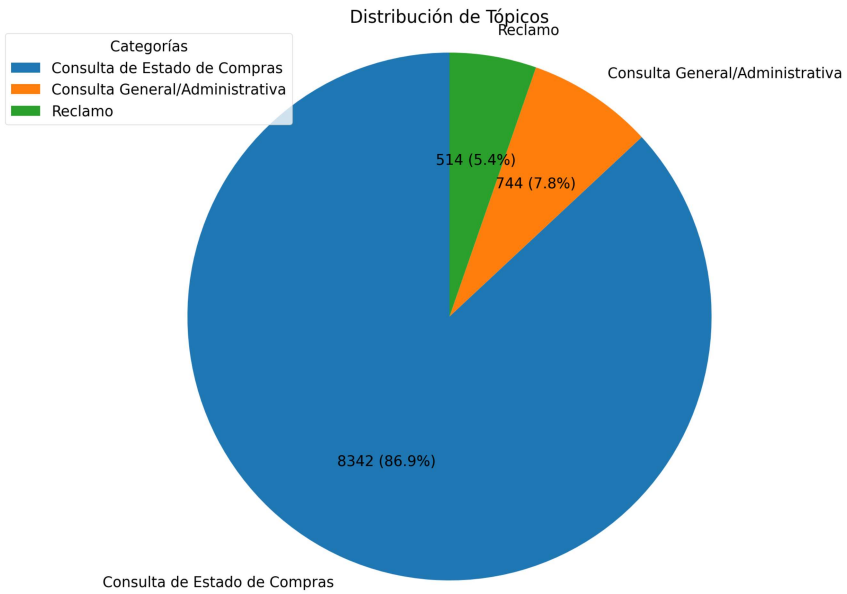

Figura 7.16: Pie chart de la distribución de la columna 'Tópico' en el dataframe Muestreo 2

En la figura 7.16 , se presenta un gráfico de torta donde muestra la distribución de los tópicos, en el cual se observa que el tópico más predominante es "Consulta de Estado de Compras", representando el 86.9 % del total. Esto indica que la mayoría de las interacciones en el canal están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es "Consulta General/Administrativa", con un 7.8 % del total, lo que sugiere que una parte significativa de las interacciones también está relacionada con consultas de carácter más general o administrativo. Por último, el tópico "Reclamo" representa el 5.4 % del total, lo que indica que hay menos incidencias de reclamos en comparación con las otras categorías de tópicos. Esto puede ser útil para comprender mejor las necesidades y preocupaciones de los usuarios.

7.4.4.2.2. Distribución de Tópicos por canal

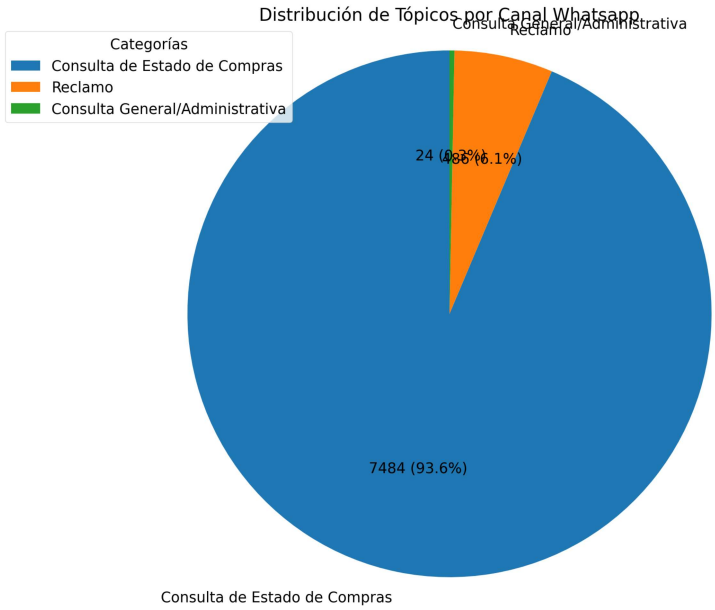

Figura 7.17: Pie Chart de la distribución de tópicos en canal de Whatsapp en Muestreo 2

El gráfico de torta presente en la figura 7.17, muestra la distribución de los tópicos específicamente para el canal de WhatsApp. Podemos observar que el tópico más predominante es "Consulta de Estado de Compras", representando el 93.6 % del total. Esto sugiere que la mayoría de las interacciones en el canal de WhatsApp están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es "Reclamo", con un 6.1 % del total, indicando que hay un número significativo de reclamos en este canal. Por último, el tópico "Consulta General/Administrativa" representa solo el 0.3 % del total, lo que sugiere que hay una proporción mínima de consultas de carácter más general o administrativo en comparación con las otras categorías de tópicos. Este análisis proporciona una visión clara de cómo se distribuyen los diferentes tipos de interacciones en el canal de WhatsApp, lo que puede ser útil para comprender las necesidades y preocupaciones de los usuarios específicamente en este canal.

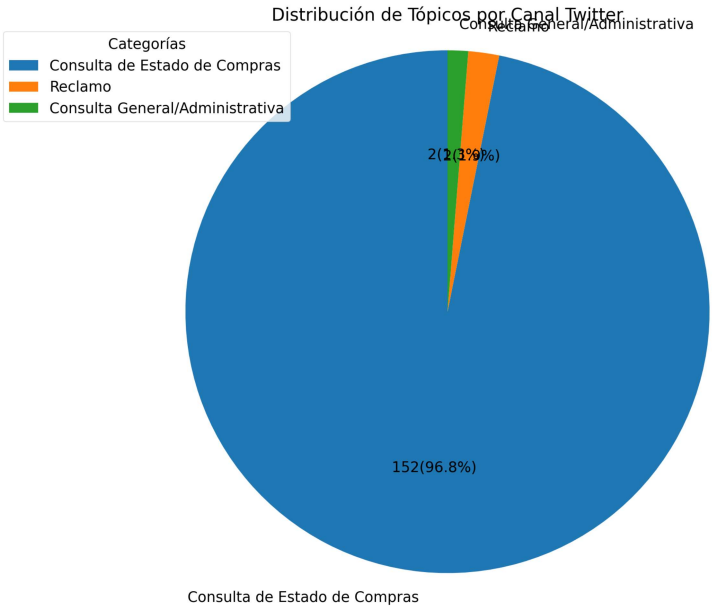

Figura 7.18: Pie Chart de la distribución de tópicos en canal de Twitter en Muestreo 2

El gráfico de torta presente en la figura 7.18 muestra la distribución de los tópicos específicamente para el canal de Twitter. Se observa que el tópico más predominante es "Consulta de Estado de Compras", representando el 96.8 % del total. Esto indica que la gran mayoría de las interacciones en el canal de Twitter están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es "Reclamo", con un 1.9 % del total, lo que sugiere que hay un número considerable de reclamos en este canal, aunque representan una proporción mucho menor en comparación con las consultas de estado de compras. Por último, el tópico "Consulta General/Administrativa" representa el 1.3 % del total, lo que indica que hay una proporción mínima de consultas de carácter más general o administrativo en comparación con las otras categorías de tópicos. Este análisis proporciona una visión clara de cómo se distribuyen los diferentes tipos de interacciones en el canal de Twitter, lo que puede ser útil para comprender las necesidades y preocupaciones de los usuarios específicamente en este canal.

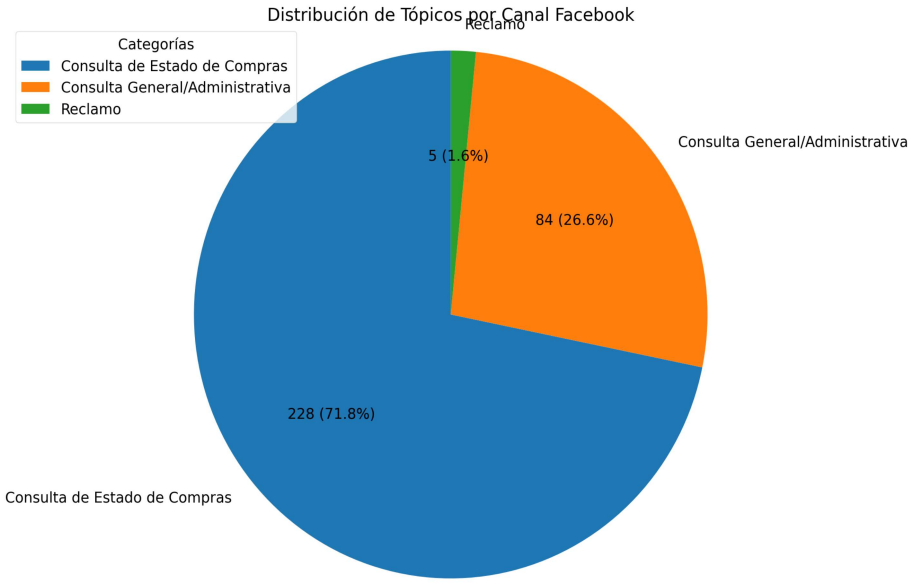

Figura 7.19: Pie Chart de la distribución de tópicos en canal de Facebook en Muestreo 2

El gráfico de torta presente en la figura 7.19 muestra la distribución de los tópicos por el canal de Facebook. Observamos que el tópico más predominante es "Consulta de Estado de Compras", representando el 71.8 % del total. Esto indica que la mayoría de las interacciones en el canal de Facebook están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es "Reclamo", con un 26.6 % del total, lo que sugiere que hay una proporción significativa de reclamos en este canal. Por último, el tópico "Consulta General/Administrativa" representa el 1.6 % del total, lo que indica que hay una proporción mínima de consultas de carácter más general o administrativo en comparación con las otras categorías de tópicos. Este análisis proporciona una visión clara de cómo se distribuyen los diferentes tipos de interacciones en el canal de Facebook, lo que puede ser útil para comprender las necesidades y preocupaciones de los usuarios específicamente en este canal.

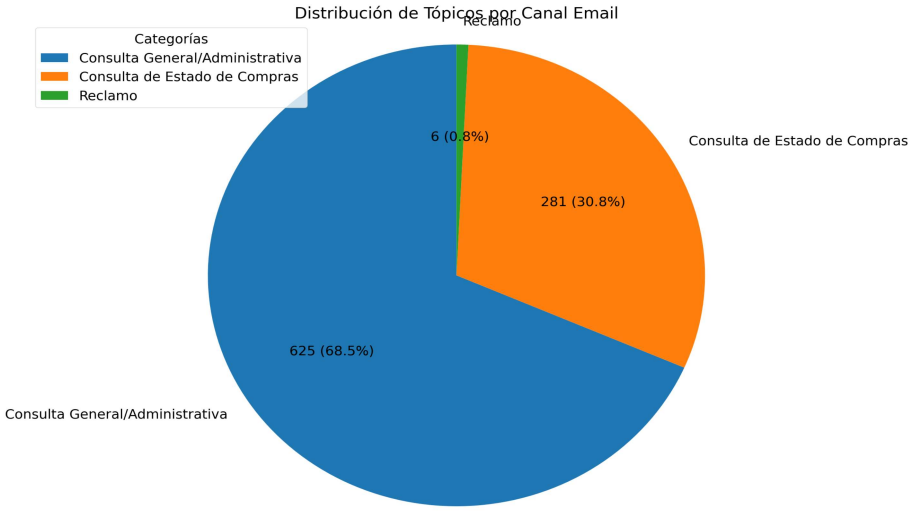

Figura 7.20: Pie Chart de la distribución de tópicos en canal de Email en Muestreo 2

El gráfico de torta presente en la figura 7.20 muestra la distribución de los tópicos por el canal de correo electrónico. Se observa que el tópico más predominante es “Consulta de Estado de Compras”, representando el 68.5 % del total. Esto indica que la mayoría de las interacciones por correo electrónico están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es “Reclamo”, con un 30.8 % del total, lo que sugiere que hay una proporción significativa de reclamos recibidos a través de este canal. Por último, el tópico “Consulta General/Administrativa” representa solo el 0.8 % del total, lo que indica que hay una proporción mínima de consultas de carácter más general o administrativo en comparación con las otras categorías de tópicos. Este análisis proporciona una visión clara de cómo se distribuyen los diferentes tipos de interacciones en el canal de correo electrónico, lo que puede ser útil para comprender las necesidades y preocupaciones de los usuarios específicamente en este canal.

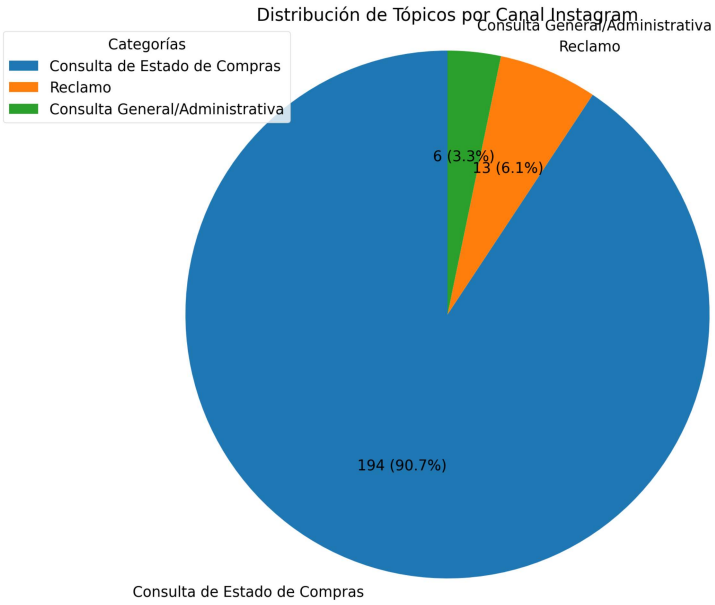

Figura 7.21: Pie Chart de la distribución de tópicos en canal de Instagram en Muestreo 2

El gráfico de torta presente en la figura 7.21 muestra la distribución de los tópicos por el canal de Instagram. Se observa que el tópico más predominante es "Consulta de Estado de Compras", representando el 90.7 % del total. Esto indica que la gran mayoría de las interacciones en el canal de Instagram están relacionadas con consultas sobre el estado de compras. El segundo tópico más común es "Reclamo", con un 6.1 % del total, lo que sugiere que hay un número significativo de reclamos recibidos a través de este canal. Por último, el tópico "Consulta General/Administrativa" representa el 3.3 % del total, lo que indica que hay una proporción más pequeña de consultas de carácter más general o administrativo en comparación con las otras categorías de tópicos. Este análisis proporciona una visión clara de cómo se distribuyen los diferentes tipos de interacciones en el canal de Instagram, lo que puede ser útil para comprender las necesidades y preocupaciones de los usuarios específicamente en este canal.

7.4.4.3. Gráficos Referentes a la Tercera Dimensión

7.4.4.3.1. Distribución General de Sentimientos

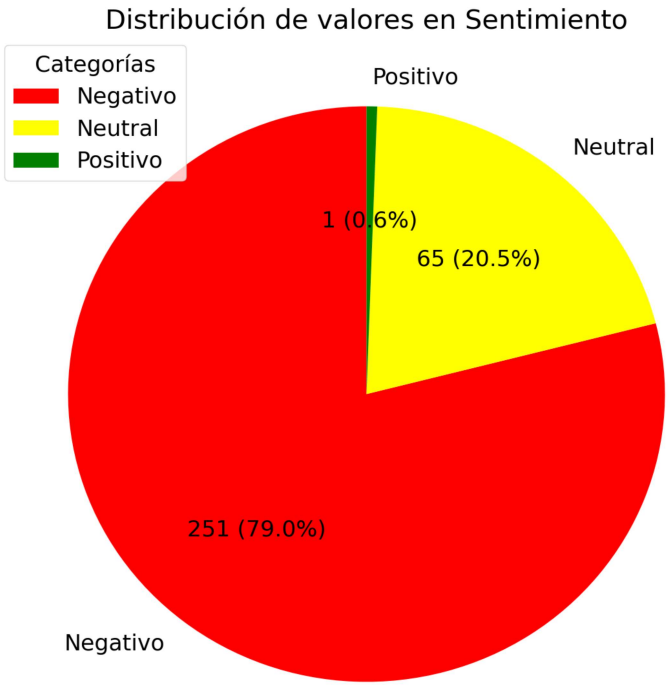

Figura 7.22: Pie chart de la distribución de la columna 'Sentimiento' en el dataframe Muestreo 2

El gráfico de torta presente en la figura 7.22 muestra la distribución del sentimiento en las interacciones analizadas. Se observa que la mayoría de las interacciones, aproximadamente el 79.0 %, denotan un sentimiento negativo. Esto indica que una proporción significativa de las interacciones contiene sentimientos negativos por parte de los usuarios. Por otro lado, el 20.5 % de las interacciones denotan un sentimiento neutral, lo que sugiere que una parte considerable de las interacciones no expresan claramente una inclinación positiva o negativa. Finalmente, solo el 0.6 % de las interacciones denotan un sentimiento positivo, lo que indica que hay una proporción muy pequeña de interacciones que expresan un sentimiento positivo. Este análisis proporciona información importante sobre la percepción general de los usuarios y puede ser útil para identificar áreas de mejora en la comunicación o la experiencia del cliente.

7.4.4.3.2. Distribución de Sentimientos por canal

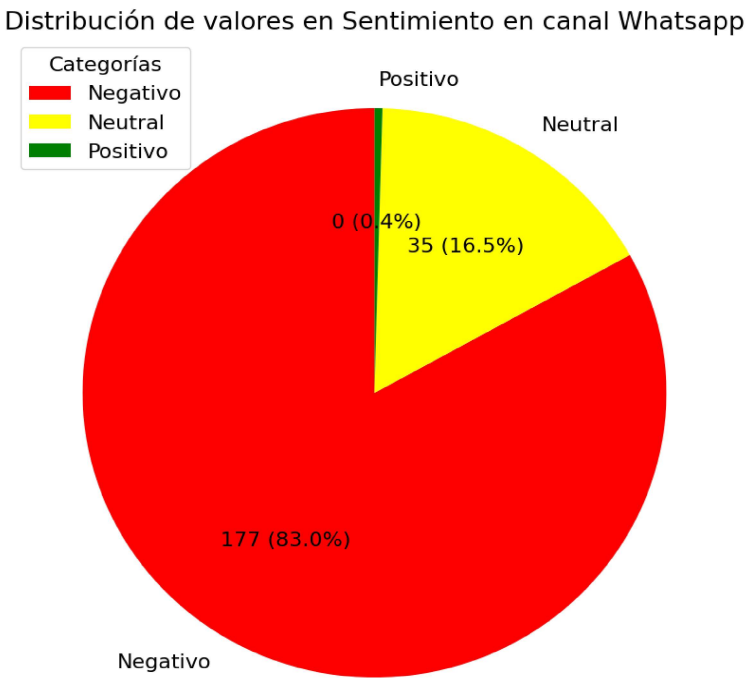

Figura 7.23: Pie Chart de la distribución de Sentimientos en canal de Whatsapp en Muestreo 2

El gráfico de torta presente en la figura 7.23 muestra la distribución del sentimiento en las interacciones del canal de WhatsApp. Se observa que la gran mayoría de las interacciones, aproximadamente el 83.0 %, denotan un sentimiento negativo. Esto sugiere que una proporción considerable de las interacciones en el canal de WhatsApp están asociadas con sentimientos negativos por parte de los usuarios. Por otro lado, el 16.5 % de las interacciones denotan un sentimiento neutral, lo que indica que una parte significativa de las interacciones no muestra claramente una inclinación positiva o negativa. Finalmente, solo el 0.4 % de las interacciones denotan un sentimiento positivo, lo que sugiere que hay una proporción muy pequeña de interacciones que expresan un sentimiento positivo en comparación con los sentimientos negativos y neutrales. Este análisis proporciona una visión importante sobre cómo se perciben las interacciones en el canal de WhatsApp y puede ayudar a identificar áreas de mejora en la comunicación o la experiencia del cliente en este canal específico.

7.4.4.3.3. Distribución de Tópicos por canal

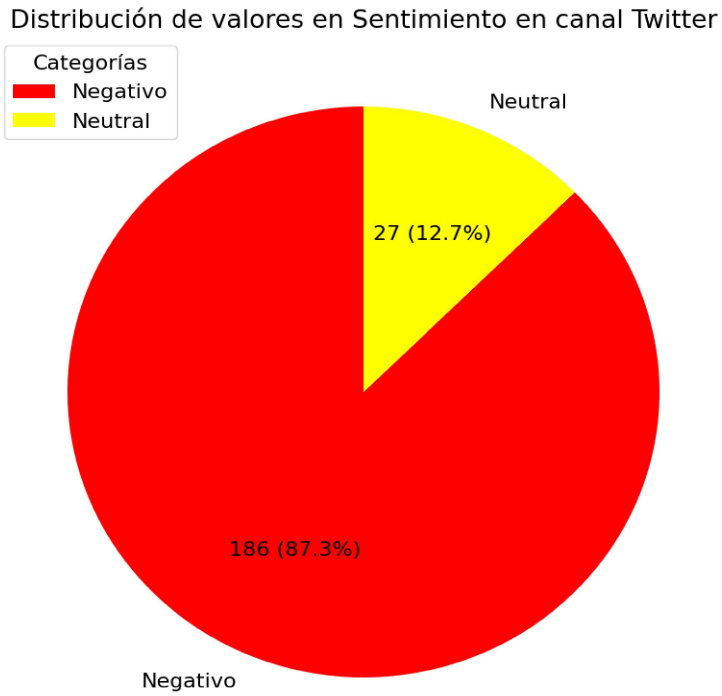

Figura 7.24: Pie Chart de la distribución de Sentimientos en canal de Twitter en Muestreo 2

El gráfico de torta presente en la figura 7.24 muestra la distribución del sentimiento en las interacciones del canal de Twitter. Se observa que la mayoría de las interacciones, aproximadamente el 87.3 %, denotan un sentimiento negativo. Esto sugiere que una proporción significativa de las interacciones en el canal de Twitter están asociadas con sentimientos negativos por parte de los usuarios. Por otro lado, el 12.7 % de las interacciones denotan un sentimiento neutral, lo que indica que una parte considerable de las interacciones no muestra claramente una inclinación positiva o negativa. Este análisis proporciona una visión importante sobre cómo se perciben las interacciones en el canal de Twitter y puede ayudar a identificar áreas de mejora en la comunicación o la experiencia del cliente en este canal específico.

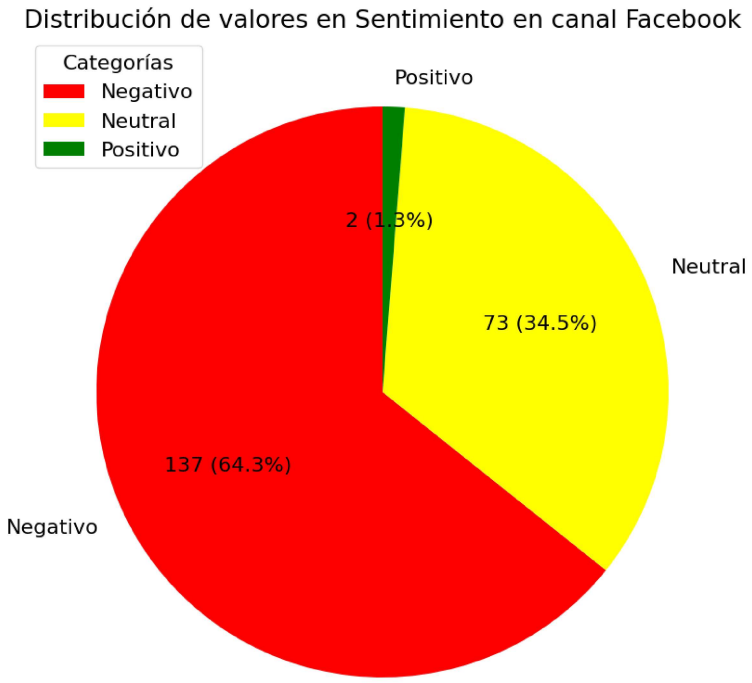

Figura 7.25: Pie Chart de la distribución de Sentimientos en canal de Facebook en Muestreo 2

El gráfico de torta presente en la figura 7.25 muestra la distribución del sentimiento en las interacciones del canal de Facebook. Observamos que el 64.3 % de las interacciones denotan un sentimiento negativo, lo que sugiere que una parte considerable de las interacciones en el canal de Facebook están asociadas con sentimientos negativos por parte de los usuarios. Por otro lado, el 34.5 % de las interacciones denotan un sentimiento neutral, indicando que una parte significativa de las interacciones no muestra claramente una inclinación positiva o negativa. Finalmente, el 1.3 % de las interacciones denotan un sentimiento positivo, lo que sugiere que hay una proporción muy pequeña de interacciones que expresan un sentimiento positivo en comparación con los sentimientos negativos y neutrales. Este análisis proporciona una visión importante sobre cómo se perciben las interacciones en el canal de Facebook y puede ayudar a identificar áreas de mejora en la comunicación o la experiencia del cliente en este canal específico.

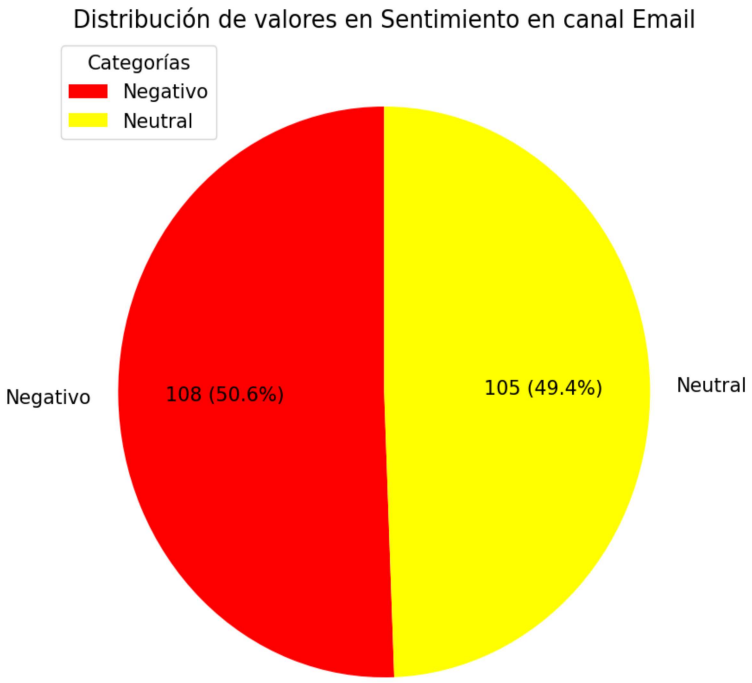

Figura 7.26: Pie Chart de la distribución de Sentimientos en canal de Email en Muestreo 2

El gráfico de torta presente en la figura 7.26 muestra la distribución del sentimiento en las interacciones del canal de correo electrónico. Observamos que aproximadamente el 50.6 % de las interacciones denotan un sentimiento negativo, lo que sugiere que una parte significativa de las interacciones en el canal de correo electrónico están asociadas con sentimientos negativos por parte de los usuarios. Por otro lado, el 49.4 % de las interacciones denotan un sentimiento neutral, lo que indica que una proporción considerable de las interacciones no muestra claramente una inclinación positiva o negativa. Este análisis proporciona una visión importante sobre cómo se perciben las interacciones en el canal de correo electrónico y puede ayudar a identificar áreas de mejora en la comunicación o la experiencia del cliente en este canal específico.

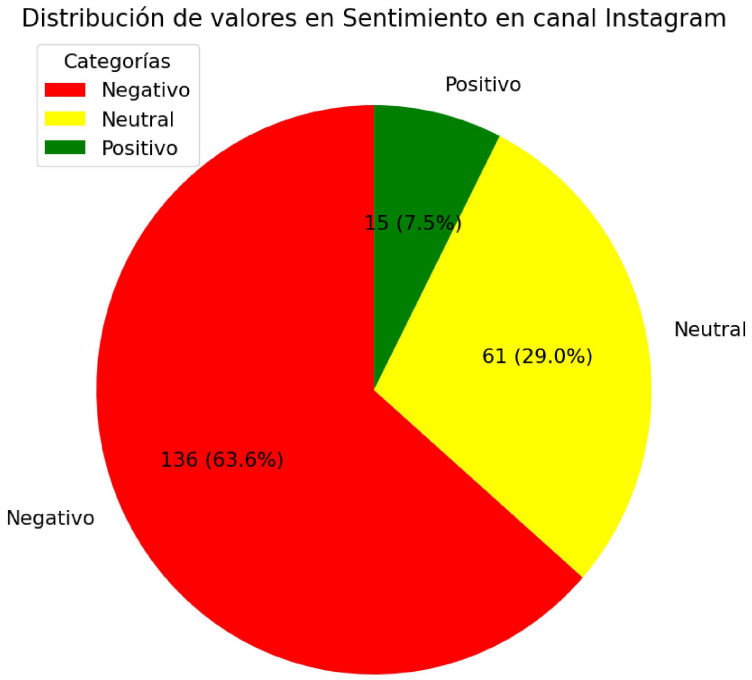

Figura 7.27: Pie Chart de la distribución de Sentimientos en canal de Instagram en Muestreo 2

El gráfico de torta presente en la figura 7.27 muestra la distribución del sentimiento en las interacciones del canal de Instagram. Observamos que aproximadamente el 63.6 % de las interacciones denotan un sentimiento negativo, lo que sugiere que una parte considerable de las interacciones en el canal de Instagram están asociadas con sentimientos negativos por parte de los usuarios. Por otro lado, el 29.0 % de las interacciones denotan un sentimiento neutral, indicando que una proporción significativa de las interacciones no muestra claramente una inclinación positiva o negativa. Finalmente, el 7.5 % de las interacciones denotan un sentimiento positivo, lo que sugiere que hay una proporción relativamente pequeña de interacciones que expresan un sentimiento positivo en comparación con los sentimientos negativos y neutrales. Este análisis proporciona una visión importante sobre cómo se perciben las interacciones en el canal de Instagram y puede ayudar a identificar áreas de mejora en la comunicación o la experiencia del cliente en este canal específico.

Capítulo 8

Conclusiones

Actualmente, el desarrollo de este trabajo de título representa un hito significativo dentro de la empresa, ya que introduce por primera vez un enfoque analítico del texto. Es importante destacar que la fase de preparación de datos es la más extensa, requiriendo aproximadamente 12 horas para procesar un conjunto de datos equivalente a seis meses de tickets. Este desafío destaca la importancia de desarrollar estrategias eficientes para la gestión y limpieza de datos en futuras implementaciones.

En relación con la fase de Modelamiento y Resultados, se lograron avances sustanciales. En la primera dimensión, se obtuvieron perspicacias valiosas a partir del análisis de conteos frecuentes de palabras, como se ilustra en las figuras 7.4, 7.5, y 7.6.

Para la segunda dimensión, se logró un resultado robusto al descubrir y caracterizar los tópicos ocultos en el Muestreo 1, así como asignarlos de manera efectiva. En términos de modelos de clasificación, se desarrollaron cuatro en total, dos basados en una red neuronal recurrente LSTM con clases desbalanceadas y otros dos en un modelo transformer llamado distilBERT, uno con clases desbalanceadas y otro con clases balanceadas. Los resultados más destacados se obtuvieron con el modelo LSTM balanceado, alcanzando un rendimiento del 95.16 %, precisión y recall del 95.00 %, y un f1-score del 95.00 %, superando eficazmente el desafío de las clases desbalanceadas.

En la tercera dimensión, se exploró el modelo GPT 3.5 Turbo para inferir el sentimiento en los tickets del Muestreo 2 mediante técnicas de prompt engineering. Comparando los resultados con un etiquetado manual, el modelo mostró un rendimiento del 95.00 % de precisión, 73.00 % de recall y un f1-score del 80.00 %. Estos resultados demuestran la capacidad del modelo para abordar la falta de etiquetas preexistentes de manera efectiva.

Finalmente, el trabajo culminó con la propuesta de diversas visualizaciones que no solo iluminan los hallazgos obtenidos, sino que también sirven como guía para futuras implementaciones de analítica en el helpdesk. De igual manera, se cumplió con la entrega de un instructivo para el correcto despliegue de los preprocesamientos y modelos desarrollados. Este enfoque analítico emergente tiene el potencial de transformar significativamente la comprensión y gestión de interacciones en el servicio al cliente digital, con lo que Adereso espera consolidar en un futuro cercano como uno de sus productos innovadores.

Capítulo 9

Trabajo Futuro

En el contexto del trabajo futuro, es crucial resaltar que el conocimiento adquirido sobre la herramienta de la API de OpenAI, especialmente en relación con la segunda dimensión del proyecto, abre una perspectiva valiosa para futuras iteraciones. La utilización de esta herramienta podría haber optimizado significativamente la resolución de problemas asociados a la falta de etiquetas preexistentes en la clasificación de tópicos. Por lo tanto, se plantea la importancia de considerar esta API en futuras investigaciones, permitiendo una mejora continua en la calidad y precisión de los modelos desarrollados en la segunda dimensión. Por otro lado, es importante recalcar que estos modelos y dimensiones en un futuro tendrán que apartarse al contenido semántico que podría ir cambiando paulatinamente, por lo que estar atento a el contenido de la data textual proveniente de sus actuales y posibles nuevos clientes del Retail es indispensable a la hora de procurar un buen desplieque de esta analítica.

En aras de expandir y enriquecer la analítica aplicada al helpdesk, se sugiere explorar nuevas dimensiones que complementen las ya abordadas. Una de estas dimensiones podría centrarse en el Reconocimiento de Entidades Nombradas (NER), enfocado específicamente en nombres propios y ubicaciones. Este enfoque proporcionaría una comprensión más profunda de la connotación que pueden tener enunciaciones específicas dentro de las conversaciones extraídas del helpdesk.

Además, se plantea la idea de incorporar un modelo subtipificador para la clasificación de reclamos. Este modelo adicional permitiría una mayor granularidad en la identificación de los tipos de reclamos que llegan al helpdesk. Al diferenciar y categorizar los reclamos de manera más detallada, se proporcionaría una herramienta valiosa para iniciar una gestión más eficiente y personalizada de estos casos.

Bibliografía

Adereso. Ia generativa para grandes empresas. (s/f). adere.so. recuperado de https://www.adere.so/. https://www.adere.so/,.

Anishnama. Understanding lstm: Architecture, pros and cons, and implementation. medium. abril 2023. https://medium.com/@anishnama20/understanding-lstm-architecture-pros-and-cons-and-implementation-3e0cca194094.

Ng A. Jordan M. Blai, D. Latent dirichlet allocation. stanford.edu. 2003. http://ai.stanford.edu/~ang/papers/nips01-lda.pdf,.

y Fernández E. Briceño, B. ¿qué son los word embeddings y para qué sirven? abierto al público; banco interamericano de desarrollo. julio 2021. https://blogs.iadb.org/conocimiento-abierto/es/que-son-los-word-embeddings/,.

Murati M. y Welinder P. Brockman, G. Openai api. 2020. https://openai.com/blog/openai-api,.

Deeplearning.Ai. Chatgpt prompt engineering for developers. mayo 2023. https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/,.

A. Grisales, C y Figueroa. Modelado de tópicos aplicado al análisis del papel del aprendizaje automático en revisiones sistemáticas. marzo 2022. https://doi.org/10.19053/20278306.v12.n2.2022.15271,.

Jin R. Liu C. Huang Y. Shi D. Supryadi Yu L. Liu Y. Li J. Xiong B. y Xiong D. Guo, Z. Evaluating large language models: A comprehensive survey. noviembre 2023. http://arxiv.org/abs/2310.19736,.

HuggingFace. Transformers — transformers 3.0.2 documentation. (2020). huggingface.co. 2020. https://huggingface.co/transformers/v3.0.2/index.html,.

IBM. ¿qué es el aprendizaje no supervisado?. ibm.com. recuperado de https://www.ibm.com/es-es/topics/unsupervised-learning. https://www.ibm.com/es-es/topics/unsupervised-learning,.

Keras. Keras.io. recuperado de https://keras.io/api/. https://keras.io/api/,.

y Kumar A. Nisha, V. M. Implementation on text classification using bag of words model. ssrn.com. 2020. https://deliverypdf.ssrn.com/delivery.php?ID=013084113099075064083099119104085007125015095067062090018069013096127069064083084110101106016015108127058124126031028105078027047047048048043097070094117123117107028032021062025124107115106107067067007126121117083064092021087120072122085086076020021117&EXT=pdf&INDEX=TRUE,.

Oracle. ¿qué es el aprendizaje profundo?. oracle.com. recuperado de https://www.oracle.com/cl/artificial-intelligence/machine-learning/what-is-deep-

learning/. https://www.oracle.com/cl/artificial-intelligence/machine-learning/what-is-deep-learning/,.

R. Schmidt. Schmidt, r. (2019, noviembre 23). recurrent neural networks (rnns): A gentle introduction and overview. noviembre 2019. http://arxiv.org/abs/1912.05911,.

Shazeer N. Brain G. Parmar N. Uszkoreit J. Jones L. Gomez A. N. y Kaiser L. Vaswani, A. Attention is all you need. 2017. http://arxiv.org/abs/1706.03762,.