Vigilada Mineducación

Análisis de la tendencia de la solución de una interacción con un Chatbot de atención al cliente, basado en análisis de sentimiento y otras variables

Analysis of the solution trend of an interaction with a customer service Chatbot, based on sentiment analysis and other variables

LUZ STELLA FLÓREZ SALAZAR
lflorezs@eafit.edu.co

TRABAJO DE GRADO

Asesor
Edwin Nelson Montoya Múnera
emontoya@eafit.edu.co

UNIVERSIDAD EAFIT
Escuela de Administración
Maestría en ciencias de los datos y la analítica
Medellín
2023

RESUMEN

Un chatbot es un programa creado con inteligencia artificial que, en el contexto de atención a usuarios, tiene la capacidad de establecer conversaciones con los clientes y son entrenados para resolver sus consultas, problemas y quejas.

La capacidad de un chatbot de identificar cuando un cliente no está resolviendo su requerimiento, representa un reto para las empresas que actualmente hacen uso de esta tecnología. Una de las estrategias para evitar el abandono de la conversación por esta causa, es el desborde o la transferencia de la conversación a un asesor humano. Por lo tanto, es indispensable detectar cuando es el momento de realizar este desborde.

En el presente proyecto se evalúan diferentes técnicas del Procesamiento del Lenguaje Natural (PLN), algoritmos de etiquetado basados en reglas, modelos clásicos de aprendizaje de máquina supervisado y una red neuronal sencilla para clasificación, aplicadas en interacciones entre un chatbot de servicio al cliente y un usuario, con el fin de encontrar un mecanismo de etiquetado automático de los datos y de construir un modelo que pueda ser empleado para tomar la decisión sobre si el cliente debe seguir interactuando con el chatbot o si debe ser transferido a una conversación con un asistente humano. El mecanismo de etiquetado también podría ser usado para clasificar datos históricos, para posteriormente entrenar un modelo.

Los diferentes modelos y técnicas se evalúan y se presentan los resultados de los que tienen el mejor desempeño al detectar las conversaciones que deben realizar el desborde a un asesor humano.

Palabras clave: Análisis de sentimientos, chatbot, escalamiento a asesor humano, inteligencia artificial, procesamiento de lenguaje natural.

ABSTRACT

A chatbot is a program created with artificial intelligence. In the context of customer service, can establish conversations with customers and they are trained to resolve their queries, problems and complaints.

A chatbot's skill to identify when a customer is not meeting their request represents a challenge for companies that currently use this technology. One of the strategies to avoid quitting the conversation for this reason, is to escalate or transfer the conversation to a human agent. Therefore, it is essential to detect when it is time to carry out this escalation.

This project evaluates different Natural Language Processing (NLP) techniques, rule-based labeling algorithms, classical supervised machine learning models and a simple neural network for classification, applied to interactions between a customer service chatbot and a user, in order to find a mechanism for automatic labeling of the data and to build a model that can be used to make the decision on whether the customer should continue interacting with the chatbot or if he should be transferred to a conversation with a human agent. The labeling mechanism could also be used to classify historical data, to later train a model.

Different models and techniques are evaluated and those with the best performance in detecting the conversations that should escalate to a human agent are presented.

Keywords: Sentiment analysis, chatbot, escalation to human agent, artificial intelligence, natural language processing.

ÍNDICE GENERAL

ÍNDICE DE TABLAS Y FIGURAS

1 INTRODUCCIÓN

1.1 Planteamiento del problema

Las empresas se preocupan cada vez más por prestar un buen servicio a sus clientes. Según Forbes, el 80% de las empresas en el mundo adelantaron su transformación digital por Covid-19 (Forbes, 2020). En 2020, Colombia dio pasos importantes para ser un "País Digital", adelantándose entre 5 y 6 años a las expectativas hechas antes de la emergencia sanitaria (ANDI, 2021).

Por otro lado, la mayoría de los proveedores de servicio al cliente han visto un aumento en la cantidad de solicitudes de soporte en línea y se espera que esta tendencia se mantenga incluso después de la pandemia (Prasad y Akana, 2021). Es por esto que las empresas buscan suplir la necesidad de que los clientes se puedan auto atender desde su hogar. En consecuencia, los canales digitales para la atención al cliente han ido evolucionando, contando actualmente con chatbots entrenados para dar respuestas a transacciones como: resolver las consultas de los clientes, quejas y reclamos, entre otras.

Como tecnología naciente, pero con gran aceptación comercial, se han implementado y desplegado masivamente los chatbots como un medio para la interacción directa del usuario, con fines de servicio al cliente (Brandtzaeg y Følstad, 2017). Sin embargo, éstos tienen muchos aspectos todavía en desarrollo. La mayoría de los chatbots disponibles no satisfacen las necesidades de los usuarios debido a respuestas sin sentido o usabilidad insuficiente (Brandtzaeg y Følstad, 2017).

Además, la capacidad de un chatbot de identificar cuando un cliente no está resolviendo su requerimiento o se está frustrando con la conversación, representa un reto para las empresas que actualmente hacen uso de esta tecnología. La falta de esta capacidad no permite evitar el abandono temprano de la conversación o tomar decisiones para ayudar al cliente a resolver su solicitud.

Para este proyecto, no se logró obtener datos de conversaciones con una etiqueta robusta que pudiera ser usada en un modelo de desborde. Por lo tanto, se propone un algoritmo de etiquetado automático de los datos y un modelo que apoya la toma de decisión basado en análisis de sentimientos, la similitud de frases (repitencia) y la cantidad de interacciones del cliente, provenientes de la interacción entre un usuario y un chatbot de servicio al cliente, que permite estimar el momento adecuado para desbordar la conversación a un asesor humano.

Así, cuando se detecte que el sentimiento de un cliente está tendiendo a ser negativo y que las otras variables asociadas indiquen que el usuario no está resolviendo su necesidad con el chatbot, éste podrá ser transferido a un asesor humano, quien continuará con la atención. Una de las variables que indican que el chatbot no está resolviendo la necesidad del usuario es, por ejemplo, una pregunta reiterada, las cuales son analizadas como frases similares contiguas.

1.2 Justificación

Cada vez, se está haciendo más frecuente la utilización de los chatbots para la atención al cliente. En 2018, más de 300.000 chatbots estaban activos solo en Facebook Messenger (Li et al., 2020). Con esta tecnología relativamente nueva, se vuelve primordial conocer cómo los clientes experimentan el chatbot, dado que es posible que sus expectativas no se cumplan por completo (Rapp et al., 2021).

Estas tecnologías aún no están listas para abordar las complejidades de las interacciones conversacionales, aspectos como la naturalidad de la conversación, la inteligencia del agente, el confort del usuario, fallos, entre otros, pueden generar experiencias negativas para los usuarios (Ashktorab et al., 2019).

Además, se ha demostrado que las emociones y pensamientos del usuario tienen un impacto en el contenido escrito (Prasad y Akana, 2021). Por esto, sin la capacidad de relacionarse emocionalmente con los clientes, un chatbot de servicio al cliente corre el riesgo de fracasar (Brandtzaeg y Følstad, 2017). Aunque algunos chatbots han sido entrenados para ser precisos en sus respuestas, la mayoría carece de la capacidad de monitorear la emoción humana (Prasad y Akana, 2021).

Por otro lado, el mal servicio al cliente no solo arruina las relaciones actuales, también pone en peligro las perspectivas futuras y puede perjudicar la línea financiera de la empresa (Prasad y Akana, 2021). Para que los chatbots tengan éxito, deben ayudar a los usuarios a resolver una tarea o lograr un objetivo concreto de manera eficaz y eficiente. Además, deben obtener ayuda o acceder a información sobre la marcha (Brandtzaeg y Følstad, 2017).

El chatbot con el que se trabajó en este proyecto no cuenta con la inteligencia suficiente para identificar situaciones de insatisfacción, sentimientos negativos o situaciones complejas que requieren intervención humana o una estrategia diferencial de atención.

Finalmente, se quiere contribuir a la comunidad científica creando nuevos modelos para detectar esta tendencia.

1.3 Objetivos

1.3.1 Objetivo general

Proponer modelos de decisión y criterios de selección sobre la continuidad de la atención al cliente con un chatbot o realizar la conmutación a humano, basado en el análisis de la tendencia de sentimientos en la interacción con un chatbot y otras variables complementarias provenientes de la conversación, ofreciendo una experiencia del cliente mejorada.

1.3.2 Objetivos específicos

- Documentar el marco teórico y estado de arte de los mecanismos actuales de afrontamiento ante la no solución de requerimientos de los usuarios o frustración en la conversación con chatbots.

- Realizar análisis exploratorio de los datos para comprender los patrones de las conversaciones y encontrar correlaciones entre las variables que se quieren utilizar en la propuesta de etiquetado y en la definición de reglas de conmutación.
- Construir un mecanismo de etiquetado automático de las conversaciones entre el chatbot y el usuario.
- Identificar las variables requeridas para implementar un modelo de análisis de tendencia de solución y/o frustración en la interacción de un cliente con un chatbot para el desborde a un asesor humano.
- Seleccionar una técnica adecuada para la detección de sentimientos en interacciones entre un humano y un chatbot.
- Desarrollar un modelo que permita identificar la tendencia de no solución en la interacción con un chatbot de servicio al cliente, basado en el análisis de sentimientos y otras variables del contexto identificadas.
- Realizar la validación del modelo de conmutación a humano del chatbot, que permita evaluar la efectividad del modelo propuesto.

2 MARCO TEÓRICO Y ESTADO DEL ARTE

2.1 Chatbots

Un chatbot es un agente conversacional que emplea principalmente entradas basadas en texto en lenguaje natural o voz (Liao et al., 2018), es un programa de inteligencia artificial y un modelo de interacción humano-computadora (Adamopoulou y Moussiades, 2020) que en el contexto de atención a usuarios, tiene la capacidad de establecer conversaciones con los clientes y son entrenados para dar resolver consultas de los clientes, sus problemas y quejas.

Un chatbot utiliza el PLN para comunicarse en lenguaje natural con humanos u otros chatbots (Khanna et al., 2015). Estos programas basados en texto ya se han aplicado en muchos servicios digitales que la gente usa ahora en su vida diaria (Rapp et al., 2021). Para las empresas, los chatbots suponen un ahorro de tiempo y costes, ya que se pueden automatizar muchos procesos y asignar empleados para tareas más complejas (Akhtar et al., 2019).

2.2 Análisis de sentimiento en texto

El análisis de sentimientos en texto, es el estudio computacional de los sentimientos, emociones y actitudes de las personas (Ligthart et al., 2021) que se pueden extraer de las opiniones positivas o negativas incluidas en las comunicaciones textuales, como reseñas de películas o productos (Wyeld et al., 2021). Es un modelo de clasificación de texto que implica PLN, aprendizaje automático, minería de datos y otros campos de investigación (Xu et al., 2019).

Antes de acuñar el término, el campo se estudiaba bajo nombres como subjetividad, punto de vista y minería de opinión (Katz et al., 2015). Una de las tareas básicas del análisis de sentimiento es la clasificación de texto, que clasifica diferentes sentimientos en categorías (Poria et al., 2017). El procesamiento de textos se lleva a cabo en varios niveles, lo que implica un análisis de nivel léxico, sintáctico, semántico y pragmático (Prasad y Akana, 2021).

Por otro lado, determinar la polaridad del sentimiento es una subtarea de la clasificación de sentimientos, que pretende identificar la polaridad en cada sentencia o documento.

Tradicionalmente, la polaridad se clasifica en: positivo, negativo o neutral (Ligthart et al., 2021). Esta subtarea permite analizar y entender una conversación, una sentencia o un documento.

Al analizar las conversaciones transcritas de un chatbot, por ejemplo, la empresa podría identificar y acercarse a los clientes insatisfechos, evaluar sus actitudes hacia los servicios y productos e identificar problemas en sus primeras etapas (Katz et al., 2015).

Las técnicas de análisis de sentimientos presentan aún varios retos por resolver:

- Las conversaciones pueden tener lenguaje implícito, que se refiere al humor, el sarcasmo y la ironía. Hay vaguedad y ambigüedad en esta forma de hablar, que a veces es difícil de detectar incluso para los humanos. Un significado implícito de una oración puede cambiar completamente la polaridad de una oración (Ligthart et al., 2021).
- Probar el modelo es muy subjetivo y no existe un estándar o punto de referencia absoluto para evaluar el rendimiento del algoritmo. Algunas de las formas más populares y apropiadas de los algoritmos de prueba de procesamiento de lenguaje natural son la satisfacción del usuario y el análisis de comentarios (Vijayaraghavan et al., 2020).

Para abordar el problema de análisis de sentimientos, existen tres enfoques principales: basado en léxico, aprendizaje automático e híbrido. El enfoque basado en léxico, responde a los léxicos emocionales para detectar las emociones de los clientes, sus principales inconvenientes son depender del contexto y los idiomas (Duong y Nguyen-Thi, 2021). El enfoque de aprendizaje automático aplica los algoritmos de aprendizaje de máquina o estadístico y utiliza características lingüísticas (Medhat et al., 2014). El enfoque híbrido es una combinación de los dos primeros. El enfoque basado en aprendizaje automático es el más utilizado para análisis de sentimientos.

Los algoritmos de aprendizaje automático para las tareas de análisis de sentimientos se pueden dividir en tres categorías: aprendizaje no supervisado, aprendizaje semi-supervisado y aprendizaje supervisado (Ligthart et al., 2021). El aprendizaje supervisado entrena un modelo con datos previamente etiquetados con las categorías o clases. El modelo entrenado puede posteriormente hacer predicciones para una salida que considere nuevos datos de entrada sin etiquetar. En la mayoría de los casos, el aprendizaje supervisado supera a los enfoques de aprendizaje no supervisados y semi-supervisados (Ligthart et al., 2021) pero esta tarea es tediosa, costosa y requiere mucho tiempo (Duong y Nguyen-Thi, 2021).

Por otro lado, también hay algoritmos de aprendizaje profundo que se utilizan en análisis de sentimientos, lo cual es una subrama del aprendizaje automático que utiliza redes neuronales profundas (Ligthart et al., 2021). Los algoritmos más utilizados son: Redes neuronales profundas (DNN), Redes neuronales convolucionales (CNN) y Redes neuronales recurrentes (RNN) (Ligthart et al., 2021).

A continuación, se describen algunas librerías para el análisis de sentimientos:

- TextBlob: Es una biblioteca de Python, que proporciona capacidades de análisis de sentimientos (Prasad y Akana, 2021). También provee funciones para el preprocesamiento del texto, como por ejemplo convertir toda la cadena de entrada en minúsculas.
- VADER: (*Valence Aware Dictionary for sEntiment Reasoning*). Es un modelo simple basado en reglas para el análisis de sentimiento general. Utiliza una combinación de métodos cualitativos y cuantitativos para producir y luego validar empíricamente, un léxico de sentimientos estándar de oro que está especialmente en sintonía con contextos similares a microblogs (Hutto y Gilbert, 2014). VADER se adapta bien a los sentimientos expresados en las redes sociales y por esto es ampliamente utilizado en este contexto.

Usa un léxico de sentimiento el cual es una lista de palabras que generalmente se etiquetan con *scores* de acuerdo con su orientación semántica como positiva o negativa.

- AFINN: Es una lista de términos en inglés clasificados manualmente por valencia. En él, se asigna a cada palabra un valor entre -5 y 5, siendo -5 el máximo de negatividad y +5 el máximo de positividad.

2.3 Análisis de tendencia de los sentimientos

El reto de este trabajo es, además de realizar la detección de sentimientos, realizar un análisis de tendencia del sentimiento durante la conversación entre un usuario y un chatbot de servicio al cliente que, junto con otras variables complementarias provenientes de la conversación, permitan identificar de forma automática, cuándo una conversación debe transferirse o desbordar a un asesor humano. Para esto, se analizaron conversaciones entre un cliente y un chatbot de servicio al cliente. El análisis de una conversación, es un proceso inductivo para analizar cómo se organizan las conversaciones de los usuarios en secuencias de acciones y prácticas sistemáticas (Li et al., 2020).

Si bien los chatbots se están convirtiendo en alternativas populares para satisfacer las necesidades de los usuarios, según Li et al. (2020), pocos estudios han investigado cómo los usuarios afrontan el "No Progreso" (NP) conversacional en su vida diaria. En el estudio realizado por la *National Chiao Tung University*, se encontraron 12 tipos de NP conversacionales; cinco tipos de contenido inesperado y difíciles de reconocer para el chatbot y 10 tipos de estrategias de afrontamiento.

Además, se identificaron relaciones específicas entre los tipos y estrategias de NP, así como señales de que los usuarios estaban a punto de abandonar el chatbot, incluyendo: 1) Tres incidencias consecutivas de NP, 2) Uso repetitivo de los mismos tipos de estrategias de afrontamiento, y 3) Reformulación de mensajes como estrategia final (Li et al., 2020). El análisis de la tendencia del sentimiento además de la identificación de estas estrategias de afrontamiento, podrían potencialmente disminuir la ocurrencia de abandono del chat y la detección de respuestas que pueden interrumpir el diálogo podría mejorar la experiencia del usuario, aumentar la confianza del consumidor en los chatbots y continuar la conversación de manera coherente (Almansor et al., 2021).

De acuerdo con Li et al. (2020), se desconoce la frecuencia de estos obstáculos, cómo los manejan los usuarios y cuál de ellos es más probable que provoque que los usuarios interrumpan la comunicación con un chatbot. Sin embargo, es necesario anticipar NP para disminuir el riesgo de que el usuario abandone la conversación. Li et al. (2020), también mencionan que casi ninguna investigación ha examinado la relación entre las interrupciones de las conversaciones de los chatbots y las estrategias que adoptan los usuarios para hacer frente a tales problemas.

Por otro lado, para el análisis de la tendencia de sentimiento, se requiere el modelado de contexto de los enunciados individuales. El contexto se puede atribuir a enunciados precedentes y se basa en la secuencia temporal de enunciados (Poria et al., 2019). Aunque se ha investigado la extracción de datos textuales y de audio de los datos de centros de llamadas, el análisis secuencial de las transcripciones de llamadas no se ha explorado ampliamente, según Lam et al. (2019). Debido a que la mayoría de los estudios se centran en la metodología en sí, solo unos pocos estudios han analizado los sentimientos en el contexto de los chatbots (Loria, 2021).

En una de las investigaciones relacionadas a este tema, Kuramoto et al (2018), realizaron un estudio con agentes conversacionales para suprimir la ira del cliente en conversaciones de

atención al cliente basadas en texto. El agente tenía dos métodos para reconocer el estado de ira del participante. Uno es el uso de un botón que se podía utilizar explícitamente y el otro es una estimación basada en corpus de la conversación entre el cliente y el agente (Kuramoto et al., 2018).

En el caso de este trabajo, no se quiere tener una opción en la que el usuario indique explícitamente que desea realizar la conmutación a humano, pero si se quiere estimar el momento en el que debe realizarse esa conmutación de forma automática, basada en corpus de la conversación entre el cliente y el chatbot.

En otras investigaciones, se ha encontrado que uno de los retos en el análisis de sentimientos en este tipo de conversaciones, es la dificultad para identificar mensajes de chat informales relativamente cortos, que con frecuencia contienen errores ortográficos y la representación de sentimientos sin un significado genuino (Prasad y Akana, 2021).

También, las entradas de los usuarios pueden ser de diferente tamaño y pueden no estar siempre compuestas por oraciones significativas. El contexto puede ser difícil de determinar y las oraciones que están mal construidas pueden ser perjudiciales para el aprendizaje si se usa un enfoque estadístico (Prasad y Akana, 2021).

2.4 Análisis de similitud entre interacciones

Para analizar la repitencia, es decir, qué tan similares son las interacciones contiguas del usuario, se usa la similitud del coseno o el kernel del coseno, el cual calcula la similitud como el producto escalar normalizado de X e Y, de la siguiente manera: K(X, Y) = <X, Y> / (||X||*||Y||). En datos normalizados L2, esta función es equivalente al kernel lineal (Pedregosa et al., 2011).

2.5 Clasificación con redes neuronales

Las redes neuronales artificiales (RNA), son una familia de algoritmos de aprendizaje automático (Mathur y Lopez, 2019) que están compuestos por elementos inspirados en la estructura del sistema nervioso de los seres humanos y que se denominan neuronas (Avila et al., 2020). Cada una de las neuronas, tiene unos elementos de entrada, generan una respuesta o salida y están organizadas en una serie de niveles que se denominan capas. El conjunto de estas capas forma una red neuronal artificial (Avila et al., 2020).

En las RNA existen 2 capas con conexiones con el exterior. Una capa de entrada y una capa de salida que devuelve al exterior la respuesta de la red a una entrada concreta. Existen, además, una serie de capas intermedias que se denominan capas ocultas (Avila et al., 2020).

La propiedad más importante de las RNA es la capacidad de aprender a partir de un conjunto de patrones de entrenamiento. El proceso de aprendizaje o entrenamiento de la red puede ser supervisado o no supervisado. El aprendizaje supervisado consiste en entrenar la red a partir de un conjunto de datos. El objetivo del algoritmo de aprendizaje es tratar de encontrar un modelo que genera la salida, de manera tal que la salida generada por la red sea lo más cercanamente posible a la verdadera salida dada una cierta entrada. Con el suficiente entrenamiento, la máquina será capaz de determinar la categoría de salida del nuevo dato de entrada. (Avila et al., 2020).

Las redes neuronales retroalimentadas (*FNN por sus siglas en inglés)* corresponden a la clase de RNA más estudiada por el ámbito científico y la más utilizada en los diversos campos de aplicación. Las redes neuronales *FNN* brindan una alternativa simple y eficiente para el reconocimiento de múltiples patrones en datos estáticos (clasificación binaria y multiclase). Las FNN con una sola capa oculta, son eficientes para resolver problemas de clasificación y regresión, lo que está respaldado por su capacidad de aproximación universal (Hora y Embiruçu, 2021).

2.6 Estado del arte

En su investigación, Prasad y Akana (2021), proponen una función de aumento (Figura 1) la cual busca en los últimos tres valores del índice de polaridad del sentimiento (cada uno de cada sentencia del usuario) y si el primero de su valor es mayor que un umbral (en este caso 0.40) y si es mayor o igual a los siguientes dos valores, es decir, que en las últimas tres conversaciones el cliente ha permanecido o aumentó el nivel de frustración y el sentimiento negativo, entonces es hora de escalar la conversación a un agente real (Prasad y Akana, 2021).

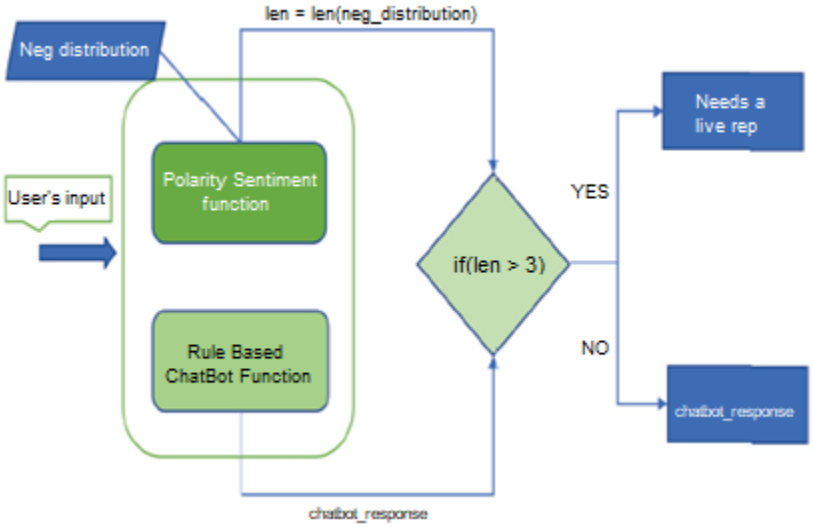

Figura 1. Función de aumento

Nota: Tomado de Augmentation Function (Prasad y Akana, 2021)

En otro estudio, Almansor et al., proponen un enfoque para etiquetar los conjuntos de datos en función del sentimiento, teniendo en cuenta el contexto de la conversión para predecir la ruptura del dialogo o abandono de la conversación. Los autores pretenden detectar el efecto del cambio de sentimiento de cada hablante en una conversación y luego manejan el problema mediante el uso de un mecanismo de traspaso, que transfiere al usuario a un asesor humano (Almansor et al., 2021).

El objetivo de su investigación es detectar de manera inteligente sobre la marcha respuestas deficientes o inapropiadas del chatbot tan pronto como ocurran antes de que puedan poner en peligro toda la conversación entre el usuario final y el chatbot. El mecanismo de traspaso se

activa cuando la calidad del servicio del chatbot es inapropiado, por lo que el chatbot transferirá al usuario a un agente en vivo para completar la conversación y ayudarlo en lo que necesita. (Almansor et al., 2021).

Aunque estos modelos se construyeron para detectar el punto de ruptura, tienen numerosas limitaciones y desafíos; por ejemplo, el proceso de etiquetado de datos aún no está automatizado, se basa en expertos humanos que hacen que el proceso sea subjetivo y sesgado. (Almansor et al., 2021).

3 DATOS

3.1 Plan de Gestión de Datos

Para el desarrollo de este proyecto, se consolidó un conjunto de datos de conversaciones entre los clientes y el chatbot de servicio al cliente. Las conversaciones tienen fecha de creación desde enero de 2021 hasta junio de 2022.

Los datos son propiedad exclusiva de la empresa para la cual se realiza este proyecto. Si bien no se tienen datos sensibles, éstos son privados y confidenciales, de uso académico exclusivamente para este proyecto. Solo el estudiante y el profesor tienen acceso a ellos.

Las conversaciones son anónimas; no se tiene la identificación del usuario dentro de los atributos. Sin embargo, se tiene el riesgo de poder identificar a una persona en particular. Con el fin de proteger la privacidad, estos datos no deben ser publicados en ningún medio por fuera de este proyecto. Tampoco podrán ser utilizados para uso comercial, no se podrán reutilizar ni redistribuir para uso en otro proyecto. Éstos se almacenan bajo custodia del estudiante.

Los datos producto derivados de este proyecto son públicos y están documentados en este informe junto con los resultados obtenidos.

3.2 Adquisición de datos

La fuente de datos son archivos de conversaciones entre el usuario y el chatbot de servicio al cliente, en formato JSON, extraídos de una base de datos *Cosmos DB* propiedad de la empresa, en la cual se almacenan las conversaciones. Los archivos JSON fueron convertidos a CSV y cargados en dataframes de pandas en cuadernos de python.

Las partes de la conversación se extrajeron desde el tag 'UtteranceList' y de éste, se tomaron los atributos de interés:

- 'IdConversacion'
- 'Dialogo'
- 'Texto'
- ‘Fecha’

El atributo fecha se descartó ya que no todas las observaciones la tienen, como se puede observar en la Figura 2.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1495683 entries, 0 to 1495682
Data columns (total 5 columns):
 #   Column          Non-Null Count    Dtype
---  ------          --------------    -----
 0   Archivo         1495683 non-null  int64
 1   IdConversacion  1495683 non-null  int64
 2   Dialogo         1495683 non-null  object
 3   Texto           1467913 non-null  object
 4   Fecha           289634 non-null   datetime64[ns]
dtypes: datetime64[ns](1), int64(2), object(2)
memory usage: 57.1+ MB
```

Figura 2. Atributos extraídos de las conversaciones

Este conjunto de datos contiene 392699 conversaciones entre enero y septiembre de 2021, con 1495683 registros o interacciones del usuario.

En 2022, el chatbot del canal WhatsApp cuenta con una funcionalidad de desborde manual, con la que el usuario puede ingresar la palabra “asesor” para ser transferido a un asesor humano. Estas conversaciones, junto con otras que no realizaron el desborde manual, adicionan al conjunto de datos 29584 conversaciones.

3.3 Descripción y análisis preliminar de los datos

Para la ejecución del proyecto, se utilizaron 4 archivos de datos:

- 100 encuestas de usuarios del chatbot con calificaciones de 1 y 2
- 100 encuestas de usuarios del chatbot con calificaciones de 3, 4 y 5
- 392699 conversaciones del año 2021, con 1495683 interacciones del usuario.
- 29584 conversaciones del año 2022, con 730816 interacciones del usuario.

El conjunto de datos de 2022 incluye un atributo que permite identificar las conversaciones que hicieron desborde manual por solicitud explícita del usuario. El conjunto de datos de 2021 no cuenta con este atributo. Sin embargo, se realizó un etiquetado manual de algunas conversaciones a partir de los datos disponibles, con el sentimiento y la cantidad de frases repetidas. Esta etiqueta es la única diferencia entre los conjuntos de datos de 2021 y los de 2022.

Con el fin de analizar la correlación entre la calificación numérica dada por el usuario y el sentimiento del texto escrito por el mismo con respecto a la atención del chatbot, se usaron los dos conjuntos de datos de las encuestas. El texto se tomó del atributo “ResultadoEncuestaCuentanosObservaciones” de las conversaciones y se comparó con la calificación tomada del atributo “ResultadoEncuestaFacilidad”. Se observó que, si bien los escritos de algunos usuarios presentan sentimientos negativos, otorgan una buena calificación a la atención y viceversa. Esto pasa en un 7% de los datos de las 200 encuestas analizadas. La

Figura 3, muestra los resultados de algunas encuestas que se considera que no tienen consistencia entre la calificación y el texto escrito por el usuario.


```
{
    "Id": "46T77IOrHxhLE0CERdKN9c-6",
    "ResultadoEncuestaFacilidad": "1",
    "ResultadoEncuestaCuentanosObservaciones": "Me equivoqué con el número"
},
{
    "Id": "EmE2xiH75qOEiCpdr0YV8I-6",
    "ResultadoEncuestaFacilidad": "2",
    "ResultadoEncuestaCuentanosObservaciones": "5"
},
{
    "Id": "JB8xZU6uMBL5XVIdLGAme-6",
    "ResultadoEncuestaFacilidad": "3",
    "ResultadoEncuestaCuentanosObservaciones": "hasta ahora no hemos hablado nadda"
},
{
    "Id": "G40e7ScHH2I7oP7kIxd0lP-j",
    "ResultadoEncuestaFacilidad": "4",
    "ResultadoEncuestaCuentanosObservaciones": "POR QUE NO ME DIERON LA RESPUESTA QUE QUERIA\n"
}
```


Figura 3. Calificaciones de las conversaciones

Por otro lado, para los conjuntos de datos de las conversaciones, se identificaron las interacciones con sentimientos positivo, negativo y neutro. Se analizaron las conversaciones que tuvieron más de 2 frases negativas y se compararon con las calificaciones dadas por el usuario en estas conversaciones, para validar la correlación. En este caso, se utilizaron 989 conversaciones, con un total de 6832 interacciones.

De las 989 conversaciones, 88 fueron calificadas por el usuario. En las que presentan más de 2 frases negativas, se observó que algunos usuarios, manifiestan sentimientos negativos en sus interacciones y otorgan una buena calificación a la atención. En la Figura 4, se puede observar que, si bien predominan las calificaciones bajas, también se presentan algunas calificaciones altas para estas conversaciones.

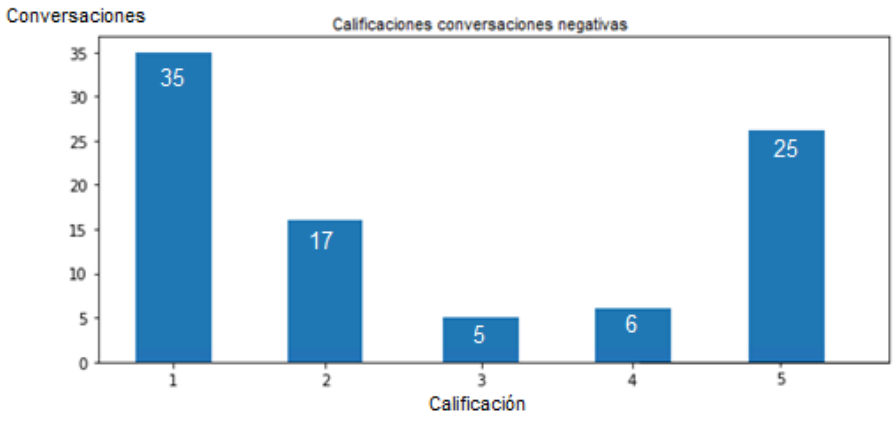

Figura 4. Calificaciones de las conversaciones con frases negativas

También, se comparó la calificación dada por el usuario contra el sentimiento general de la conversación, basado en el sentimiento entregado por VADER. En la Figura 5, no se aprecia una correlación entre estas dos variables.

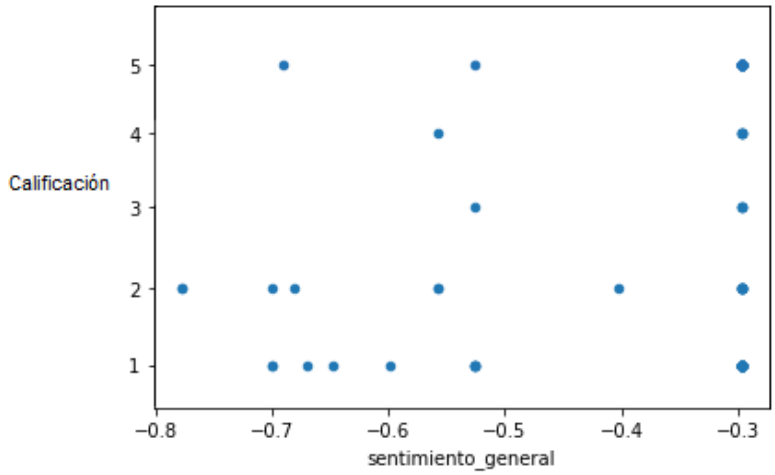

Figura 5. Calificaciones de las conversaciones con sentimiento negativo vs el sentimiento general de la conversación

Además, se analizaron las conversaciones que tuvieron más de 2 frases positivas contra sus calificaciones, para validar si estas tienen calificaciones altas. De 159 conversaciones seleccionadas, 16 tienen calificación dada por el usuario. En la Figura 6, se puede observar que, si bien las conversaciones consideradas como positivas tienen calificaciones altas y bajas, predominan más las calificaciones altas (de 3 a 5).

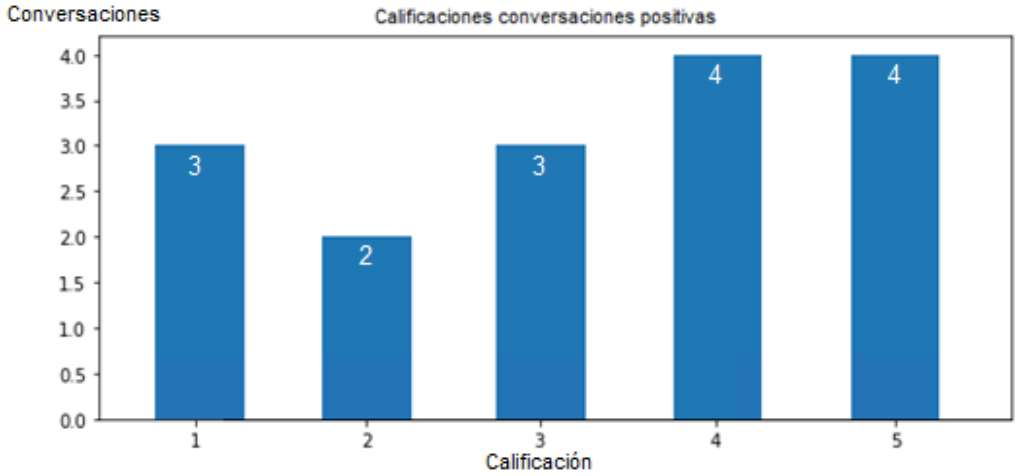

Figura 6. Calificaciones de las conversaciones con frases positivas

De forma general, los clientes no tienden a calificar la atención; de 392699 conversaciones, solo 23223, que corresponde a un 5%, tienen calificación. Además, en algunos casos, el usuario resolvió su inquietud y otorgó una baja calificación, mientras que existen conversaciones en las que el usuario manifestó que no resolvió su inquietud y sin embargo envió una calificación entre 3 y 5. El análisis preliminar realizado, no permite concluir que la calificación esté correlacionada con el sentimiento general de la conversación. Tampoco da muestra de si el usuario pudo resolver o no su inquietud, por lo cual no se tomará como atributo para entrenar un modelo de decisión de desborde/no desborde.

Por otro lado, se analizaron los tipos de diálogos más comunes, los cuales dan una idea de los tipos de transacciones que más utilizan los usuarios con el chatbot. En la Figura 7, se puede observar el histograma de los tipos de diálogo más comunes. Este atributo no se tuvo en cuenta para el modelo, dado que no se considera representativo para el propósito de este proyecto.

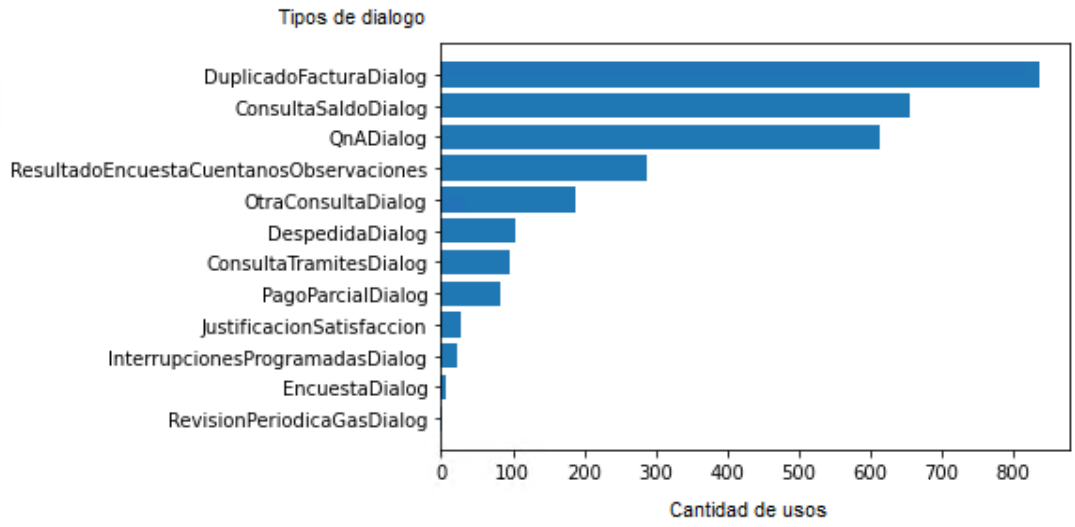

Figura 7. Tipos de dialogo más comunes

En la Figura 8, se presentan las palabras más utilizadas en las interacciones del usuario en conversaciones marcadas como que hicieron desborde y las conversaciones sin desborde de forma independiente.

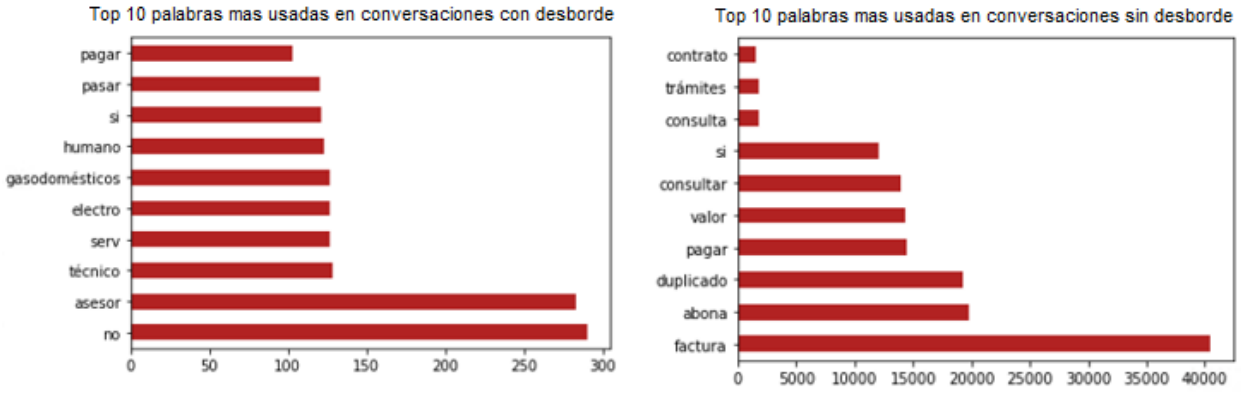

Figura 8. Top 10 palabras más utilizadas

La mayoría de las palabras utilizadas por los usuarios, corresponden a acciones de los trámites que efectivamente se pueden realizar a través del chatbot y son consistentes con el vocabulario del proceso de atención al cliente de la empresa. También en las conversaciones con desborde, se identifica la recurrencia en el uso de las palabras asesor y humano, lo cual es coherente con la funcionalidad que tiene del desborde manual haciendo uso de ellas. La palabra “no” fue removida de la lista de *stopwords,* dado que se considera que es importante para la precisión del análisis del sentimiento.

La Figura 9, muestra en una nube, los lemas de las palabras más comúnmente utilizadas en las interacciones de los usuarios, extraídas de las conversaciones con el chatbot.

Figura 9. Nube de palabras

La Figura 10 presenta la clasificación del sentimiento de las conversaciones que hicieron y no hicieron desborde.

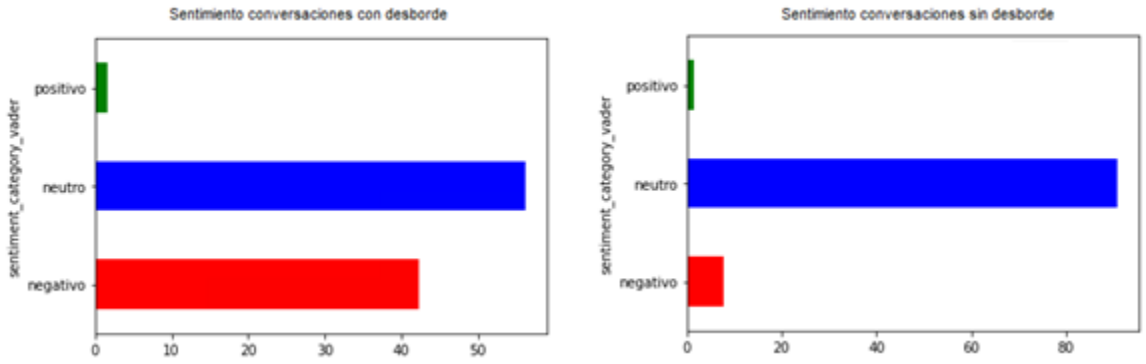

Figura 10. Clasificación del sentimiento de las interacciones del usuario

Al analizar el sentimiento de las interacciones de las conversaciones que realizaron desborde, se evidencia que éste tiende a ser neutro seguido del sentimiento negativo. Para las conversaciones que no realizaron desborde, se nota la mayoría de las interacciones con un sentimiento neutro y si bien está presente el sentimiento negativo, éste no es tan alto como en las conversaciones que si realizaron desborde. Sin embargo, el sentimiento negativo sigue estando más presente que el sentimiento positivo.

De todas las conversaciones que son calificadas, la mayoría de las calificaciones son 5 (la más alta), lo cual se aprecia en la gráfica izquierda de la Figura 11. La distribución del sentimiento promedio de todas las conversaciones tiende a ser negativo, lo cual puede ser una alerta para los responsables del proceso de atención al cliente.

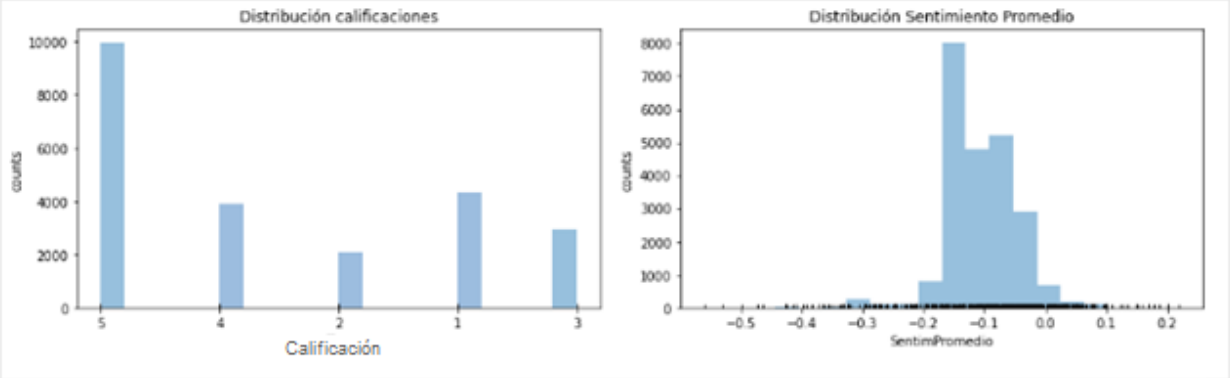

Figura 11. Distribución de las calificaciones y del promedio del sentimiento en las conversaciones

Desagregando el análisis del sentimiento por interacciones, se evidencia que predomina el sentimiento negativo en las que tienen desborde, mientras que para las que no hicieron desborde, predomina el sentimiento neutro. Al revisar todo el conjunto de conversaciones en general, se nota que predomina el sentimiento neutro, lo cual se puede observar en la Figura 12.

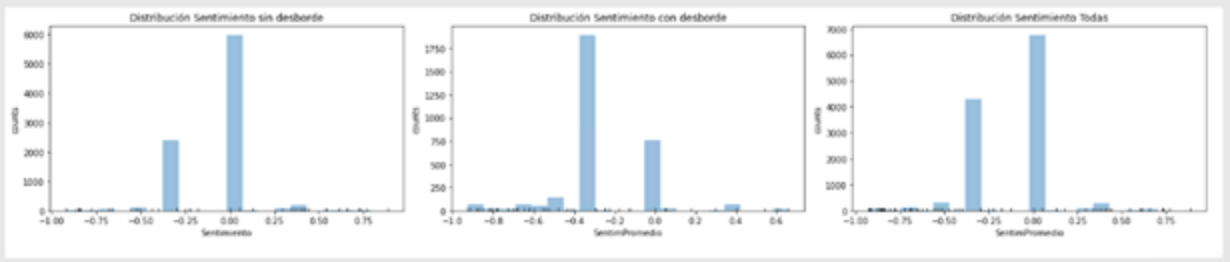

Figura 12. Distribución del sentimiento en las conversaciones

Además, se analizó la cantidad de interacciones de acuerdo con el desborde o no de las conversaciones. En la Figura 13, se puede observar que se distribuyen de manera muy similar al conjunto total de conversaciones. Sin embargo, se nota un leve aumento en la cantidad de frases para las conversaciones que no hacen desborde, lo cual indica que son conversaciones más largas.

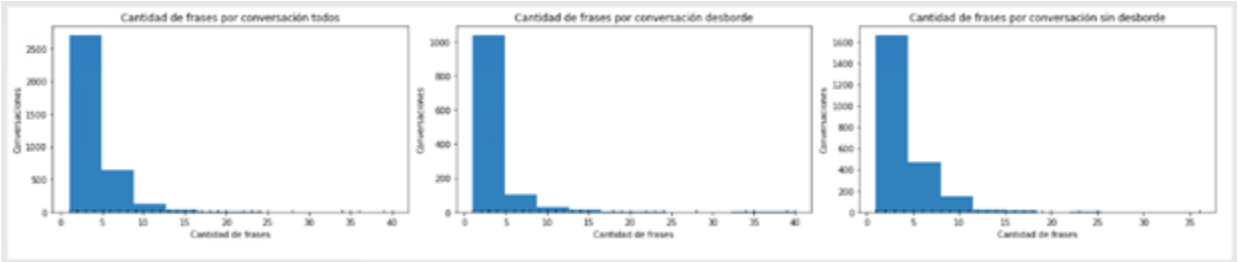

Figura 13. Cantidad de frases por conversación

Este conjunto de conversaciones tiene un valor máximo de 40 interacciones por conversación y un mínimo de 1 interacción. Al analizar los percentiles para el atributo de cantidad de interacciones o frases por conversación, se encuentra que el percentil 99 de las conversaciones tienen 15 frases o menos. El 75% de las conversaciones tienen 4 frases o menos, el 50% de las conversaciones tienen 3 frases o menos y el 25% de las conversaciones tienen 1 sola frase. Esto apoyó la decisión de usar los 15 primeros sentimientos como atributos para el conjunto de datos con el que se entrenaría el modelo de estimación del momento de desborde.

Para el conjunto de datos final, se obtuvieron 2331 conversaciones que no hicieron desborde y 1201 conversaciones que realizaron desborde, para un total de 3532 conversaciones. Éste es el que se usa para entrenar el modelo de estimación del momento de desborde. Este conjunto de datos cuenta con 66% de los datos que no realizaron desborde y un 34 % de las conversaciones que si realizaron desborde. Estas conversaciones tienen en promedio 6 interacciones (utterances) del usuario.

La Figura 14, presenta gráficamente, la cantidad de conversaciones que hicieron y no hicieron desborde.

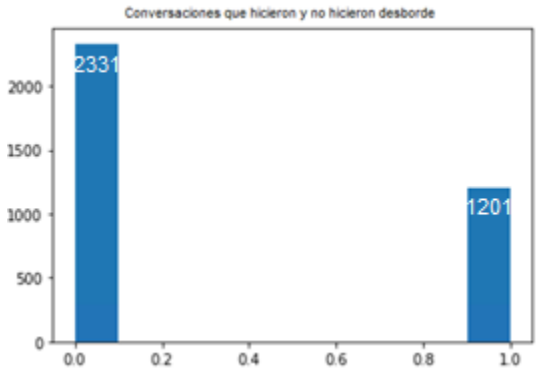

Figura 14. Cantidad de conversaciones con desborde/no desborde

La Tabla 1 describe el conjunto de datos utilizados para el modelo de desborde.

	compound	total_conversaciones	sentimiento_general	desborde
count	12121.000000	12121.000000	12121.000000	12121.000000
mean	-0.050773	6.176966	-0.119614	0.261695
std	0.140979	5.729328	0.209271	0.439575
min	-0.927400	1.000000	-0.927400	0.000000
25%	0.000000	3.000000	-0.296000	0.000000
50%	0.000000	5.000000	0.000000	0.000000
75%	0.000000	7.000000	0.000000	1.000000
max	0.883400	40.000000	0.883400	1.000000

Tabla 1. Descripción del conjunto de datos

Conjunto de datos para entrenamiento

Se utilizaron 2 conjuntos de datos. El primero, es usado en los algoritmos de etiquetado basados en reglas, con los que se pretende marcar el desborde o no de cada conversación. El segundo conjunto de datos es el resultado de la aplicación del algoritmo de etiquetado y se usa para entrenar un modelo supervisado de clasificación.

Si bien desde la adquisición de los datos se tiene una etiqueta de “desborde/ no desborde”, ésta no se considera lo suficientemente robusta para entrenar el modelo que prediga el momento de transferir a un asesor humano, dado que es un desborde solicitado por el mismo usuario y se identificó que algunos de ellos solicitan un asesor humano desde la primera interacción, sin presentar aún un nivel de frustración, no solución o insatisfacción con la atención del chatbot. Usar esta etiqueta sesgaría el modelo y podría llevar a realizar más desbordes de los que el proceso de atención al cliente puede soportar. Además, no fue posible obtener un conjunto de datos etiquetado por un experto del negocio. Por esto, se propone el algoritmo de etiquetado de desborde/ no desborde, el cual se explica más adelante.

El conjunto de conversaciones para etiquetar cuenta con 9 atributos y 12121 registros. Cada registro corresponde a una interacción del usuario en una conversación con el chatbot. En total se tienen 3532 conversaciones. El conjunto de datos que se requiere etiquetar cuenta con los atributos presentados en la Tabla 2.

Variable	Descripción	Tipo de dato
id_conversacion	Identificador de la conversación	String
dialogo	Tipo de transacción o interacción	String
texto	Texto enviado por el usuario	String
texto_limpio	Texto preprocesado y limpio. Se genera a partir del atributo texto.	String
texto_lemmatizado	Texto limpio lematizado	String
compound	Valor del sentimiento de la interacción, va desde -1 (el más negativo) hasta 1 (el más positivo)	Float
sentiment_category_vader	Categoría del sentimiento: negativo, positivo o neutro.	String
total_conversaciones	Cantidad total de interacciones del usuario en una conversación.	Float
sentimiento_general	Sentimiento general de la conversación calculado a partir del compound.	Float

Tabla 2. Conjunto de datos de conversaciones para etiquetar

Este conjunto de datos es transformado para obtener el segundo conjunto de datos, que se utiliza para el modelo de desborde. De la aplicación del algoritmo de etiquetado, se obtiene la variable objetivo de desborde. Se toman los primeros 15 compound de cada conversación como atributos de sentimiento. Se toman 15 ya que el percentil 99 de las conversaciones tienen hasta 15 frases. Además, se calculan la cantidad de interacciones y la cantidad de frases similares de cada conversación y se agrupan por el identificador de la conversación, quedando el conjunto de datos con 3532 registros y 19 atributos, los cuales son presentados en la Tabla 3.

Variable	Descripción	Tipo de dato
cant_frases	Cantidad total de interacciones del usuario en una conversación	Float
frases_similares	Cantidad de frases similares que contiene la convesación	Float
compound0	Compound del sentimiento de la primera frase de la conversación	Float
compound 1	Compound del sentimiento de la frase 2 de la conversación	Float
compound 2	Compound del sentimiento de la frase 3 de la conversación	Float
compound 3	Compound del sentimiento de la frase 4 de la conversación	Float
compound 4	Compound del sentimiento de la frase 5 de la conversación	Float
compound 5	Compound del sentimiento de la frase 6 de la conversación	Float
compound 6	Compound del sentimiento de la frase 7 de la conversación	Float

compound 7	Compound del sentimiento de la frase 8 de la conversación	Float
compound 8	Compound del sentimiento de la frase 9 de la conversación	Float
compound 9	Compound del sentimiento de la frase 10 de la conversación	Float
compound 10	Compound del sentimiento de la frase 11 de la conversación	Float
compound 11	Compound del sentimiento de la frase 12 de la conversación	Float
compound 12	Compound del sentimiento de la frase 13 de la conversación	Float
compound 13	Compound del sentimiento de la frase 14 de la conversación	Float
compound 14	Compound del sentimiento de la frase 15 de la conversación	Float
desborde_hat	Variable respuesta. Indica si la conversación hace desborde (1) o no (0).	Int
id_conversacion	Identificador de la conversación	String

Tabla 3. Conjunto de datos de conversaciones etiquetadas

3.4 Preprocesamiento de los datos

Del conjunto de interacciones de cada conversación, se utilizaron solamente las que escribió el usuario, es decir, se eliminaron las del chatbot. Además, se excluyeron las interacciones que corresponden a términos y condiciones, ya que no agregan información al conjunto de datos.

Se realizó limpieza del atributo texto:

1. Eliminación de nulos.
2. Se llevó todo a minúscula.
3. Eliminación de páginas Web (textos que comienzan por http).
4. Eliminación de signos de puntuación.
5. Eliminación de números.
6. Eliminación de espacios en blanco múltiples.
7. Eliminación de espacios en blanco al inicio.
8. Eliminación de textos nulos.

La Tabla 4, presenta un ejemplo del resultado de la limpieza del texto.

Who	texto	texto_limpio
User	Serv. técnico electro/gasodomésticos	serv técnico electro gasodomésticos
User	Porque no ha llegado el agua	porque no ha llegado el agua
User	Agua	agua
User	Reportar daño	reportar daño
User	Consulta de trámites	consulta de trámites
...	...	...
User	Sí	sí
User	Abona a tu factura	abona a tu factura
User	Consulta de trámites	consulta de trámites
User	Otros	otros
User	Consultar el estado de una Solicitud o Caso	consultar el estado de una solicitud o caso

Tabla 4. Texto original y pre procesado

Las conversaciones atípicas fueron eliminadas, por ejemplo, para el conjunto de datos de las conversaciones que no hacen desborde, se tenían conversaciones con más de 100 interacciones, lo cual no se considera una conversación típica. También se eliminaron algunas frases duplicadas, obtenidas por error desde la adquisición de los datos.

Para el conjunto de datos de 2021 se creó un nuevo atributo llamado desborde, que indica si la conversación realiza (1) o no realiza desborde (0), calculado a partir del sentimiento general de la conversación y la cantidad acumulada de interacciones similares. Este atributo no fue posible validarlo con los expertos de negocio.

Como el conjunto de conversaciones que no realizan desborde son más que las que si hacen desborde, se seleccionaron las que tenían interacciones con más prosa, haciendo uso del atributo Dialogo con el valor “QnADialog” y de las conversaciones que incluyen notas de voz, identificadas porque su texto comienza con “Origen nota de voz”.

Además, se realizó tokenización y lematización del texto limpio con el fin de hallar la bolsa de palabras. Se eliminaron las palabras de parada con ayuda del listado de *stopwords* de la librería NLTK. De esta lista, se eliminó la palabra “no” ya que se considera de valor para el análisis realizado y se adicionaron a la lista, las palabras: “nota”, “voz” y “origen”, ya que son palabras que el sistema agrega por defecto al texto cuando son notas de voz.

Con ingeniería de características se crearon nuevas variables:

- Cantidad de interacciones de la conversación. Se realizó un conteo agrupando por el identificador de la conversación.
- Sentimiento general de la conversación. A partir del sentimiento dado por VADER, se calculó el mínimo, el máximo y el promedio de los sentimientos de cada conversación, si el promedio es negativo, se asigna como sentimiento general el mínimo sentimiento y si es positivo, se asigna el máximo, de lo contrario se asigna el promedio.
- Categoría sentimiento VADER. De acuerdo con el *compound*, se creó una nueva columna de tipo texto, que indica si el sentimiento es positivo, negativo o neutro.

Los datos fueron normalizados con *MinMaxScaler* de la librería *sklearn*, para que los atributos estuvieran en la misma escala.

La Figura 15 presenta la información de los atributos del conjunto de datos, luego de la preparación de los datos.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 12121 entries, 0 to 8948
Data columns (total 10 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   id_conversacion          12121 non-null  object
 1   dialogo                  12121 non-null  object
 2   texto                    12121 non-null  object
 3   texto_limpio             12121 non-null  object
 4   texto_lemmatizado        12121 non-null  object
 5   compound                 12121 non-null  float64
 6   sentiment_category_vader 12121 non-null  object
 7   total_conversaciones     12121 non-null  int64
 8   sentimiento_general      12121 non-null  float64
 9   desborde                 12121 non-null  int64
dtypes: float64(2), int64(2), object(6)
memory usage: 1.3+ MB
```

Figura 15. Información de los datos pre procesados

4 DESARROLLO DE MODELOS

Este trabajo se realizó siguiendo la metodología CRISP-DM (Cross-Industrie Standard Process for Data Mining), la cual incluye descripciones de las fases típicas de un proyecto de analítica, las tareas involucradas en cada fase y una explicación de las relaciones entre estas tareas. Como modelo del proceso, CRISP-DM proporciona una visión general del ciclo de vida de la minería de datos (IBM, 2020), independiente del área en la que se realice la investigación.

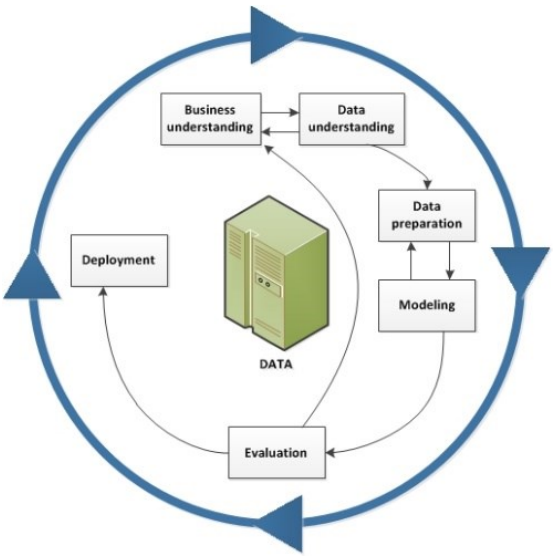

Figura 16. Ciclo de vida de minería de datos (IBM, 2020)

A continuación, se explica el desarrollo de las actividades realizadas en las diferentes fases del ciclo CRISP-DM.

4.1 Entendimiento del negocio

El desarrollo de este proyecto está enmarcado en una empresa de servicios públicos domiciliarios, para la cual es de gran importancia la atención al cliente dada la regulación que la rige, además de su responsabilidad social, que es el hilo que articula y les da sentido a las decisiones de la empresa.

Entre las alternativas de atención al cliente que ofrece, se encuentra un chatbot que permite a los clientes autogestionarse en un amplio portafolio de transacciones. Éste cuenta con tres canales: Web, WhatsApp y robot físico, y está construido con modelos de inteligencia artificial para resolver los requerimientos de los usuarios. A partir de 2022, el canal de WhatsApp cuenta con un mecanismo manual de desborde para la atención con asesores humanos, en el que, en caso de ser necesario, éste debe ser indicado explícitamente por el usuario mediante la escritura de la palabra “asesor”.

En esta fase, se realizaron reuniones con los distintos actores del proceso de atención al cliente, entre ellos, los usuarios funcionales responsables del proceso y los lideres técnicos del chatbot. Se realizaron reuniones de lluvia de ideas, explicaciones del funcionamiento técnico del chatbot, funcionalidades actuales y se revisaron las expectativas con respecto al funcionamiento futuro y capacidades de éste.

Adicionalmente, en esta fase, se definieron los objetivos y el alcance de los modelos.

4.2 Entendimiento de los datos

El chatbot con el cual se trabajó, permite al usuario autoatenderse para una cantidad limitada de transacciones. Su nombre particular es: Ema.

Ema inició funcionamiento en 2019, con 3 transacciones y cien mil clientes. En 2020 se incrementó el flujo de datos a 700.000 (setecientas mil) transacciones. En septiembre de 2020, con la implementación del primer robot de autoatención, la cantidad de transacciones aumentó considerablemente. Además, en 2021 salió a producción el canal WhatsApp. En 2022, se implementó una funcionalidad de desborde manual, con la cual el usuario puede solicitar comunicarse con un asesor humano, de forma explícita, escribiendo la palabra “asesor”.

En esta fase, se realizó un análisis descriptivo de los datos en el que se determinó el preprocesamiento y la preparación requerida, como se describe en el numeral “3.4 Preprocesamiento de los datos”.

Con el fin de entender mejor los datos, se analizó la forma de tratarlos y abordarlos a partir de una muestra de las conversaciones entre el chatbot y los usuarios. Se partió de un conjunto de datos conformado por 423186 conversaciones (2230699 registros), provenientes de una base de datos “Cosmos DB”. Se exportaron las muestras a archivos en formato JSON y luego se almacenaron en archivos Excel para efectos de este proyecto. La Tabla 5 presenta el tamaño del conjunto de datos original.

Año de los datos	Cantidad de interacciones	Cantidad de conversaciones
2021	1495683	392699
2022	735016	30487
Total	2230699	423186

Tabla 5. Conjunto de datos original

En la Figura 17, se puede observar un extracto de una conversación, muestra de los datos que se utilizaron:


```
"UtteranceList": {
    "$type": "System.Collections.Generic.List`1[[AsistenteComercial.Model.UtteranceMessage, Asist
    "$values": [
        {
            "$type": "AsistenteComercial.Model.UtteranceMessage, AsistenteComercial",
            "Id": 1,
            "Dialogo": "ConsultaSaldoDialog",
            "Who": "User",
            "Text": "Consultar el valor a pagar",
            "Date": "2021-03-24T21:32:08.1433983"
        },
        {
            "$type": "AsistenteComercial.Model.UtteranceMessage, AsistenteComercial",
            "Id": 2,
            "Dialogo": "ConsultaSaldoDialog",
            "Who": "Asistente",
            "Text": "Para consultar el valor a pagar, digita el número de contrato de tu hogar
            "Date": "2021-03-24T21:32:08.144043"
        },
```


Figura 17. Extracto conversación usuario y chatbot

De cada conversación, se extrae el fragmento “UtteranceList” y en éste se identifican los valores del asistente y del usuario. Los valores contienen las sentencias de cada uno de los actores. De los 89 atributos de las conversaciones, se seleccionaron:

- Id: El identificador de la conversación.
- Dialogo: Tipo de transacción.
- Text: El texto.
- Who: indica si la interacción la realizó el chatbot o el usuario.

La Tabla 6, muestra las interacciones de un usuario en una conversación.

IdConversacion	Texto
27055	hola porque no me permite hacer el pago en linea?
27055	me dice error efectuando el pag0: supera valo...
27055	que hago? gracias!!
27055	ya tengo el numero me meto y no me deja pagar
27055	ya tengo el numero de contrato y el valor
27055	alo
27055	alo
27055	alo
27055	si esta correcto
27055	si esta correcto
27055	si esta correcto
27055	que puedo hacer?

Tabla 6. Interacciones de un usuario en una conversación

Los datos y sus estadísticas se describen en el numeral “3.3 Descripción y análisis preliminar de los datos”.

4.3 Preparación de los datos

Las conversaciones se encontraban originalmente en formato JSON. Para preparar los datos, se realizó:

- Extracción de cada una de las conversaciones, separando los fragmentos de conversación del usuario y del asistente (chatbot).
- Extracción de las características de los datos.
- Limpieza de los textos del usuario, con transformaciones básicas sobre las sentencias, como paso a minúsculas y eliminación de caracteres especiales.
- Selección de los atributos de interés.
- Normalización de los datos como se describe en el numeral “3.4 Preprocesamiento de los datos”.
- Vectorización de los textos para hallar similitudes.
- Hallar la bolsa de palabras.
- Lematización de los textos.

Además, se realizó una trasposición de los sentimientos de los textos escritos por los usuarios para utilizarlos como atributos del modelo de desborde. Esto se realizó para las primeras 15 interacciones de cada conversación, para finalmente obtener el conjunto de datos para entrenamiento, como se describe en el numeral “3.3 Descripción y análisis preliminar de los datos”.

4.4 Modelado

Se realizaron varios experimentos en los que se utilizaron técnicas de análisis de sentimientos, se construyeron algoritmos basados en reglas para etiquetar las conversaciones con “desborde” o “no desborde” y finalmente con las conversaciones etiquetadas, se exploraron modelos de aprendizaje de máquina para predecir si una conversación debe desbordar a asesor humano o no. La Figura 18, presenta el flujo de trabajo realizado.

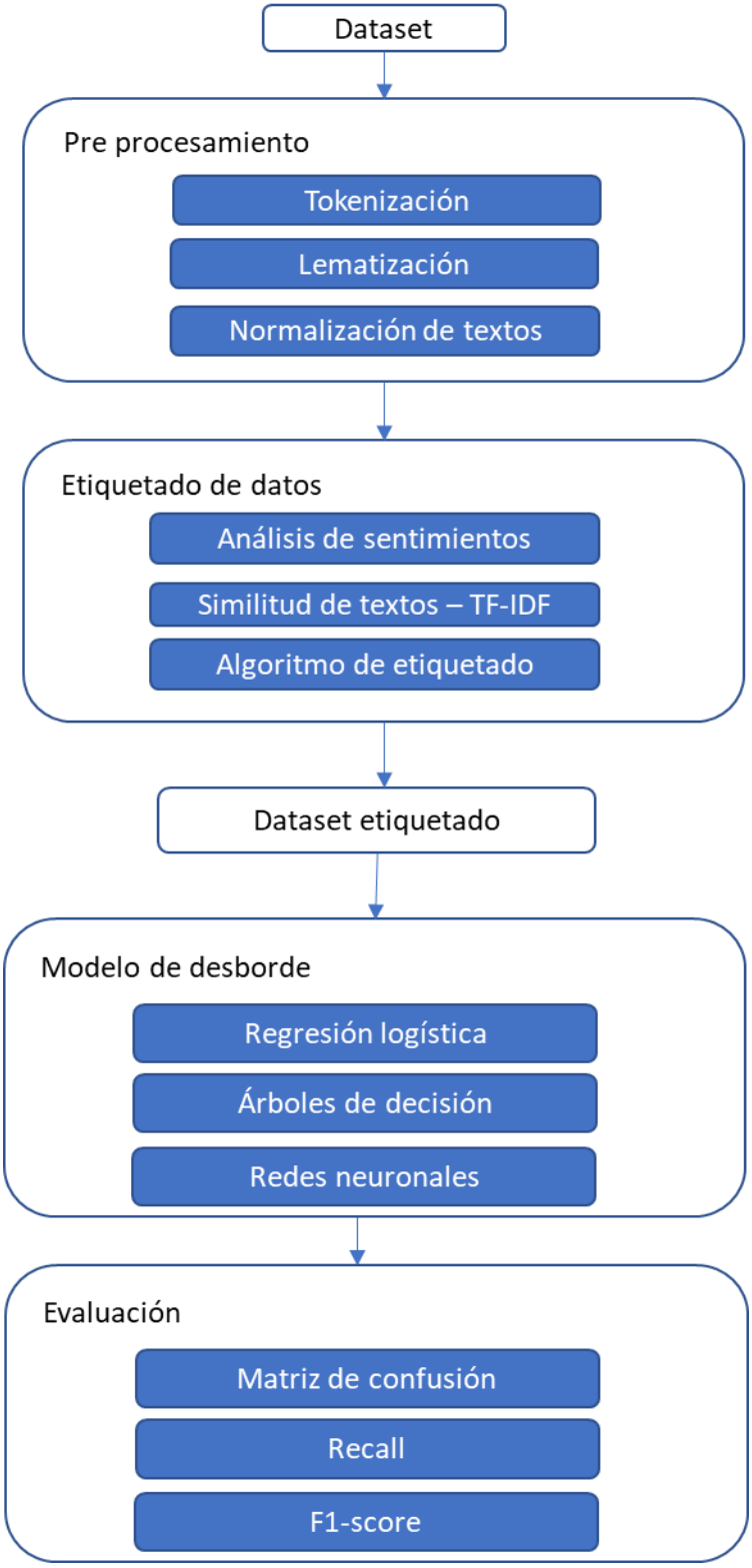

Figura 18. Flujo de trabajo

4.4.1 Análisis de sentimientos de las interacciones

En esta etapa, se realizó análisis de sentimientos de los textos enviados por los usuarios. Dado que no se cuenta con los datos con el sentimiento etiquetado, se experimentó con modelos preentrenados basados en reglas y de aprendizaje automático: AFINN, VADER y TextBlob. Se analizaron 1067880 frases y se seleccionó VADER, siendo el modelo que mejor clasificó los textos en español.

Al revisar la clasificación de las interacciones con AFINN, el cual asigna a cada palabra un valor entre -5 y 5, siendo -5 el máximo de negatividad y +5 el máximo de positividad, no se observó el comportamiento esperado. Por ejemplo, para la interacción presentada en la Tabla 7, se esperaba un valor negativo y el resultado fue un valor positivo.

texto	sentimiento
La página no sirve para nada la experiencia del usuario es super mala	3

Tabla 7. Clasificación del sentimiento con AFINN

Para la clasificación con TextBlob, los textos fueron traducidos de español a inglés con la función *translate* de TextBlob y se halló la polaridad del sentimiento a los textos traducidos. En la Tabla 8 se observa, como VADER logró identificar una frase que se considera negativa, mientras que TextBlob no la detectó como negativa.

Texto_limpio	traduccion	polaridad	setimiento_textblob	nltk_results	compound	neg_vader	pos_vader	neu_vader	sentiment_category_vader
las lineas no sirven para nada	The lines do not serve at all	0.000000	neutro	{'neg': 0.306, 'neu': 0.694, 'pos': 0.0, 'comp...	-0.296	0.306	0.0	0.694	negativo

Tabla 8. Sentimiento VADER y sentimiento TextBlob

La Figura 19, presenta las gráficas de la clasificación del sentimiento de los textos pre procesados con VADER y TextBlob.

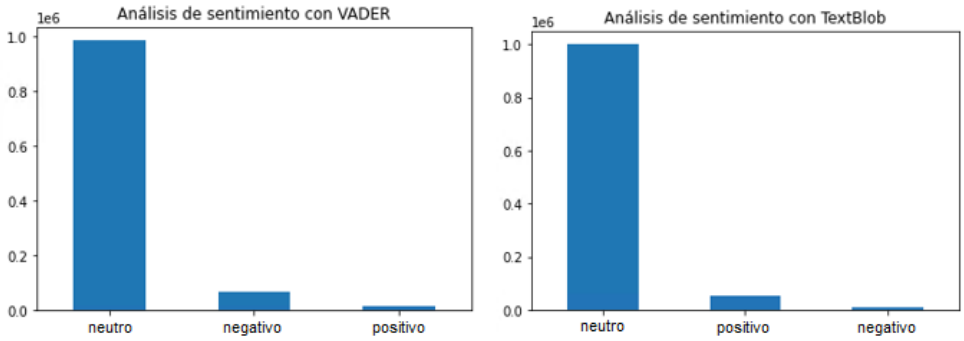

Figura 19. Cantidad interacciones sentimiento VADER y TextBlob

Mientras VADER detectó más frases negativas, TextBlob detectó más frases positivas. Al revisar manualmente, se encontró más acertada la clasificación de VADER, esto puede deberse al hecho de que para usar TextBlob, las frases fueron traducidas al inglés y luego clasificadas.

En este caso, no se cuenta con el valor real de la variable objetivo. Sin embargo, se realizó una revisión manual general de los 3 modelos utilizados y se seleccionó el que más se ajusta al idioma y tipo de texto enviado por los usuarios del chatbot. Se seleccionó VADER como el mejor modelo.

4.4.2 Análisis de similitud de las interacciones

Se analizó la similitud entre los textos continuos de la misma conversación para detectar preguntas o sentencias reiteradas, lo cual combinado con la tendencia del sentimiento de la conversación, sirve de base para la construcción del algoritmo de etiquetado de desborde/no desborde de las conversaciones explicado en el numeral “4.4.3 Algoritmo de etiquetado de las conversaciones”.

Posteriormente, se utilizó *TfidfVectorizer* para vectorizar cada interacción y la interacción inmediatamente anterior; se calculó el conteo de palabras, los valores IDF y las puntuaciones TF-IDF, para luego hallar la similitud de coseno de los vectores. Además, se estableció el umbral de similitud como hiperparámetro del modelo. Si la similitud de coseno supera este umbral, entonces se dice que las frases son similares.

4.4.3 Algoritmo de etiquetado de las conversaciones

El algoritmo de etiquetado marca las conversaciones para identificar las que deben hacer desborde a asesor humano y las que no. Se plantea debido a que la etiqueta de desborde que se tiene actualmente en la fuente de datos se toma de las conversaciones que realizan desborde manual, es decir, por solicitud explícita del usuario, para el conjunto de datos de 2022. El conjunto de datos de 2021 no cuenta con esta etiqueta, aunque fue calculada con base en los datos disponibles, pero no fue posible validarla ni obtenerla con un experto de negocio. Si se crea un modelo basado en estas etiquetas, es posible que se desborden más conversaciones de las necesarias o más de las que el proceso pueda soportar.

Actualmente, el negocio cuenta con 20 asesores para atender estos desbordes, por lo cual se debe tener especial cuidado en no desbordar conversaciones que efectivamente pueden ser atendidas por el chatbot. Sin embargo, es de mayor importancia que el modelo logre identificar las conversaciones que si deben transferirse a un asesor humano, para prevenir el abandono de la conversación o la insatisfacción del cliente.

Para el etiquetado de desborde/no desborde, se propusieron 3 algoritmos basados en reglas, los cuales parten de la teoría de que las conversaciones que desbordan se pueden identificar a partir del sentimiento de sus interacciones, la repitencia de frases y la cantidad total de interacciones de la conversación. Se utilizó el conjunto de datos descrito en la Tabla 2. Éste incluye el sentimiento calculado con VADER y el total de conversaciones. A continuación, se explica el trabajo realizado en cada uno de los algoritmos:

- **Algoritmo 1: Distribución negativa**

Este algoritmo se basó en la función de aumento de Prasad y Akana (2021), en el cual se tienen en cuenta los sentimientos de las últimas 3 interacciones del usuario. Si el sentimiento más reciente es más negativo que el anterior y a su vez, este es más negativo que el primero y además el sentimiento de la última interacción supera un umbral, se define que la conversación debe realizar desborde.

La función planteada es la siguiente:

```
function etiquetar_conversacion(sentimiento_frase, negDistribution[])

     desborde = 0

     if(negDistribution[2]<= negDistribution[1]

        and negDistribution[1] <= negDistribution[0]

        and negDistribution[0] <= umbral_desborde)

     then

        desborde = 1
```

Con esta función, no se obtuvo un buen modelo, ya que no logró identificar las conversaciones que debían hacer desborde. La validación se realizó contra los datos que tienen el atributo desborde desde la fuente (manual). En la Figura 20, se puede observar a la izquierda, el gráfico de la cantidad de conversaciones que desbordaron realmente y las que no y, en el gráfico de la derecha la cantidad de conversaciones que desbordan y las que no, según el etiquetado realizado por el algoritmo basado en la distribución negativa de las interacciones. Los resultados de este algoritmo se presentan en el numeral “4.5 Evaluación”.

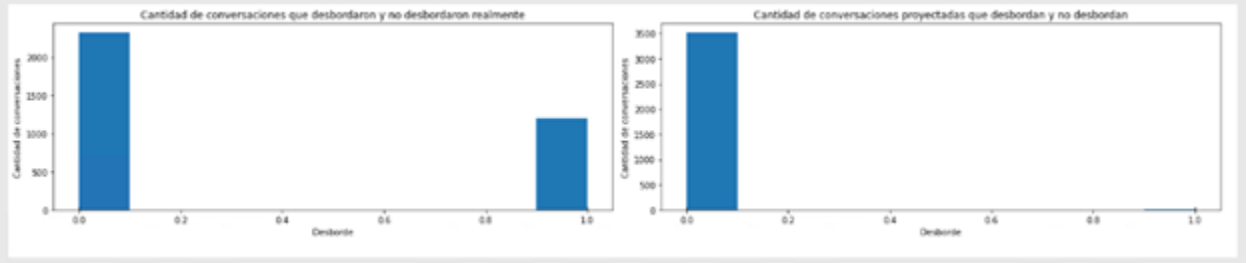

Figura 20. Desborde/no desborde con función distribución negativa

- **Algoritmo 2: Acumulación no lineal del sentimiento**

Se creó un algoritmo basado en una acumulación no lineal del sentimiento de cada una de las interacciones. Se estableció un hiperparámetro como umbral de desborde con valor de 0.7, y se utilizó una variable que acumula el sentimiento de la conversación. A continuación, se presenta la función:

```
function etiquetar_conversacion(sentimiento_frase, sentimiento_acumulado)
        desborde = 0
        if(abs(sentimiento_frase) ! = 1) then
               sentimiento_acumulado - = log (sentimiento_frase + 1)
        else
               sentimiento_acumulado - = sentimiento_frase
       if(sentimiento_acumulado < 0) then
               #Sentimientos positivos no acumulan menos de cero.
               sentimiento_acumulado = 0
        # Si el sentimiento acumulado supera el umbral, se dice que
        # la conversación debe desbordar
        if(sentimiento_acumulado  >  umbral_desborde) then
        desborde = 1
```

Con este algoritmo se pretende acumular de una manera no lineal el sentimiento. Los sentimientos negativos suman, mientras que los sentimientos positivos restan hasta llegar a 0 como punto mínimo. Los resultados de este algoritmo se presentan en el numeral “4.5 Evaluación”.

- **Algoritmo 3: Combinado acumulación del sentimiento, repitencia y cantidad de interacciones.**

Se creó un tercer algoritmo, combinando el sentimiento acumulado, la repitencia y la cantidad de interacciones en la conversación, creando umbrales como hiperparámetros y estableciendo sus valores iniciales según la experiencia con los anteriores algoritmos. Además, se realizó un recorrido en forma de malla, modificando los hiperparámetros para conocer cuál sería la mejor combinación. Se plantea la siguiente ecuación para etiquetar las conversaciones:

$$\hat{y} = \begin{cases} 1,\ si\ \omega_1\left(\max\left(\sum_{i=0}^{n} -C(x_i), 0\right)\right) + \omega_2\left(\sum_{i=1}^{n} cos\ sim(x_i, x_{i-1}) + (i/(n*10))\right) - u \geqq 0 \\ 0, si\ \omega_1\left(\max\left(\sum_{i=0}^{n} -C(x_i), 0\right)\right) + \omega_2\left(\sum_{i=1}^{n} cos\ sim(x_i, x_{i-1}) + (i/(n*10))\right) - u < 0 \end{cases} \quad (1)$$

Donde:

```
ω1: peso sentimiento
C(): compound del sentimiento calculado con VADER
ω2: peso similitud
cossim(): similitud de coseno. 1 si las frases son similares, 0 si no lo son.
u: umbral de desborde
```

A continuación, se expone la función propuesta:

```
function etiquetar_conversacion(conversacion_usuario, peso_sentimiento,
                            peso_repitencia, umbral_desborde, umbral_similitud):
    desborde = 0
    sentimiento_acumulado = 0
    cantidad_similares = 0
    for(frase in conversacion_usuario)do
        sentimiento_acumulado -= frase.sentimiento
        # Se vectorizan la frase y la frase anterior
        vectores = vectorizar(frase, frase_anterior)
               similitud = cosine_similarity(vectores)
               if(similitud > umbral_similitud) then
                  cantidad_similares += (+ 1 + (posicion_frase/
                                        (total_frases*10)))
               puntaje = (cantidad_similares * peso_repitencia) +
                         (sentimiento_ acumulado * peso_sentimiento)
        if(puntaje > umbral_desborde) then
               desborde = 1
```

En la Figura 21, se puede observar a la izquierda, el gráfico de la cantidad de conversaciones que desbordaron realmente y las que no y en el gráfico de la derecha las cantidades según el etiquetado realizado por el algoritmo basado en la combinación del sentimiento acumulado, la repitencia y la cantidad de frases en la conversación, con la segunda mejor combinación de hiperparámetros encontrada:

- peso_sentimiento = 3.3
- peso_repitencia = 0.7
- umbral_desborde = 1
- umbral_similitud = 0.4

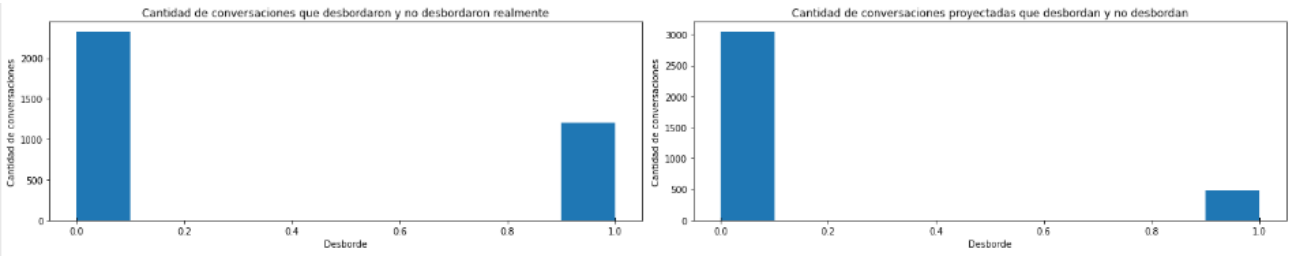

Figura 21. Desborde/no desborde con función combinada

Con la mejor combinación de hiperparámetros encontrada, el algoritmo identifica como desborde más conversaciones que el conjunto de datos original, como se puede observar en la Figura 22. La mejor combinación de hiperparámetros encontrada es la siguiente:

- peso_sentimiento = 3.5
- peso_repitencia = 0.75
- umbral_desborde = 1
- umbral_similitud = 0.4

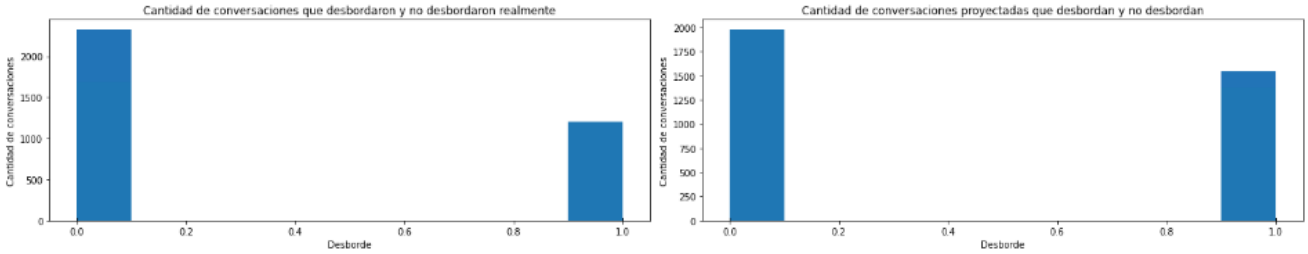

Figura 22. Desborde/no desborde con mejor hiperparametrización

Utilizar esta configuración de hiperparámetros, podría llevar al modelo a desbordar más conversaciones de las necesarias y, sobre todo, más de las que el negocio podría atender. Los resultados de este algoritmo se presentan en el numeral “4.5 Evaluación”.

4.4.4 Modelo de aprendizaje de máquina para la predicción del desborde

Se realizaron experimentos para entrenar un modelo a partir de una serie de variables de entrada, con el fin de predecir la variable respuesta: desborde o no desborde de la conversación entre un usuario y un chatbot, es decir, para predecir la conmutación de la conversación a un asesor humano, combinando información del análisis de sentimiento, la repitencia y la cantidad de interacciones en la conversación.

Como datos de entrada, se utilizó el conjunto de conversaciones etiquetado con el algoritmo de etiquetado 3 combinado, tomando los sentimientos de las 15 primeras interacciones del usuario en la conversación, como atributos y las columnas “cantidad de frases similares” y “cantidad de frases de la conversación”. Este conjunto de datos cuenta con 3532 registros y 19 atributos, incluida la variable respuesta.

Los datos fueron previamente preprocesados (ver numeral 3.4) y se obtuvieron de la siguiente manera:

1. Cálculo del compound del sentimiento con VADER para cada interacción.
2. Selección de los primeros 15 compound para cada conversación y se traspusieron en el conjunto de datos, quedando cada valor como atributo.
3. Cálculo de la cantidad de interacciones de la conversación, realizando un conteo y agrupando por el identificador de la conversación. Este dato se normaliza para que tome un valor entre 0 y 1.
4. Cálculo de la cantidad de interacciones contiguas similares en cada conversación con similitud de coseno. Este dato es agrupado y se normaliza para que tome un valor entre 0 y 1.
5. La variable respuesta se toma de la etiqueta calculada con el algoritmo 3 combinado.

La Tabla 9 describe el conjunto de datos usado en estos modelos de aprendizaje automático para la predicción del desborde.

	cant_frases	frases_similares	0	1	2	3	4	5	6	7	8
count	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000
mean	0.062353	0.005111	-0.090479	-0.022904	-0.015027	-0.014724	-0.006883	-0.006068	-0.003537	-0.003221	-0.001517
std	0.078712	0.029913	0.172342	0.099070	0.078772	0.074692	0.059490	0.053242	0.041420	0.040061	0.033920
min	0.000000	0.000000	-0.862500	-0.918600	-0.880700	-0.915300	-0.778300	-0.680800	-0.872000	-0.859100	-0.865800
25%	0.000000	0.000000	-0.296000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.051282	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	0.076923	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
max	1.000000	1.000000	0.765000	0.883400	0.612400	0.401900	0.599400	0.726900	0.401900	0.401900	0.401900

9	10	11	12	13	14	desborde
3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000	3532.000000
-0.001342	-0.000589	-0.000520	-0.000391	-0.000870	-0.000793	0.135617
0.025008	0.018093	0.015258	0.018667	0.018188	0.021595	0.342430
-0.778300	-0.526700	-0.526700	-0.526700	-0.526700	-0.680800	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.296000	0.401900	0.051600	0.401900	0.051600	0.296000	1.000000

Tabla 9. Conjunto de datos etiquetado para el modelo de desborde

Se dividió el conjunto de datos en 2 subconjuntos: para entrenamiento (80%) y validación (20%). En este caso, no se usó un subconjunto de datos adicional para test. Se podría en el futuro, particionar el subconjunto de validación, en dos, para así obtener 3 subconjuntos de datos: entrenamiento, validación y test. Se entrenaron diferentes modelos de clasificación clásicos: regresión logística y árboles de decisión, además de un modelo con redes neuronales y se probó su respectivo ajuste con el subconjunto de validación. Se revisaron la matriz de confusión y las métricas de evaluación y se seleccionó el modelo con mejor desempeño. A continuación, se presentan los modelos que se entrenaron con sus respectivos resultados.

- **Regresión logística**

La regresión logística fue utilizada debido a su simplicidad y facilidad de interpretación. Se utilizó el conjunto de conversaciones etiquetado (ver Tabla 3), normalizado. Con este modelo, se obtuvo un resultado aceptable el cual se detalla en el numeral “4.5 Evaluación”. Sin embargo, para el negocio es importante que el modelo logre identificar un alto porcentaje de las conversaciones que deben hacer desborde, por lo tanto, se requiere un *recall* más alto que el que resulta de la regresión logística con estos datos.

Por consiguiente, se realiza balanceo de los datos para tener más conversaciones en la clase que desborda del conjunto de datos. Esto se realiza con la librería *RandomOverSampler* de *imblearn*. Sin el balanceo, el conjunto de datos tiene 3053 conversaciones que no hacen desborde (con etiqueta de desborde en 0) y 479 que si hacen desborde. Al realizar el sobre muestreo, se obtienen 3053 conversaciones que no hacen desborde y 3053 conversaciones que

si lo hacen. Al entrenar el modelo de regresión logística con los datos balanceados, se obtienen mejores resultados.

Adicionalmente, se realiza un experimento de balanceo de datos con *RandomUnderSampler* al 80%, en el que se eliminan aleatoriamente conversaciones de la clase con mayor número de observaciones, en este caso, las conversaciones que no hacen desborde, para que la clase minoritaria quede en un 80% de la clase mayoritaria. Así, la cantidad de conversaciones que no desbordan pasan de 3053 a 598 y las conversaciones que, si desbordan, quedan en la misma cantidad.

Este modelo fue entrenado sin especificación de hiperparámetros, con un 80% de los datos. Se realizó validación cruzada con *k-fold,* con un k de 10.

- **Árboles de decisión**

Se experimentó utilizando árboles de decisión, logrando obtener un buen modelo. Se utilizó *DecisionTreeClassifier* de *sklearn* y el conjunto de conversaciones etiquetado (ver Tabla 3), normalizado, balanceado por debajo, con un total de 1077 conversaciones. Este modelo, generó un árbol con 95 nodos, cuya estructura se muestra a en la Figura 23. Este modelo fue entrenado sin especificación de hiperparámetros, con un 80% de los datos.

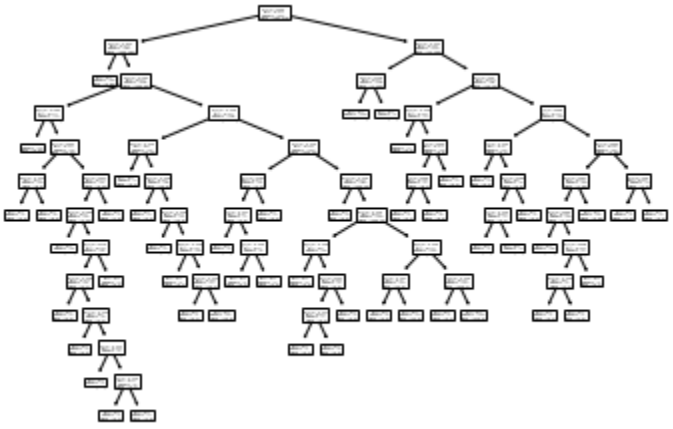

Figura 23. Estructura del árbol de decisión

- **Redes neuronales**

Se utilizó el conjunto de conversaciones etiquetado (ver Tabla 3), normalizado. Se realizó una experimentación básica con redes neuronales sencillas para obtener resultados preliminares y visualizar que tan alentador sería seguir profundizando en los modelos de redes neuronales en un trabajo futuro. Se utilizó un modelo de red neuronal retroalimentada con tipo de capa dense que es la que se usa más comúnmente. La red creada tiene una serie de capas de neuronas secuenciales, es decir, una delante de otra. Se definió una capa de entrada con 17 neuronas (para los atributos del conjunto de datos) y una capa oculta de 51 neuronas. Como función de activación se utilizó '*relu'* y se agregó una capa con 1 neurona de salida y función de activación sigmoide, ya que el problema de clasificación es binario. Así, la arquitectura de la red queda de

3 capas. También, se configuró el tipo de pérdida (*loss*) con *'mean_squared_error'* y el optimizador de los pesos de las conexiones de las neuronas con '*adam*'.

Se entrenaron modelos variando el número de épocas y el conjunto de datos balanceado. A continuación, se presentan los resultados del entrenamiento con 20 y con 30 épocas, resultando mejor el modelo entrenado con 20 épocas.

4.5 Evaluación

Los resultados de los algoritmos y modelos propuestos se evalúan desde la perspectiva de un problema de clasificación binaria en el que las clases corresponden al desborde o no desborde de una conversación, entre el usuario y un chatbot de servicio al cliente.

El desempeño de los modelos se determinó mediante la métrica de *F1-score*, una métrica para medir calidad de la clasificación que pondera las métricas de *precision* y *recall*. Si bien se analizó el *accuracy*, este no se tuvo en cuenta para la selección de los modelos. Estas métricas parten del concepto de la matriz de confusión que, en el caso binario, consiste en una tabla con las 4 combinaciones entre los valores predichos y reales del conjunto de pruebas.

A continuación, se describen las métricas que se usaron para evaluar los algoritmos y modelos:

- ***Precision:*** indica la porción de conversaciones que realmente desbordan, del total de conversaciones que el modelo predijo como desborde.
- ***Recall:*** indica del total de conversaciones que realmente desbordan, el porcentaje que el modelo es capaz de detectar como desborde.
- ***F1-score:*** es una métrica que permite medir *precision* y *recall* simultáneamente.

Para el algoritmo de etiquetado, se exploraron diferentes técnicas basadas en reglas para generar etiquetas de predicción de desborde o no desborde, las cuales se compararon con las etiquetas reales del conjunto de conversaciones. Para medir el desempeño de la metodología, se analizaron la matriz de confusión y las métricas de evaluación antes mencionadas.

Para el algoritmo 1 de etiquetado basado en distribución negativa, se obtuvo un *accuracy* de 0.66, *precision* de 0.83, *recall* de 0.004 y un *F1-score* de 0.008. En la Figura 24, se presenta la matriz de confusión resultante de este algoritmo.

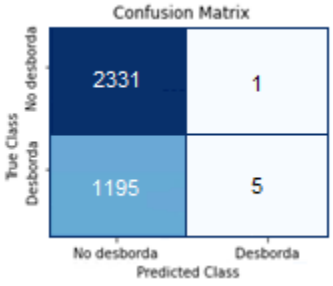

Figura 24. Matriz de confusión distribución negativa

En el algoritmo 2 de etiquetado con acumulación no lineal del sentimiento, el resultado mejoró con respecto al primer algoritmo, pero aún no se considera un buen modelo. Se obtuvo un *accuracy* de 0.69, *precision* de 0.68, *recall* de 0.205 y un F1-score de 0.316. En la Figura 25, se presenta la matriz de confusión, se puede observar que aún no detecta bien las conversaciones que deben desbordar.

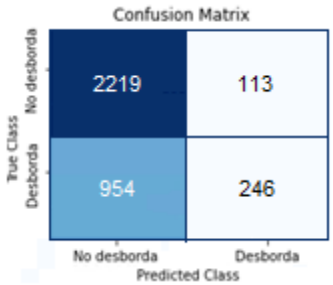

Figura 25. Matriz de confusión acumulación no lineal del sentimiento

De los diferentes algoritmos que se exploraron, el que mejor resultado presentó fue el 3, que combina variables: la acumulación del *compound* del sentimiento dado por VADER, la repitencia, dada por la similitud de coseno entre interacciones contiguas y la cantidad de interacciones en la conversación.

Para este algoritmo 3, se realizó una validación de las métricas de evaluación con varias combinaciones de hiperparámetros, utilizando una malla de valores. Se analizaron las 2 mejores combinaciones de hiperparámetros y se seleccionó la segunda mejor combinación, dado que la mejor combinación presenta un comportamiento no deseable para el proceso de negocio, como se explicó en el numeral “4.4.3 Algoritmo de etiquetado de las conversaciones”.

Los resultados se muestran en la Figura 26. Se nota una mejoría al usar la siguiente configuración de hiperparámetros:

- peso_sentimiento = 3.3
- peso_repitencia = 0.7
- umbral_desborde = 1
- umbral_similitud = 0.4

	precision	recall	f1-score	support
0	0.70	0.92	0.80	2331
1	0.62	0.25	0.35	1201
accuracy			0.69	3532
macro avg	0.66	0.58	0.58	3532
weighted avg	0.68	0.69	0.65	3532

Figura 26. Resultados algoritmo combinado

En la Figura 27, se presentan los resultados del algoritmo con la mejor combinación de hiperparámetros encontrada:

- peso_sentimiento = 3.5
- peso_repitencia = 0.75
- umbral_desborde = 1
- umbral_similitud = 0.4

	precision	recall	f1-score	support
0	0.89	0.76	0.82	2331
1	0.63	0.82	0.71	1201
accuracy			0.78	3532
macro avg	0.76	0.79	0.77	3532
weighted avg	0.80	0.78	0.78	3532

Figura 27. Mejores resultados algoritmo combinado

Se observan mejores resultados del algoritmo 3. Sin embargo, debido a que se usaron los datos etiquetados desde la fuente para realizar esta validación, se logra identificar que con esta configuración de hiperparámetros, el modelo se ajusta a unas observaciones cuyo desborde se realiza por parte del usuario, sin mostrar ningún nivel de frustración, en algunos casos con solo una interacción negativa. Además, identifica varias conversaciones con “desborde” que en el conjunto de datos original no realizan desborde, es decir, identifica como desborde más conversaciones de las que debería.

Con respecto al modelo de aprendizaje de máquina para desborde, se experimentaron diferentes modelos de clasificación clásicos y un modelo simple de redes neuronales, para predecir el desborde/no desborde de las conversaciones. La evaluación se realizó comparando la predicción con la variable objetivo, obtenida del algoritmo de etiquetado. Para medir el desempeño de los modelos, se analizaron la matriz de confusión y las métricas de evaluación *recall*, *precision* y *F1-score*. Adicionalmente, para la regresión logística, se realizó una validación cruzada con *k-fold* y entre éstos, se seleccionó el mejor. Para el k-fold, se utilizó el conjunto de conversaciones etiquetadas normalizado, un k = 10 y un tamaño del conjunto de validación del 20%.

La Figura 28 muestra los resultados de la aplicación del modelo de regresión logística.

```
Son 2825 datos para entrenamiento y 707 datos para prueba
Logistic Regression: 0.936636 (0.013376)
0.942008486562942
[[622   2]
 [ 39  44]]
              precision    recall  f1-score   support

           0       0.94      1.00      0.97       624
           1       0.96      0.53      0.68        83

    accuracy                           0.94       707
   macro avg       0.95      0.76      0.83       707
weighted avg       0.94      0.94      0.93       707
```

Figura 28. Resultados regresión logística

La regresión logística con los datos balanceados con el sobre muestreo arrojan los resultados que se presentan en la Figura 29.

```
Son 4884 datos para entrenamiento y 1222 datos para prueba
Logistic Regression: 0.920957 (0.015619)
0.9206219312602292
[[581  26]
 [ 71 544]]
              precision    recall  f1-score   support

           0       0.89      0.96      0.92       607
           1       0.95      0.88      0.92       615

    accuracy                           0.92      1222
   macro avg       0.92      0.92      0.92      1222
weighted avg       0.92      0.92      0.92      1222
```

Figura 29. Resultados regresión logística con oversampling

La Figura 30 presenta el resultado de la regresión logística con el submuestreo de los datos.

```
Son 861 datos para entrenamiento y 216 datos para prueba
Logistic Regression: 0.907070 (0.013840)
0.8888888888888888
[[112   3]
 [ 21  80]]
              precision    recall  f1-score   support

           0       0.84      0.97      0.90       115
           1       0.96      0.79      0.87       101

    accuracy                           0.89       216
   macro avg       0.90      0.88      0.89       216
weighted avg       0.90      0.89      0.89       216
```

Figura 30. Resultados regresión logística con undersampling

Se encuentra que el modelo de regresión logística con sobre muestreo tiene mejor desempeño general que los modelos que utilizaron los datos sin balancear y con el submuestreo.

En el entrenamiento del modelo de desborde, los mejores resultados se obtuvieron con el árbol de decisión, usando los datos normalizados y balanceados con el submuestreo. La Figura 31 muestra estos resultados.

```
Precisión Árboles de Decisión Clasificación: 0.9259259259259259
[[111   4]
 [ 12  89]]
              precision    recall  f1-score   support

         0.0       0.90      0.97      0.93       115
         1.0       0.96      0.88      0.92       101

    accuracy                           0.93       216
   macro avg       0.93      0.92      0.93       216
weighted avg       0.93      0.93      0.93       216
```

Figura 31. Resultados árbol de decisión

Se observa un modelo con muy buen desempeño, con mejores resultados que la regresión logística en las diferentes versiones experimentadas. Este modelo también superó el resultado de la red neuronal utilizada. Si bien se puede continuar explorando otro tipo de redes neuronales para optimizar el modelo, esto se propone como trabajo futuro. La Figura 32 muestra el resultado de la red neuronal entrenada con 20 épocas.

	precision	recall	f1-score	support
0.0	0.88	0.99	0.93	115
1.0	0.99	0.85	0.91	101
accuracy			0.93	216
macro avg	0.94	0.92	0.92	216
weighted avg	0.93	0.93	0.93	216

Figura 32. Resultado redes neuronales con 20 épocas

La Figura 33 muestra el resultado de la red neuronal entrenada con 30 épocas.

	precision	recall	f1-score	support
0.0	0.88	0.98	0.93	115
1.0	0.98	0.85	0.91	101
accuracy			0.92	216
macro avg	0.93	0.92	0.92	216
weighted avg	0.93	0.92	0.92	216

Figura 33. Resultado redes neuronales con 30 épocas

Se observa mejor *precision* para la clase con desborde en la red neuronal entrenada con 20 épocas.

4.6 Despliegue

La fase de despliegue no está dentro del alcance de este proyecto.

5 ANÁLISIS DE RESULTADOS

El proyecto inició con un análisis exploratorio de los datos, con el fin de conocer y comprender las conversaciones entre el usuario y el chatbot y encontrar correlaciones entre las variables que se querían utilizar. De este análisis se encontró que la calidad y cantidad de datos es buena, para realizar el ejercicio. Sin embargo, se evidencia la necesidad de plantear un mecanismo de etiquetado automático de las conversaciones, previo al entrenamiento del modelo de desborde.

Con las calificaciones dadas por los clientes a las conversaciones con el chatbot, se revisó si el chatbot resolvió o no el problema. Se planteó la hipótesis de que, si el usuario envió una calificación baja (puntuación 1 o 2), podría ser un criterio para que el modelo marque la conversación para realizar el desborde a un asesor humano, dado que no resolvió el requerimiento del cliente. Al realizar análisis exploratorio de los datos, no se encontró una correlación entre la calificación del usuario y la resolución del problema, tampoco se encontró correlación con el sentimiento general de la conversación.

Adicionalmente, se analizó la cantidad total de calificaciones bajas con el fin de revisar si éstas pudiesen ser directamente proporcionales a la cantidad de conversaciones que se debían haber redirigido a un asesor humano. Sin embargo, tampoco se pudo comprobar esta relación. Algunas conversaciones a las que, si se les dio solución, fueron calificadas con una puntuación baja y algunas conversaciones a las que no se les dio solución, fueron calificadas con una puntuación alta.

Con el análisis exploratorio de la calificación, también se encontró que, en el conjunto de datos, solo el 5% de los usuarios otorgaron una calificación y de estas calificaciones, sobresalen las de puntuación más alta (5). En contraste, el sentimiento promedio de estas conversaciones tiende a ser negativo.

Por lo anterior y debido a que no es posible esperar a que la conversación termine o que el usuario otorgue una calificación para realizar el desborde, la calificación se descarta como atributo para entrenar el modelo de aprendizaje de máquina de desborde.

También, se analizó el atributo tipo de dialogo, del cual se pudo observar que ofrece buena visualización de los tipos de proceso que son más solicitados por los usuarios en el chatbot. Sin embargo, este atributo no ofrece información para el modelo objetivo de este proyecto, por lo cual no se hace uso de éste.

Al realizar un análisis a nivel de texto y el vocabulario de las interacciones de los usuarios, se evidencia una frecuencia mayor en el uso de palabras que son propias de las acciones de los trámites que efectivamente se pueden realizar a través del chatbot y son consistentes con el vocabulario del proceso de atención al cliente de la empresa. En las conversaciones que hacen

desborde, se identifica la recurrencia en el uso de las palabras asesor y humano, lo cual es coherente con la funcionalidad que tiene el chatbot del desborde manual.

Con respecto a los datos se encontró que:

- Los datos de 2021 no cuentan con una etiqueta de desborde/no desborde. Por lo cual se creó manualmente, una etiqueta basada en criterios de sentimientos y repitencia.
- Los datos de 2022 cuentan con etiqueta de desborde/no desborde, dada por la solicitud explícita del usuario en cualquier momento de la conversación.

Al analizar y experimentar con este atributo como variable objetivo, no se encontró adecuada para entrenar un modelo de aprendizaje de máquina de desborde, lo cual creó la necesidad de obtener un mecanismo de etiquetado de los datos, por lo cual se incluyó una experimentación con algoritmos de etiquetado basados en reglas.

Luego del preprocesamiento y la preparación de los datos, se cuenta con datos de buena calidad y con la cantidad suficiente para realizar el ejercicio de modelado.

Para el análisis de sentimientos, VADER fue el modelo que más se ajustó a la clasificación de los textos escritos por los usuarios en las conversaciones con el chatbot. Esto fue validado comparando las clasificaciones realizadas por los diferentes modelos (AFINN, TextBlob, VADER), tomando un subconjunto de los datos en el que los resultados eran diferentes y revisando manualmente si la interacción debía ser positiva, negativa o neutra.

Con respecto al análisis de la cantidad de interacciones por conversación, se observó un leve aumento en la cantidad de frases para las conversaciones que no hacen desborde, con respecto a las que, si lo hacen, lo cual indica que las conversaciones que no hacen desborde tienen a ser más largas. Además, el percentil 99 de las conversaciones tienen hasta un máximo de 15 interacciones, lo cual permitió establecer en 15, la cantidad de sentimientos que se usarían como atributos para entrenar el modelo de desborde.

Por otro lado, se creó un algoritmo de etiquetado de desborde/no desborde de las conversaciones entre el usuario y el chatbot de servicio al cliente, se seleccionó el algoritmo 3 combinado, que marca la conversación según el sentimiento acumulado, la repitencia, la cantidad de frases en la conversación y los pesos establecidos, ya que presenta el mejor desempeño y se ajusta al objetivo de etiquetado que se requiere.

En la evaluación de este algoritmo, se evidenció que no era conveniente utilizar la mejor combinación de hiperparámetros encontrada, dado que el modelo se ajusta a unas observaciones cuyo desborde se realiza por parte del usuario, sin mostrar ningún nivel de frustración, en algunos casos con solo una interacción negativa.

Además, identifica varias conversaciones con “desborde” que en el conjunto de datos original no están marcados como desborde, es decir, identifica como desborde más conversaciones de las que debería, lo cual podría llevar a la transferencia de conversaciones que no lo requieren o en un momento dado, a colapsar el proceso de atención al cliente debido a la alta cantidad de conversaciones que se transferirían a un asesor humano, contrastada con los 20 asesores que se tienen actualmente para la atención. Por lo anterior, se selecciona, la segunda mejor combinación de hiperparámetros encontrada.

En la Tabla 10, se muestra la diferencia entre el conjunto de datos con la etiqueta de desborde desde el origen y el conjunto de datos con la etiqueta calculada según las reglas del algoritmo combinado, con la configuración de hiperparámetros seleccionada.

Desborde	Etiqueta original	Etiqueta calculada
0	2331	3053
1	1201	479

Tabla 10. Cantidad de conversaciones con etiqueta calculada

El uso del sentimiento, la similitud entre las interacciones del usuario y la cantidad de interacciones, para el etiquetado automático del conjunto de datos conversacionales del chatbot de servicio al cliente, es un aspecto esencial para superar la limitación de los datos etiquetados con el desborde manual o los datos etiquetados manualmente, lo que generaría un posible riesgo de sesgo humano. Con esto, se logra detectar de manera inteligente el momento de desborde de las conversaciones y de manera automática y así poder tomar las decisiones sobre la continuación de la atención de la solicitud del usuario.

Como resultado del análisis exploratorio de los datos, se identificaron, además, las variables requeridas para implementar el modelo que estima la tendencia de no solución de la interacción entre el usuario y el chatbot de servicio al cliente:

- Los sentimientos de las primeras 15 interacciones del usuario, representadas numéricamente con el *compound* dado por VADER.
- La cantidad de interacciones en la conversación.
- La cantidad de interacciones similares contiguas del usuario.

Para el modelo de desborde, se midieron: el *accuracy*, la *precision*, el *recall* y el *F1-score* para comparar el desempeño de los diferentes modelos de clasificación del desborde de la conversación. Se observó que todos los modelos no se comportan mejor con los datos balanceados por encima o con los datos balanceados por debajo, por lo cual para cada modelo se exploró con ambos conjuntos de datos balanceados y de cada uno, se tomó el que mejor resultados tuvo.

Finalmente, se seleccionó el modelo de árbol de decisión que da el mejor ajuste para realizar la conmutación a un asesor humano. Si bien los 3 modelos se consideran buenos y tienen un desempeño muy similar, el árbol de decisión presenta el mejor *recall* y *F1-score* para la clase de desborde, que es el centro del interés de este proyecto. En la Tabla 11, se pueden apreciar los resultados de los diferentes modelos entrenados.

		Clase desborda			Clase No desborda		
Modelo	Accuracy	Precision	Recall	F1-score	Precision	Recall	F1-score
Árbol de decisión	0.93	0.96	0.88	0.92	0.90	0.97	0.93
Regresión logística	0.92	0.95	0.88	0.92	0.89	0.96	0.92
Redes neuronales	0.93	0.99	0.85	0.91	0.88	0.99	0.93

Tabla 11. Resultados evaluación modelo de desborde

A partir de los resultados, se observa que fue posible obtener un modelo a partir de la clasificación de sentimientos de los textos del usuario en la conversación, la similitud de textos y la cantidad de interacciones del usuario, para detectar la frustración o tendencia a la no solución de la necesidad de éste y así poder manejar la situación con el desborde a un asesor humano.

6 CONCLUSIONES Y TRABAJO FUTURO

Los chatbot de atención al cliente se emplean para que los clientes puedan realizas sus consultas, problemas o quejas, pero todavía están muy lejos de ser lo suficientemente naturales y resolver realmente todos los requerimientos del usuario. Es importante entonces tratar de inferir mediante algún mecanismo la tendencia en solución o no de este requerimiento. Para esto, los algoritmos de aprendizaje de máquina son ideales.

El aporte principal de este trabajo consiste en el diseño y construcción de un algoritmo de etiquetado y un modelo de desborde de la conversación para predecir el momento de transferir la conversación a un asesor humano, cuando se detecte que no se está resolviendo el requerimiento del usuario o éste se esté frustrando con la conversación.

Para plantear el modelo de desborde, era clave contar con datos etiquetados que facilitaran el entrenamiento de un modelo de aprendizaje de máquina. Los datos originales de trabajo plantean la etiqueta de desborde en dos formas: la primera es una etiqueta calculada manualmente a partir de los atributos del conjunto de datos y la segunda es una etiqueta que traen los datos desde el origen dado por el desborde manual que solicita explícitamente el usuario. Sin embargo, en el análisis exploratorio de los datos, se encontró que la etiqueta que se tenía no era consistente ni suficientemente robusta.

Como una propuesta innovadora, se plantearon 3 algoritmos de etiquetado de las conversaciones entre el chatbot y el usuario, que permitieron marcar las conversaciones con las clases desborde/no desborde. Dichos algoritmos son:

1. Función de distribución de la negatividad (Prasad y Akana, 2021).
2. Acumulación no lineal del sentimiento.
3. Combinación de la acumulación del sentimiento, interacciones similares contiguas y cantidad de interacciones del usuario.

El algoritmo 3 tuvo los mejores resultados y por lo tanto fue seleccionado. Del análisis de resultados se puede concluir que este algoritmo podría ser utilizado además como un mecanismo de etiquetado de los datos en tiempo real o para que el chatbot tenga un primer acercamiento para la conmutación a asesor humano. Además, puede servir para etiquetar datos históricos para posteriormente entrenar un modelo.

El conjunto de datos etiquetado con el algoritmo seleccionado fue usado para entrenar modelos de aprendizaje de máquina con regresión logística, árboles de decisión y una red neuronal sencilla. Los modelos fueron explorados y evaluados con diferentes versiones de los datos: con muestreo por encima y con submuestreo. Los resultados indican que el árbol de decisión entrenado con un submuestreo de los datos es el modelo que mejor desempeño tiene y, además que la etiqueta de los datos es consistente para desarrollar un modelo de aprendizaje de máquina.

Como resultado, se obtuvo un modelo de aprendizaje de máquina basado en árboles de decisión, que permite identificar la tendencia de no solución en la interacción del usuario con el chatbot de servicio al cliente, combinando el análisis de sentimientos, la repitencia y el total de interacciones del usuario, con el fin de predecir la conmutación a un asesor humano cuando se identifique frustración del usuario (dada por la tendencia negativa del sentimiento acumulado) o la tendencia de no solución de la atención (dada por la repitencia). En síntesis, el modelo predice si el usuario debe continuar siendo atendido por el chatbot o si debe ser transferido a un asesor humano,

usando los atributos identificados para lograr mayor precisión en el modelo, lo cual se presenta como algo novedoso para este tipo de proyectos.

En este proyecto se demuestra que el algoritmo de etiquetado creado permite tener un conjunto de datos con una variable respuesta consistente. La cual está basada en los atributos identificados en el análisis exploratorio de los datos, a saber: sentimiento acumulado de las interacciones del usuario, cantidad de interacciones contiguas similares y cantidad de interacciones del usuario.

Así mismo, tener el conjunto de datos con la variable respuesta dada por el algoritmo de etiquetado permitió construir un modelo de desborde, que permite estimar el momento en el que la conversación entre un chatbot y un usuario debe ser transferida a un asesor humano.

Las principales contribuciones de este proyecto se exponen a continuación:

1. Identificación de las variables que permiten establecer los algoritmos y modelos para realizar el desborde de la conversación a un asesor humano.
2. Algoritmo de etiquetado automático que además de etiquetar el conjunto de datos, también puede usarse para que el chatbot prediga cuando es el momento de hacer desborde a asesor humano.
3. Modelo de desborde basado en árboles de decisión, con un resultado satisfactorio para resolver el problema planteado, liviano y rápido.

Si bien el modelo planteado ha mostrado un buen desempeño (*F1-score* de 0.92), la identificación de la frustración o no solución del requerimiento de un usuario en una conversación con un chatbot es un proceso difícil debido a la misma complejidad del lenguaje natural de los seres humanos. Sin embargo, el modelo planteado puede apoyar este proceso reduciendo el abandono temprano del chat.

Finalmente quedaría plantear cuáles pueden ser las líneas futuras de este proyecto. Como continuación del trabajo desarrollado, una línea futura inmediata podría ser la optimización del modelo, adicionando como hiperparámetros el número de asesores disponibles y el encolamiento máximo permitido por el negocio.

Además, se propone crear un modelo de ensamble con el algoritmo de reglas combinado y el modelo de árbol de decisión, con el fin de optimizar el modelo de estimación del desborde de la conversación.

También se propone continuar la experimentación con redes neuronales para determinar qué tipo de red pudiera ser la más adecuada para este tipo de modelos de predicción de desborde de las conversaciones entre un usuario y un chatbot.

Por otro lado, el análisis de tendencia del sentimiento de las conversaciones realizado en este proyecto permitirá, tener un insumo para generar reportes adicionales como: consultar el sentimiento general de los clientes en las interacciones con el chatbot durante el día, realizar evaluaciones implícitas del servicio en general, ya que muy pocos usuarios hacen uso de la encuesta y contrastar la calificación que realiza el usuario con la tendencia del sentimiento a lo largo de la conversación.

7 REFERENCIAS

Adamopoulou, E., & Moussiades, L. (2020). Chatbots: History, technology, and applications. *Machine Learning with Applications*, *2*, 1–18. https://doi.org/10.1016/j.mlwa.2020.100006

Akhtar, M., Neidhardt, J., & Werthner, H. (2019). The potential of chatbots: Analysis of chatbot conversations. *Proceedings - 21st IEEE Conference on Business Informatics, CBI 2019*, *1*, 397–404. https://doi.org/10.1109/CBI.2019.00052

Almansor, E.H., Hussain, F.K. & Hussain, O.K. Supervised ensemble sentiment-based framework to measure chatbot quality of services. Computing 103, 491–507 (2021). https://doi.org/10.1007/s00607-020-00863-0

ANDI. (2021). *El 2020 fue el año de la aceleración de la transformación digital en Colombia: ANDI*. https://www.andi.com.co/Home/Noticia/15881-el-2020-fue-el-ano-de-la-aceleracion-de

Ashktorab, Z., Jain, M., Vera Liao, Q., & Weisz, J. D. (2019). Resilient chatbots: Repair strategy preferences for conversational breakdowns. *Conference on Human Factors in Computing Systems - Proceedings*, 1–12. https://doi.org/10.1145/3290605.3300484

Avila J., Mayer M. & Quesada V. (2020). La inteligencia artificial y sus aplicaciones en medicina I: introducción antecedentes a la IA y robótica, Atención Primaria, 52(10), 778-784, https://doi.org/10.1016/j.aprim.2020.04.013.

Brandtzaeg, P. B., & Følstad, A. (2017). Why People Use Chatbots. Springer International Publishing AG 2017, 10673 LNCS, 377–392. https://doi.org/10.1007/978-3-319-70284-1_8

Duong, H. T., & Nguyen-Thi, T. A. (2021). A review: preprocessing techniques and data augmentation for sentiment analysis. *Computational Social Networks*, *8*(1), 1–16. https://doi.org/10.1186/s40649-020-00080-x

Forbes. (2020). *El 80% de las empresas en el mundo adelantaron su transformación digital por Covid-19*. Https://Forbes.Co/2020/09/27/Tecnologia/El-80-de-Las-Empresas-En-El-Mundo-Adelantaron-Su-Transformacion-Digital-Por-Covid-19/.

Hora, C., Embiruçu M. (2021). An approach combining a new weight initialization method and constructive algorithm to configure a single Feedforward Neural Network for multi-class classification, *Engineering Applications of Artificial Intelligence*, 106, 1-11. https://doi.org/10.1016/j.engappai.2021.104495

Hutto, C. J., & Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the 8th International Conference on Weblogs and Social Media, ICWSM 2014, 216–225.

IBM. (2020). *CRISP-DM Help Overview*. https://www.ibm.com/docs/es/spss-modeler/saas?topic=dm-crisp-help-overview

Katz, G., Ofek, N., & Shapira, B. (2015). ConSent: Context-based sentiment analysis. *Knowledge-Based Systems*, *84*, 162–178. https://doi.org/10.1016/j.knosys.2015.04.009

Khanna, A., Pandey, B., Vashishta, K., Kalia, K., Pradeepkumar, B., & Das, T. (2015). A Study of Today's A.I. through Chatbots and Rediscovery of Machine Intelligence. *International Journal of U- and e-Service, Science and Technology*, *8*(7), 277–284. https://doi.org/10.14257/ijunesst.2015.8.7.28

Kuramoto, I., Yoshikawa, Y., Baba, J., Kawabata, T., Ogawa, K., & Ishiguro, H. (2018). Conversational agents to suppress customer anger in text-based customer-support conversations. *HAI 2018 - Proceedings of the 6th International Conference on Human-Agent Interaction*, 114–121. https://doi.org/10.1145/3284432.3284457

Lam, S., Chen, C., Kim, K., Wilson, G., Crews, J. H., & Gerber, M. S. (2019). Optimizing customer-agent interactions with natural language processing and machine learning. *2019 Systems and Information Engineering Design Symposium, SIEDS 2019*, 0–5. https://doi.org/10.1109/SIEDS.2019.8735616

Li, C., Yeh, S. F., Chang, T. J., Tsai, M. H., Chen, K., & Chang, Y. J. (2020). A Conversation Analysis of Non-Progress and Coping Strategies with a Banking Task-Oriented Chatbot. *Conference on Human Factors in Computing Systems - Proceedings*, 1–12. https://doi.org/10.1145/3313831.3376209

Liao, Q. V., Mas-ud Hussain, M., Chandar, P., Davis, M., Khazaen, Y., Crasso, M. P., Wang, D., Muller, M., Shami, N. S., & Geyer, W. (2018). All Work and No Play? Conversations with a Question-and-Answer Chatbot in the Wild. *Industrial and Commercial Training*, *8*(8), 1–13. https://doi.org/10.1108/eb003561

Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Systematic reviews in sentiment analysis: a tertiary study. In *Artificial Intelligence Review* (Issue February). Springer Netherlands. https://doi.org/10.1007/s10462-021-09973-3

Loria, S., (2021) Textblob: Simplified text processing. [Online]. Available: https://textblob.readthedocs.io/en/dev

Mathur, S. & Lopez, D., (2019). A scaled-down neural conversational model for chatbots, *Concurrency and Computation: Practice and Experience*, 31, 1-10. https://doi.org/10.1002/cpe.4761

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, *5*(4), 1093–1113. https://doi.org/10.1016/j.asej.2014.04.011

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825-2830.

Prasad, B. S., & Akana, C. M. V. S. (2021). Polarity Sentiment-Based Intelligent Chat Bot for Judicious Customer Service Escalation. 2021 2nd Global Conference for Advancement in Technology, GCAT 2021, 1–6. https://doi.org/10.1109/GCAT52182.2021.9587592

Poria, S., Cambria, E., Bajpai, R., & Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, *37*, 98–125.

https://doi.org/10.1016/j.inffus.2017.02.003

Poria, S., Majumder, N., Mihalcea, R., & Hovy, E. (2019). Emotion Recognition in Conversation: Research Challenges, Datasets, and Recent Advances. *IEEE Access*, *7*, 100943–100953. https://doi.org/10.1109/ACCESS.2019.2929050

Rapp, A., Curti, L., & Boldi, A. (2021). The human side of human-chatbot interaction: A systematic literature review of ten years of research on text-based chatbots. *International Journal of Human Computer Studies*, *151*(January), 102630. https://doi.org/10.1016/j.ijhcs.2021.102630

Vijayaraghavan, V., Cooper, J. B., & Rian Leevinson, R. L. (2020). Algorithm Inspection for Chatbot Performance Evaluation. *Procedia Computer Science*, *171*(2019), 2267–2274. https://doi.org/10.1016/j.procs.2020.04.245

Wyeld, T., Jiranantanagorn, P., Shen, H., Liao, K., & Bednarz, T. (2021). Understanding the effects of real-time sentiment analysis and morale visualisation in backchannel systems: A case study. *International Journal of Human Computer Studies*, *145*(August 2020), 102524. https://doi.org/10.1016/j.ijhcs.2020.102524

Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on BiLSTM. *IEEE Access*, *7*, 51522–51532. https://doi.org/10.1109/ACCESS.2019.2909919