Maestría en

Ciencia de Datos y Máquinas de Aprendizaje con mención en Inteligencia Artificial

Trabajo previo a la obtención de título de Magister en Ciencia de Datos y Máquinas de Aprendizaje con mención en Inteligencia Artificial

AUTOR/ES:

TRAVEZ MONGE ANTONIO PAOLO

TRUJILLO MANCHENO BOLÍVAR PATRICIO

VELASCO BAQUE KEVIN STEVE

VILLACIS SARMIENTO HIPATIA NATASHA

TUTOR/ES:

Alejandro Cortés López
Iván Reyes Chacón.

TEMA: Análisis de Sentimientos en Redes Sociales Utilizando Técnicas de Aprendizaje Automático y Procesamiento de Lenguaje Natural

Quito - Ecuador

julio – 2025

REINVENTEMOS EL FUTURO

Certificación de autoría

Nosotros, Hipatia Natasha Villacis Sarmiento, Kevin Steve Velasco Baque, Antonio Paolo Trávez Monge, Bolívar Patricio Trujillo Mancheno**,** declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido presentado anteriormente para ningún grado o calificación profesional y que se ha consultado la bibliografía detallada.

Cedemos nuestros derechos de propiedad intelectual a la Universidad Internacional del Ecuador (UIDE), para que sea publicado y divulgado en internet, según lo establecido en la Ley de Propiedad Intelectual, su reglamento y demás disposiciones legales.

Firma del graduando
Hipatia Natasha Villacis Sarmiento

Firma del graduando
Kevin Steve Velasco Baque

Firma del graduando
Antonio Paolo Trávez Monge

Firma del graduando
Bolívar Patricio Trujillo Mancheno

Autorización de Derechos de Propiedad Intelectual

Nosotros, Hipatia Natasha Villacis Sarmiento, Kevin Steve Velasco Baque , Antonio Paolo Trávez Monge, Bolívar Patricio Trujillo Mancheno, en calidad de autores del trabajo de investigación titulado **Análisis de Sentimientos en Redes Sociales Utilizando Técnicas de Aprendizaje Automático y Procesamiento de Lenguaje Natural**, autorizamos a la Universidad Internacional del Ecuador (UIDE) para hacer uso de todos los contenidos que nos pertenecen o de parte de los que contiene esta obra, con fines estrictamente académicos o de investigación. Los derechos que como autores nos corresponden, lo establecido en los artículos 5, 6, 8, 19 y demás pertinentes de la Ley de Propiedad Intelectual y su Reglamento en Ecuador.

D. M. Quito, julio 2025.

Firma del graduando
Hipatia Natasha Villacis Sarmiento

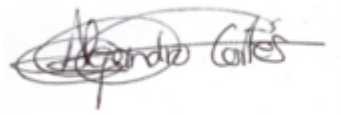
Firma del graduando
Kevin Steve Velasco Baque

Firma del graduando
Antonio Paolo Trávez Monge

Firma del graduando
Bolívar Patricio Trujillo Mancheno

Aprobación de dirección y coordinación del programa

Nosotros, **Ing. Alejandro Cortés y Mg. Iván Reyes ,** declaramos que: **Hipatia Natasha Villacis Sarmiento**, **Kevin Steve Velasco Baque**, **Antonio Paolo Trávez Monge**, **Bolívar Patricio Trujillo Mancheno**, son los autores exclusivos de la presente investigación y que ésta es original, auténtica y personal de ellos.



Ing. Alejandro Cortés
Director/a de la
Maestría en Maestría en Ciencia de Datos y Máquinas de Aprendizaje con mención en Inteligencia Artificial

---

Mg. Iván Reyes
Coordinador/a de la
Maestría en Maestría en Ciencia de Datos y Máquinas de Aprendizaje con mención en Inteligencia Artificial

DEDICATORIA

Este trabajo va dedicado a mis padres, Silvia y Bladimir, quienes, con su amor incondicional, sacrificio y sabiduría me han enseñado el valor del esfuerzo y la perseverancia. Gracias por siempre creer en mí y por brindarme todo su apoyo, incluso en momentos difíciles. A mi hijo Pablo David, el motor de mi vida, quien con su sonrisa y su amor me ha dado fuerza para seguir adelante, este logro también es tuyo, porque cada paso que doy está inspirado en ti y en el futuro que quiero construir para nosotros. A mis hermanos, Adriana e Isaac, por su apoyo, gracias por estar siempre allí, por comprenderme y por motivarme siempre que pueden. A cada uno de ustedes, por ser mi fuente de energía, mi aliento y mi razón para seguir adelante. Este logro no sería posible sin su amor y apoyo.

Natasha Villacís

A mi familia, cuya presencia ha sido siempre la raíz invisible de cada uno de mis logros.
A mis padres, por haber sembrado en mí la curiosidad como virtud y la disciplina como camino.
A Amy, por ser faro en los días nublados, impulso en los momentos de duda, y alegría en el trayecto. Su compañía ha sido un recordatorio constante de que el conocimiento también se construye con amor.
A mi tía Goita y a mi abuelita Mamá Inés que, aunque ya no estén conmigo, siempre desearon lo mejor para mí y siguen presentes en cada paso que doy.

Antonio Trávez

Este trabajo va dedicado a mi amada esposa Katherine y a mis padres Mónica y Washington, gracias por su extraordinario apoyo, son una motivación fundamental en mi vida para seguir siendo cada día mejor. Agradezco a mi magnífico instructor, Jehová Dios, que le da propósito a mi vida y permanentemente me brinda la oportunidad de aprender acerca de su maravillosa creación. “Porque contigo está la fuente de la vida; por luz de ti podemos ver luz.” (Salmos 36:9).

Bolívar Trujillo

A mi familia, aunque la distancia nos separe y las tormentas parezcan infinitas, siempre hay una luz que guía el camino. Sé que, de haberlo sabido, me habrían apoyado sin dudar. Por eso, esto va por ustedes.

A Dayana y a mi madre, que guardaron el secreto y estuvieron a mi lado en cada paso haciéndolo más firme cuando más tambaleante se convertía, ustedes han sido mi faro constante, mi refugio en la oscuridad. Gracias por ser mi luz cuando más lo necesitaba

Kevin Velasco Baque

AGRADECIMIENTOS

En primer lugar, queremos expresar nuestro más sincero agradecimiento a todas las personas que han hecho posible la culminación de este trabajo de fin de máster. Este proyecto no solo refleja nuestro esfuerzo individual, sino también el apoyo de aquellos que han estado a nuestro lado en cada etapa del proceso.

A nuestros profesores y mentores, por su constante guía y apoyo, por brindarnos las herramientas necesarias para superar los desafíos y por inspirarnos a seguir explorando el vasto mundo de la Ciencia de Datos y la Inteligencia Artificial.

A nuestras familias, por su amor, paciencia y comprensión, por ser un pilar inquebrantable en momentos de estrés y por brindarnos el espacio y la confianza para dedicarnos plenamente a este proyecto.

Finalmente, a nosotros mismos, por la perseverancia, el trabajo en equipo y el esfuerzo constante para lograr esta meta.

RESUMEN

El objetivo principal de este trabajo es la creación de un modelo de aprendizaje profundo para la categorización de emociones, mediante el uso de Redes Neuronales Recurrentes (RNN) con LSTM (Long Short-Term Memory), implica tanto los elementos técnicos de la puesta en marcha del modelo como los métodos de interpretación, con la finalidad de proporcionar un instrumento que no solo sea exacto, sino también entendible.

Se crean modelos de Deep Learning bidireccionales, concretamente MLP (Multilayer Perceptron) y LSTM, que resultan adecuados para manejar datos secuenciales e identificar dependencias a largo plazo en los textos de las redes sociales. El uso de Transfer Learning y Embeddings previamente entrenados tiene un rol crucial en la optimización de la exactitud y eficacia del modelo, dado que facilita la implementación del conocimiento obtenido en modelos entrenados sobre grandes cantidades de datos generales, adaptándolo luego a las particularidades específicas del conjunto de datos de redes sociales. Este método de transferencia acelera el proceso de entrenamiento, potencia la habilidad predictiva del modelo y facilita la obtención de resultados exactos con menos datos categorizados, un aspecto esencial en el ámbito del análisis de emociones en plataformas sociales. Se aplica el método de interpretabilidad a través de la utilización de SHAP para elucidar las decisiones del modelo de forma precisa y clara. Esta metodología facilita la visualización del efecto de cada palabra o elemento del texto en la predicción final, otorgando un entendimiento detallado de cómo el modelo mide y categoriza las emociones.

Palabras Clave: Deep Learning, SHAP, embeddings, Transfer Learning

ABSTRACT

The primary goal of this work is the development of a deep learning model for emotion classification using Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) units. This project addresses both the technical implementation of the model and the methods for interpretability, aiming to deliver a tool that is not only accurate but also transparent and interpretable.

Bidirectional deep learning architectures are implemented, specifically Multilayer Perceptrons (MLP) and LSTM networks, which are well-suited for processing sequential data and capturing long-term dependencies in social media text. The integration of Transfer Learning and pre-trained word embeddings plays a pivotal role in optimizing the model's accuracy and efficiency. By leveraging knowledge acquired from large-scale, general-purpose corpora, the model is better adapted to the specific linguistic and contextual characteristics of social media datasets. This transfer learning approach accelerates training, enhances predictive performance, and enables the model to achieve high accuracy with a relatively small amount of labeled data—an essential factor in emotion analysis on social platforms.

For model interpretability, SHAP (SHapley Additive exPlanations) is applied to provide clear and precise insights into the model's decision-making process. This method enables the visualization of the impact of individual words or textual elements on the model's predictions, offering a detailed understanding of how emotional categories are inferred from user-generated content.

Keywords: Deep Learning, SHAP, embeddings, transfer learning

TABLA DE CONTENIDOS

LISTA DE TABLAS

LISTA DE FIGURAS

CAPITULO 1

1. INTRODUCCIÓN

1.1. Definición del Proyecto

La utilización masiva de las redes sociales ha transformado el modo en que los individuos se comunican, comparten información y manifiestan sus puntos de vista. Plataformas como Facebook, Instagram y Twitter han simplificado la generación de grandes cantidades de información textual y multimodal, generando así nuevas posibilidades para el estudio de la opinión pública en distintos escenarios. En este contexto, el estudio de las emociones se ha establecido como un instrumento esencial en el ámbito de la ciencia de datos, posibilitando obtener datos útiles acerca de la visión de los usuarios con relación a asuntos de interés.

El estudio de emociones, o la extracción de puntos de vista, es un método que categoriza textos en grupos como positivos, negativos o neutrales, a través de algoritmos de aprendizaje automático y procesamiento de lenguaje natural (PLN). Su uso es extenso, incluyendo desde el cálculo de la satisfacción del cliente en compañías hasta la valoración del efecto de políticas del gobierno y el estudio de tendencias en el mercado. Al contrario de lo que se pudiera pensar, las emociones humanas no son obstáculos para la inteligencia artificial, más bien, constituyen una oportunidad para que los sistemas puedan adaptarse adecuadamente al usuario tomando un factor fundamental de la conducta humana, las emociones, esto es principalmente relevante en campos como la educación, la salud emocional y atención al cliente (Picard, 1997).

El propósito principal de esta investigación es implementar un algoritmo de machine learning que permita clasificar emociones humanas correctamente, tomando como referencia cadenas de texto de un set de datos que se encuentra en el repositorio de Kaggle, el campo de texto consiste en comentarios de la red social Twitter en inglés (Nelgiriyewithana, 2023). La

implementación del algoritmo contemplará la recolección de información del repositorio, el tratamiento previo de los textos, incluyendo un análisis exploratorio de datos, la optimización de las variables de texto y la aplicación de modelos de aprendizaje automático para categorizar las emociones manifestadas.

1.2. Justificación e importancia del trabajo de investigación

El estudio de las emociones en las redes sociales ha surgido como un instrumento esencial para entender la percepción y conducta de los usuarios en una diversidad de situaciones. En los sectores de la educación, financiero, saludo, entre otros, las empresas afrontan el desafío permanente de robustecer su comunicación y potenciar las relaciones con sus clientes. Es fundamental entender los puntos de vista y sentimientos de las personas y la comunidad en general para poder ofrecer productos y servicios que respondan a necesidades específicas y precisamente tomen en cuenta un factor fundamental de la naturaleza humana: las emociones.

El presente trabajo ofrece una oportunidad única para reconocer y examinar las emociones expresadas en las publicaciones en Twitter, la detección de emociones complejas es de especial importancia en los sistemas de recomendación y la atención automatizada al cliente, esto permite a las empresas y organizaciones la utilización eficiente de recursos al reducir gastos y mejorar ingresos, sin embargo, para una clasificación correcta de las emociones en texto se requiere la aplicación de enfoques semánticos y contextuales, además de una aproximación técnica y computacional apropiadas (Cambria y otros, 2012).

Este análisis, mediante la aplicación de métodos sofisticados de aprendizaje automático y procesamiento de lenguaje natural, facilita la automatización del manejo de grandes cantidades de información textual, permitiendo una valoración exacta, dinámica y en tiempo real de las interacciones en las plataformas sociales. Además, la importancia práctica de este proyecto se

basa en su habilidad para proporcionar instrumentos y técnicas que las entidades educativas pueden aplicar para optimizar sus tácticas de comunicación y administrar de forma más eficaz su reputación en internet. Este tipo de evaluaciones no solo beneficia a la comunidad en sí, sino que presenta el potencial de fomentar el progreso de los métodos de recolección de opiniones y estudio de datos en el ámbito educativo, generando nuevas visiones para el incremento de la calidad educativa y la toma de decisiones basadas en información.

1.3. Alcance

El objetivo de este estudio es desarrollar un proyecto que pueda examinar y categorizar emociones manifestadas en publicaciones de Twitter redactadas en inglés, esto se logró mediante la aplicación de métodos sofisticados de procesamiento de lenguaje natural (PLN) y algoritmos de aprendizaje automático, se detectaron patrones emocionales en los mensajes que los usuarios compartieron, lo que permitió adquirir un entendimiento más detallado de los sentimientos manifestados.

La recolección de datos se llevó a cabo utilizando un conjunto de datos de la plataforma Kaggle que se encuentra a libre disposición de los usuarios con fines educativos y de investigación, este conjunto de datos denominado “Emotions – Dataset” contiene registros de las opiniones de usuarios en Twitter y adicionalmente existe un campo donde se clasifica cada comentario con una emoción específica en inglés: sadness, joy, love, anger, fear, surprise (Nelgiriyewithana, 2023).

En la gestión de los datos, se utilizaron varias técnicas como el análisis exploratorio de datos, la limpieza de texto, la eliminación de “stop words” y la tokenización, con la finalidad de optimizar la calidad de las entradas que serían analizadas por los modelos. La categorización de emociones abarcó categorías concretas como la tristeza, alegría, amor, ira, miedo y sorpresa y se

llevó a cabo a través de modelos neuronales como el Perceptrón Multicapa (MLP) y las Redes Neuronales Recurrentes (RNN) con capas bidireccionales LSTM. Además, se implementó la utilización de embeddings previamente capacitados y métodos de aprendizaje por transferencia para potenciar la habilidad de generalizar los modelos.

Igualmente, se incorporaron instrumentos de interpretación como SHAP (Explanaciones Aditivas de Shapley), lo que facilitó una mayor claridad en las decisiones adoptadas por el sistema. La comparación entre modelos posibilitó valorar su desempeño a través de indicadores estándar como la precisión, recall y F1-score.

Los hallazgos de este estudio proporcionan una perspectiva organizada de las emociones en el diálogo digital relacionado con instituciones educativas, lo cual puede ser beneficioso para estrategias de comunicación institucional e investigaciones de percepción pública en contextos educativos.

1.4. Objetivos

1.4.1. Objetivo general

Analizar las emociones expresadas en tuits en ingles en el dataset “Emotions” ubicado en el repositorio en línea de Kaggle, utilizando técnicas de procesamiento de lenguaje natural y aprendizaje automático, con el fin de categorizar las emociones predominantes en las publicaciones de los usuarios.

1.4.2. Objetivo especificos

Aplicar técnicas de procesamiento de lenguaje natural (PLN), como limpieza de texto, análisis exploratorio de datos y tokenización, para preparar los tuits para su análisis y clasificación en categorías emocionales como alegría, tristeza, enojo, miedo y sorpresa.

Evaluar y clasificar las emociones presentes en los tuits recopilados utilizando modelos de aprendizaje automático, como Perceptrón Multicapa (MLP) y Redes Neuronales Recurrentes (RNN) con LSTM bidireccional, y analizar el rendimiento de estos modelos mediante métricas de clasificación estándar como precisión, recall y F1-score.

CAPITULO 2:

2. REVISIÓN DE LITERATURA

2.1. Estado del Arte

Muñiz investiga la aplicación del Análisis de Sentimientos mediante el Deep Learning (DL), resaltando la utilización de Redes Neuronales Profundas (DNN) en labores como la categorización de polaridad en tweets y comentarios. Se toman en cuenta tres perspectivas fundamentales: secuales, recursivos e híbridos. Las Redes Feed-Forward, Convolucionales (CNN) y Recurrentes (RNN) resultan fundamentales para manejar secuencias de datos y capturar dependencias contextuales, destacando las LSTM por su eficacia en el aprendizaje de dependencias a largo plazo. Los modelos híbridos fusionan las ventajas de las redes secuenciales y recursivas, en cambio, las Redes Tensoriales Recursivas (RsNN) utilizan árboles sintácticos para incrementar la exactitud en la categorización de emociones. Además, los enfoques centrados en la atención fortalecen la interpretación semántica, optimizando los resultados en comparación con los métodos convencionales. Los experimentos indican que los modelos DL sobrepasan notablemente a los procedimientos tradicionales en la categorización de emociones, logrando una mayor exactitud en diferentes bases de datos (Muñiz, 2018).

En 2020, Calvo creó un sistema inteligente que puede examinar emociones en las publicaciones en español, utilizando métodos de minería de opiniones y algoritmos de aprendizaje automático. La meta es determinar la polaridad de los mensajes (positiva, negativa o neutra) a través de un procedimiento organizado que comprende la recopilación de información a través de la API de Twitter, el preprocesamiento textual (purificación, Tokenización, lematización) y la clasificación subsiguiente utilizando modelos de aprendizaje automático. Este

sistema posibilita el análisis en tiempo real de la percepción pública y se utiliza en campos como la política, el marketing, la comunicación institucional y el análisis social (Calvo, 2020).

Para realizar el análisis, se empleó Python, bibliotecas como NLTK y herramientas como RapidMiner para entrenar y contrastar tres algoritmos. Naive Bayes, Logística de Regresión y Máquinas de Apoyo Vectorial (SVM). Los hallazgos revelaron que SVM fue el modelo con el mejor rendimiento, con una precisión del 86.58%, y le siguió Regresión Logística. El estudio evidencia la factibilidad de emplear métodos de inteligencia artificial para analizar la opinión pública en las redes sociales con elevados grados de exactitud y fiabilidad (Calvo, 2020).

Kumar y colaboradores destacan la importancia del aprendizaje profundo como un método esencial para optimizar el estudio de las emociones en las redes sociales. Se investigan varias arquitecturas de redes neuronales profundas, tales como las redes neuronales convolucionales (CNN) y las redes neuronales recurrentes (RNN), que facilitan la captura de patrones complejos en la información desestructurada de las redes sociales. Mediante un método de aprendizaje jerárquico, los modelos profundos adquieren características complejas a partir de información textual, superando las restricciones de métodos convencionales que se basan en la obtención manual de atributos. Adicionalmente, la inclusión de un diccionario de emociones a medida optimiza la categorización de las emociones, lo que facilita una mayor exactitud en la comprensión de emociones complejas. Los hallazgos experimentales señalan que, al calibrarse adecuadamente, los modelos de deep learning proporcionan una precisión superior en comparación con otras técnicas, lo que evidencia su efectividad para gestionar grandes cantidades de datos y el lenguaje informal propio de las plataformas sociales (Kumar y otros, 2024).

Lazo y Rodas presentaron su investigación titulada "Análisis de sentimiento en las redes sociales de las universidades de Cuenca", en la que se propone el desarrollo de un sistema de análisis de emociones para clasificar las opiniones en las redes sociales, tomando como referencia la Universidad Central del Ecuador. El objetivo era establecer la percepción de los estudiantes sobre su institución mediante la clasificación automática de opiniones como positivas, negativas o neutrales. Para alcanzar este objetivo, se utilizaron métodos de scraping para recopilar datos de las redes sociales, técnicas de procesamiento de lenguaje natural (PLN) para la purificación y cambio de los textos, y algoritmos de aprendizaje automático como Naive Bayes y Máquinas de Apoyo Vector para la categorización (Lazo & Rodas, 2024).

Los hallazgos evidenciaron que el modelo sugerido puede categorizar comentarios con una precisión adecuada, lo que permite a la universidad obtener una perspectiva más precisa de las opiniones de los estudiantes. Además, se creó una interfaz visual que simplifica la carga de datos y la representación de los resultados. Esta propuesta aporta al robustecimiento de la toma de decisiones en el contexto institucional mediante el estudio de datos en tiempo real (Lazo & Rodas, 2024).

El artículo examina métodos anteriores en el estudio de las emociones, como la aplicación de TF-IDF y SVM, que se encuentran con restricciones al gestionar grandes cantidades de datos no estructurados y lenguaje informal. Por otro lado, el aprendizaje profundo (deep learning), utilizando arquitecturas como CNN y RNN, ha probado ser más eficaz para identificar patrones complejos. La tesis titulada "Análisis de Sentimientos en Redes Sociales usando Deep Learning" es revolucionaria al fusionar el aprendizaje profundo con un diccionario de emociones a medida, potenciando la exactitud y la capacidad de adaptación. Esta idea

trasciende las restricciones de los procedimientos convencionales, mejorando el análisis en plataformas sociales.

2.2.Marco Teórico

2.2.1. Fundamentos de Análisis de sentimientos

2.2.1.1. Definición: Análisis de sentimiento y Minería de opiniones

Análisis de Sentimientos. El análisis de sentimientos, también llamado minería de opiniones (opinion mining), es una disciplina del procesamiento de lenguaje natural (PLN) y la inteligencia artificial, cuyo objetivo es reconocer, extraer y categorizar de manera automática la información subjetiva contenida en textos digitales, tales como puntos de vista, emociones y actitudes manifestadas por los usuarios (D'Andrea y otros, 2015).

Su meta principal es establecer la emoción que subyace a una serie de palabras, categorizando el contenido en términos positivos, negativos o neutros. Este procedimiento se utiliza en grandes cantidades de información textual obtenida de fuentes como las redes sociales, críticas de productos, sondeos, correos electrónicos y más, facilitando a las entidades una mejor comprensión de la percepción pública respecto a marcas, productos o servicios públicos. El análisis de sentimiento se basa en técnicas de aprendizaje automático, modelos estadísticos y diccionarios léxicos, y puede operar en diferentes niveles de granularidad (D'Andrea y otros, 2015):

- Abroad: Establece la percepción global de un documento o serie de textos.
- Por frase: Examina la emoción de expresiones personales.
- Por característica: Reconoce la emoción vinculada a atributos o particularidades dentro del texto (como la calidad de un producto, el servicio al cliente, etc.).

Minería de Opiniones. La minería de opiniones es una rama del análisis de sentimientos que se centra en obtener datos más exhaustivos sobre las perspectivas manifestadas respecto a elementos o características específicas de un producto, servicio o asunto. Aunque el análisis de emociones puede determinar si un texto es positivo, negativo o neutral, la minería de opiniones desglosa el texto para determinar sobre qué elemento particular se está hablando y cuál es la valoración vinculada a dicho elemento (Liu, 2012).

Por ejemplo, en un análisis de un móvil, la recolección de comentarios puede detectar que el usuario manifiesta una valoración favorable sobre la cámara, pero una valoración negativa acerca de la autonomía de la batería (Liu, 2012).

Características principales:

- Utiliza técnicas avanzadas de PLN y machine learning.
- Permite analizar grandes volúmenes de datos de manera automatizada.
- Ofrece información procesable para la toma de decisiones empresariales, mejora de productos y estrategias de marketing.
- Puede adaptarse a distintos idiomas y contextos

2.2.1.2.Clasificación de sentimientos

La categorización de emociones es un trabajo esencial en el estudio de las emociones, cuyo propósito principal es establecer la polaridad de un texto, o sea, si la perspectiva manifestada es positiva, negativa o neutral (Zachar, 2021).

Clasificación básica de sentimientos:

- *Sentimiento positivo:* El documento manifiesta una perspectiva positiva, satisfacción o aprobación respecto a un producto, servicio, marca o asunto.
  Ejemplo: "Me impresionó el servicio que obtuve."

- *Sentimiento negativo*: El texto manifiesta descontento, desacuerdo o crítica. Ejemplo: "El producto llegó con fallas y no opera."
- *Sentimiento neutral:* El texto no manifiesta una emoción evidente ni una perspectiva positiva o negativa; generalmente es informativo o descriptivo sin contenido emocional. "El paquete se presentó ayer."

Esta categorización se denomina análisis de polaridad y es el método más habitual y simple de evaluación de emociones. Algunos sistemas sofisticados pueden perfeccionar esta polaridad en clasificaciones más precisas, como "extremadamente positivo" o "extremadamente negativo", o asignar calificaciones numéricas para determinar la intensidad de la sensación (Zachar, 2021).

La categorización positiva, negativa y neutral posibilita que empresas e investigadores entiendan con rapidez la visión general de un asunto y tomen decisiones fundamentadas en grandes cantidades de información textual.

2.2.1.3.Análisis de emociones vs Análisis de polaridad

El siguiente ejemplo demuestra que el estudio de las emociones trasciende la mera polaridad, facilitando la identificación de emociones específicas y sutilezas que el análisis de polaridad no identifica. Las dos perspectivas pueden fusionarse para lograr una percepción más integral del contenido emocional de los textos (Nandwani & Verma, 2021).

Tabla 1

Emociones vs Polaridad

Característica	Análisis de Emociones	Análisis de Polaridad
Definición	Identifica emociones específicas expresadas en el texto (alegría, tristeza, ira, etc.)	Determina si el sentimiento general es positivo, negativo o neutro

Granularidad	Alta: clasifica en múltiples categorías emocionales	Baja: clasifica en tres categorías principales (positivo, negativo, neutro)
Ejemplo de salida	Alegría, tristeza, miedo, enojo, sorpresa, etc.	Positivo, negativo, neutro
Aplicaciones	Análisis profundo de estados emocionales, marketing emocional, salud mental	Monitoreo de reputación, satisfacción del cliente, encuestas de opinión
Complejidad	Mayor, requiere modelos y léxicos emocionales más avanzados	Menor, generalmente más sencilla de implementar
Relación	Puede complementar el análisis de polaridad para mayor precisión	Es la base del análisis de sentimientos, pero menos detallado

Nota: Tabla de comparación de emociones con la polaridad.

2.2.2. Recolección y almacenamiento de datos desde Redes Sociales

2.2.2.1.Estructuración de datos textuales

Es crucial la organización de datos textuales para convertir la información no estructurada (como opiniones, comentarios o reseñas) en formatos ordenados que simplifiquen su almacenamiento, consulta y análisis, particularmente en labores de análisis de emociones y extracción de opiniones, especialmente en labores de análisis de sentimientos y extracción de opiniones (Syaamantak y otros, 2019).

Aplicación práctica en análisis de sentimiento

- Los textos y sus características (como el sentimiento asignado, la fecha, el usuario, la fuente) pueden ser guardados en JSON para preservar la estructura integral y los metadatos relacionados.
- Para análisis estadísticos o representación visual, generalmente se exportan los datos a CSV, lo que simplifica su manejo en herramientas como R o Python.
- En el manejo de grandes cantidades de información con diversas relaciones

(usuarios, publicaciones, emociones), se recurre a bases de datos relacionales que facilitan consultas eficaces y una segmentación sofisticada (Syaamantak y otros, 2019).

Tabla 2

Formatos Base de Datos

Formato	Descripción	Ventajas	Uso Común
JSON	Formato de texto ligero y flexible que representa datos estructurados en forma de objetos y arrays. Permite almacenar texto junto con metadatos (fecha, usuario, sentimiento, etc.)	Muy flexible, fácil de integrar con APIs y sistemas web, soporta estructuras anidadas complejas	Almacenamiento de datos extraídos de redes sociales, APIs y sistemas de scraping; intercambio de datos entre aplicaciones
CSV	Archivo de texto plano con valores separados por comas, donde cada fila representa un registro y cada columna un atributo	Simple, ampliamente soportado, fácil de abrir y manipular en hojas de cálculo y programas estadísticos	Guardar resultados de análisis, exportar datos para procesamiento en R, Python u otros entornos de análisis
Formatos relacionales (bases de datos SQL)	Datos almacenados en tablas con filas y columnas, que permiten consultas complejas y relaciones entre tablas	Estructura organizada, soporte para consultas avanzadas, integridad y escalabilidad	Almacenamiento y gestión de grandes volúmenes de datos estructurados.

Nota: Tabla de definiciones JSON, CSV, BASES DE DATOS SQL: descripción, ventajas y uso común.

2.2.2.2.Almacenamiento en bases de datos

La fusión de ambas clases de bases de datos, relacionales para datos estructurados y NoSQL para datos versátiles, es habitual en arquitecturas contemporáneas de ciencia de datos, favoreciendo la integración, procesamiento y análisis eficaz de datos diversos (Kyaw, 20215).

La elección entre bases de datos relacionales y NoSQL debe estar en consonancia con las necesidades del proyecto de análisis de sentimientos, teniendo en cuenta factores como el tipo de datos, la escalabilidad requerida y las operaciones previstas para garantizar un almacenamiento ideal y un procesamiento eficaz.

Tabla 3

Características Bases de Datos

Característica	Bases de datos relacionales (SQL)	Bases de datos NoSQL
Modelo de datos	Tablas con filas y columnas, esquema rígido	Documentos, clave-valor, columnas, grafos, esquema flexible
Tipo de datos	Datos estructurados y consistentes	Datos no estructurados o semiestructurados
Escalabilidad	Vertical (potenciar un solo servidor)	Horizontal (distribución en varios servidores)
Flexibilidad de esquema	Baja, requiere esquema definido previamente	Alta, esquema dinámico y adaptable
Lenguaje de consulta	SQL	Varía según tipo (consultas JSON, CQL, etc.)
Ejemplos	MySQL, PostgreSQL, Oracle	MongoDB, Redis, Cassandra, Neo4j

Nota: Comparación de base de datos relacionales y no relacionales.

2.2.3. Procesamiento de Lenguaje Natural (PLN)

2.2.3.1.Introducción al PLN y su papel en el análisis de texto

El Procesamiento de Lenguaje Natural (PLN) es una disciplina de la inteligencia artificial dedicada a la interacción entre las computadoras y el idioma humano, con el objetivo de que las máquinas comprendan, interpreten, generen y reaccionen en lenguaje natural, al igual que nosotros, los seres humanos (IMB, 2024).

El PLN fusiona saberes de lingüística computacional, informática, inteligencia artificial y aprendizaje automático con el fin de procesar y examinar textos y voz en idiomas naturales como el español, inglés, chino, y más.

Facilita que las máquinas identifiquen, entiendan y produzcan texto y voz, simplificando la interacción entre humanos y máquinas en aplicaciones diarias como asistentes virtuales (Siri, Alexa), sistemas de traducción, chatbots, motores de búsqueda y análisis de emociones (IMB, 2024).

Análisis de Texto

- El PLN convierte textos sin estructura en datos estructurados a través de métodos como la tokenización, la lematización, la supresión de palabras irrelevantes (stop words) y la limpieza de texto, preparando la información para futuros análisis.
- En el estudio de emociones y la extracción de puntos de vista, el PLN es esencial para reconocer emociones, posturas y polaridad en textos que provienen de redes sociales, críticas o sondeos, lo que facilita la comprensión de la percepción y punto de vista de los usuarios.
- Emplea modelos estadísticos, aprendizaje automático y redes neuronales para descifrar el sentido, contexto y sutilezas del lenguaje, incrementando la exactitud en labores como la categorización de emociones, identificación de temas y producción de respuestas (IMB, 2024).

2.2.3.2.Limpieza de texto: eliminación de ruido stopwords, emojis

La purificación de texto es un paso esencial en el procesamiento de lenguaje natural (PLN), que implica preparar y depurar los datos textuales con el fin de optimizar la calidad y

exactitud de los análisis subsiguientes, tales como el estudio de emociones o la extracción de puntos de vista (Kaur & Kaur, 2018).

Componentes principales de la limpieza de texto

Eliminación de ruido: Hace referencia a la separación de elementos no relevantes o que pueden obstruir en el análisis, tales como (Kaur & Kaur, 2018):

- Etiquetas HTML o código residual.
- URLs y enlaces.
- Números o caracteres especiales innecesarios.
- Saltos de línea y espacios en blanco excesivos.

Eliminar este ruido sirve para que el modelo se centre en la información relevante del texto.

Eliminación de stopwords: Las stopwords son palabras muy recurrentes en un idioma que contribuyen poco significado por sí solas, como artículos, preposiciones y conjunciones (ejemplo: “el”, “de”, “y”, “en”) (Kaur & Kaur, 2018).

Su exclusión disminuye el ruido y mejora la eficacia del análisis, ya que estas palabras no suelen influir en la polaridad o el contenido semántico. Las listas de stopwords son adaptables según el contexto o idioma.

Eliminación o manejo de emojis: Los emojis son símbolos gráficos que enuncian emociones o ideas y son muy comunes en textos informales, principalmente en redes sociales. Basándose en el objetivo, los emojis pueden (Kaur & Kaur, 2018):

- Ser eliminados para simplificar el texto.
- Ser convertidos a texto descriptivo (por ejemplo, ☺ a “sonrisa”) para preservar su carga emocional.

El uso adecuado de emojis es importante para captar matices emocionales en el análisis de sentimientos.

Beneficios de la limpieza de texto

- Mejora la eficacia de los datos para modelos de PLN y machine learning.
- Disminuye el ruido que puede influir la precisión del análisis.
- Facilita la extracción de información relevante y patrones significativos.
- Aligera los procesos computacionales al disminuir el volumen de datos innecesarios.

2.2.3.3. Tokenización, lematización y stemming

La Tokenización es la etapa inicial y esencial en el procesamiento del lenguaje natural (PLN), que implica segmentar un texto en unidades más reducidas conocidas como tokens, que pueden ser palabras, expresiones o incluso caracteres únicos (Trujillo, 2018).

Divide el texto en secciones comprensibles para que los algoritmos sean capaces de entender la estructura y el contexto del lenguaje. Los tokens de palabras generalmente se separan por espacios vacíos, mientras que los tokens de frases se separan por puntos u otros signos de puntuación (Trujillo, 2018).

Puede aplicarse a diferentes niveles de granularidad:

- **Tokenización por palabras:** fragmenta el texto en palabras individuales.
- **Tokenización por oraciones:** divide el texto en oraciones completas.
- **Tokenización por caracteres o subpalabras:** ventajoso en modelos avanzados para manejar vocabularios extensos o idiomas con morfología compleja.

Importancia en PLN

- Facilita el análisis y el proceso posterior del texto, como la extracción de características, análisis sintáctico y semántico, y clasificación.

- Admite que las máquinas descifren el lenguaje humano al convertir datos no estructurados en estructuras que los modelos computacionales pueden procesar.
- Es un requisito preciso para tareas como análisis de sentimientos, traducción automática, reconocimiento de entidades y generación de texto (Trujillo, 2018).

Ejemplo de Tokenización por palabras

Para la frase: "El procesamiento de lenguaje natural es fascinante."

La Tokenización por palabras produce:

["El", "procesamiento", "de", "lenguaje", "natural", "es", "fascinante", "."].

Herramientas comunes para Tokenización

- **NLTK**: biblioteca común en Python para Tokenización de palabras y frases.
- **SpaCy:** opción rápida y eficiente, con soporte multilingüe.
- **Tokenizadores de modelos como BERT:** operan Tokenización consciente del contexto y subpalabras para mejorar la comprensión en modelos avanzados.

El proceso de lematización es un método de preprocesamiento en el procesamiento del lenguaje natural (PLN), que implica disminuir las palabras flexionadas o derivadas a su forma base o lema, que es la forma canónica que se muestra en el diccionario. La lematización emplea estudios morfológicos y gramaticales para determinar la forma adecuada de acuerdo con el contexto y la clasificación gramatical (componente del discurso) del término. Por ejemplo, se lematizan las palabras "corriendo", "corrió" y "corre" a "correr" (Trujillo, 2018).

Características principales de la lematización

- **Forma base o lema:** Proporciona la palabra en su versión de diccionario (infinitivo para verbos, singular masculino para adjetivos, singular para sustantivos).
- **Contexto y POS:** Considera la sección del discurso y el contexto para desambiguar palabras que pueden tener distintas significaciones dependiendo de su aplicación (como "ama" como sustantivo o verbo).
- **Precisión:** Genera palabras auténticas y precisas, incrementando la calidad del análisis en comparación con el stemming.
- **Complejidad:** Se trata de un procedimiento más pausado y arduo ya que demanda análisis morfológico y acceso a diccionarios o bases de datos de lenguaje.

El stemming es un método de preprocesado de texto en PLN que implica reducir las palabras a su raíz o forma fundamental conocida como stem, suprimiendo sufijos y prefijos derivados o flexionados. La meta es reunir las variantes morfológicas de una palabra bajo un mismo concepto para simplificar su análisis y procesamiento (Trujillo, 2018).

Características principales

- Es un procedimiento rápido y heurístico que secciona los extremos de las palabras para lograr una raíz compartida, aunque esta raíz no siempre sea una palabra válida en el habla.
- Permite normalizar el texto para que palabras relacionadas (por ejemplo, “running”, “runs”, “run”) se traten como equivalentes (“run”).
- Es muy utilizado en sistemas de recuperación de información, motores de búsqueda y clasificación de texto, donde se busca reducir la dimensionalidad y mejorar la

eficiencia.

- Contrario de la lematización, el stemming no considera el contexto ni la categoría gramatical, por lo que puede formar raíces incorrectas o poco precisas.

Tabla 4
Ejemplos Stemming

Palabras originales	Raíz (stem) obtenida
running, runs, runner	run
fishing, fished, fisher	fish
argue, argued, arguing, argument	argu

Nota: Palabras originales y raíces de las mismas.

Algoritmos comunes

- *Porter Stemmer:* Uno de los algoritmos más ocupados, que aplica reglas secuenciales para eliminar sufijos comunes en inglés.
- *Snowball Stemmer:* Versión superior del Porter, con soporte para múltiples idiomas.
- *Regex Stemmer:* Maneja expresiones regulares para definir patrones de sufijos a eliminar**.**

2.2.3.4.Vectorización: Bagof words, TF-IDF, Word Embeddings

La vectorización es un procedimiento esencial en el manejo del lenguaje natural (PLN), que convierte texto en representaciones numéricas (vectores) que las computadoras pueden comprender y manejar para actividades como el análisis de emociones, la clasificación o la traducción (Dauda & Umar, 2022).

1. Bag of Words (Bolsa de Palabras)

- Simboliza un texto como un conjunto (bolsa) de palabras sin considerar el orden ni la gramática (Dauda & Umar, 2022).
- Cada texto se transforma en un vector donde cada dimensión corresponde a una palabra del vocabulario, y el valor muestra la frecuencia de esa palabra en el texto.
- Es simple y fácil de implementar, pero no captura el contexto ni el significado semántico.
- Puede formar vectores muy dispersos y de alta dimensión si el vocabulario es grande.

2. TF-IDF (Term Frequency - Inverse Document Frequency)

- Perfecciona la ilustración de Bag of Words al evaluar la frecuencia de las palabras en un texto (TF) y su escasez en el conjunto total de documentos (IDF)
- Las palabras habituales en numerosos documentos tienen un peso reducido, mientras que las palabras más concretas y pertinentes adquieren mayor relevancia.
- Contribuye a resaltar palabras relevantes para distinguir textos y aumentar la exactitud en labores de categorización y análisis.
- Continúa siendo un modelo fundamentado en frecuencias, sin reconocer conexiones semánticas profundas • No capta relaciones semánticas profundas (Dauda & Umar, 2022).

3. Word Embeddings (Representaciones Vectoriales de Palabras)

- Figuran palabras en un espacio vectorial continuo y de baja dimensión, donde palabras con significados equivalentes están cerca unas de otras.
- Recolectan el contexto y la semántica de las palabras, prevaleciendo las limitaciones de modelos basados solo en frecuencia.

- Se obtienen por medio de modelos de aprendizaje automático como Word2Vec, FastText o GloVe, que aprenden relaciones semánticas a partir de grandes corpus de texto.
- Son vectores densos que admiten realizar operaciones matemáticas para medir similitud, analogías y relaciones semánticas (Dauda & Umar, 2022).

2.2.4. Aprendizaje automático

El aprendizaje automático es un campo de la inteligencia artificial que posibilita a las computadoras adquirir y perfeccionar de manera automática a partir de datos, sin estar programadas de manera explícita para funciones determinadas. En vez de acatar directrices estrictas, los sistemas de aprendizaje automático detectan patrones y vínculos en los datos con el fin de realizar pronósticos o tomar decisiones (Management Solutions, 2018).

Papel del aprendizaje automático en el procesamiento de lenguaje natural (PLN)

- El aprendizaje automático es importante para que las máquinas comprendan, descifren y formen lenguaje humano de forma efectiva.
- Permite elaborar modelos con la capacidad de analizar grandes volúmenes de texto, reconocer patrones semánticos y sintácticos, y cumplir con tareas complejas como análisis de sentimientos, traducción automática o generación de texto

2.2.4.1. Tipos de aprendizaje automático

El aprendizaje automático (machine learning) se divide principalmente en cuatro tipos, según la naturaleza de los datos y el tipo de supervisión en el entrenamiento (Blanquicett & Murillo, 2022):

Tabla 5

Tipos de aprendizaje automatizado

Tipo de aprendizaje	Datos requeridos	Objetivo principal	Ejemplos de uso
Supervisado	Datos etiquetados	Predecir etiquetas o valores	Clasificación de texto, predicción de ventas
No supervisado	Datos sin etiquetar	Encontrar patrones o grupos	Segmentación de clientes, detección de anomalías
Semisupervisado	Datos parcialmente etiquetados	Mejorar aprendizaje con pocos datos etiquetados	Reconocimiento facial, clasificación con pocos datos
Por refuerzo	Interacción con entorno	Aprender políticas óptimas mediante prueba y error	Juegos, robots autónomos

Nota: Descripción de los tipos de aprendizaje

2.2.4.2.Algoritmos comunes

Los algoritmos de aprendizaje automático son grupos de comandos que habilitan a las máquinas para detectar patrones en la información y realizar pronósticos o categorizaciones. A continuación, se detallan algunos de los algoritmos más frecuentemente empleados, en particular en el aprendizaje supervisado y no supervisado (Blanquicett & Murillo, 2022).

Algoritmos de aprendizaje supervisado

1. **Regresión lineal:** Simula la conexión lineal entre variables autónomas y una variable numérica constante. Es veloz y sencillo de entender, empleado para pronósticos como precios o ventas.
2. **Regresión logística:** Empleado para la clasificación binaria, modela la posibilidad de que un elemento sea de una clase. Uso en diagnóstico médico, identificación de

fraudes, entre otros.

3. **Árboles de decisión:** Organiza en un árbol que segmenta los datos de acuerdo con normas fundamentadas en características para anticipar resultados. Simple de entender y aplicar en la clasificación y la regresión.
4. **Bosque aleatorio (Random Forest):** Mixta varios árboles de decisión capacitados en distintos subconjuntos de datos para incrementar la exactitud y minimizar el sobreajuste. Utilizado en la identificación de imágenes y diagnóstico médico.
5. **Máquinas de vectores soporte (SVM):** Establece un hiperplano que distingue las clases incrementando el margen entre las mismas. Empleado en la categorización de textos, identificación facial e identificación de irregularidades.
6. **k-Vecinos más próximos (k-NN):** Incluye una instancia de acuerdo con la mayoría de sus vecinos más próximos en el espacio de características. Aplicado en la identificación de irregularidades y sugerencias.
7. **Regresor de refuerzo de gradiente (Gradient Boosting):** Modelo agrupado que fusiona varios modelos débiles para construir uno sólido, gestionando de manera eficiente no linealidades y multicolinealidad. Aplicado en proyecciones económicas y análisis complejos.

Algoritmos de aprendizaje no supervisado

1. **K-means clustering:** Recopila datos en k clusters basados en la proximidad, es útil para segmentación de clientes y análisis exploratorios.
2. **Análisis de componentes principales (PCA):** Disminuye la dimensionalidad de los datos sosteniendo la mayor varianza posible, facilitando la visualización y el procesamiento.

3. **Modelos de mezcla gaussiana:** Modelan la distribución de datos como una composición de varias distribuciones gaussianas para clustering más flexible.

2.2.5. Deep Learning para Análisis de Sentimientos

2.2.5.1. Fundamentos de redes neuronales profundas

Las redes neuronales profundas (DNN) son modelos informáticos basados en la estructura cerebral humana, concebidos para manejar información compleja a través de diversas capas de neuronas artificiales vinculadas. Estas redes se diferencian por su habilidad para adquirir patrones complejos partiendo de grandes cantidades de datos, lo que las convierte en imprescindibles en aplicaciones como la visión artificial, el procesamiento del lenguaje natural y los sistemas de recomendación (Jawad, 2023).

Componentes clave de una red neuronal profunda

1. Estructura en capas (Jawad, 2023):

- Capa de entrada: Recoge datos en bruto (ej. píxeles de imágenes o palabras de texto).
- Capas ocultas: Convierten los datos por medio de operaciones matemáticas. Una DNN típica posee decenas o cientos de dichas capas, al contrario de las redes neuronales tradicionales (2-3 capas).
- Capa de salida: Crea la predicción o clasificación final.

2. Neuronas y pesos:

Cada neurona maneja entradas mediante la aplicación de un peso (cifra que señala importancia) y una función de activación (por ejemplo, ReLU o sigmoide). Durante el entrenamiento, los pesos se modifican para reducir los errores (Jurafsky, 2025).

3. Funciones de activación:

Se incorpora no linealidad al sistema, lo que facilita la modelación de relaciones complejas. Incluyen ejemplos habituales (Jurafsky, 2025):

- ReLU (Unidad Lineal Rectificada): Ideal para capas ocultas.
- Softmax: Usada en capas de salida para clasificación multiclase

2.2.5.2. Arquitectura del Perceptrón Multicapa (MLP)

La arquitectura del Perceptrón Multicapa (MLP) se organiza en niveles interconectados que facilitan el modelado de relaciones no lineales en la información. El MLP tiene la capacidad de llevar a cabo tareas de clasificación y regresión, y puede emplearse en situaciones donde la correlación entre las entradas y las salidas no es lineal. El algoritmo de entrenamiento empleado por el MLP se fundamenta en la propagación retrospectiva (backpropagation), que modifica los pesos de los vínculos de la red con el fin de reducir el error de predicción entre las salidas generadas por la red y las salidas buscadas (Piccioni y otros, 2023).

Su diseño básico abarca elementos esenciales estructurados en un flujo secuencial de información:

Componentes estructurales principales

1. Capa de entrada
   - Simboliza las características o variables del conjunto de datos.
   - Cada neurona pertenece a una dimensión de entrada (ej. 10,000 términos en TF-IDF).
   - Función: Trasferir datos sin procesamiento a la primera capa oculta.
2. Capas ocultas
   - Cantidad configurable (típicamente 1-3 en aplicaciones prácticas).
   - Neuronas con funciones de activación no lineales (ReLU, sigmoide).

- Procesamiento a través de:
  - Suma ponderada: $z = \sum(\omega_i x_i) + b$
  - Transformación no lineal: $a = f(z)$ donde f es la función de activación

3. Capa de salida

- Configuración adecuada a la tarea específica:
- 1 neurona con sigmoide para clasificación binaria
- Variadas neuronas con softmax para clasificación multiclase
- Función lineal para regresión

2.2.5.3. Redes Neuronales Recurrentes (RNN) y LSTM Bidireccional

Las Redes Neuronales Recurrentes (RNN) son modelos creados para manejar datos secuenciales, en los que la respuesta en un momento no solo se basa en la entrada presente sino también en las entradas previas, debido a un estado encubierto que guarda datos anteriores. Esto las convierte en particularmente valiosas en labores como el procesamiento del lenguaje natural, el reconocimiento de voz y el estudio de series de tiempo (Domor y otros, 2024).

Redes Neuronales Recurrentes (RNN)

Funcionamiento básico: Las RNN procesan secuencias de manera gradual, renovando un estado encubierto que recoge datos significativos de entradas previas. La información se propaga en un círculo, lo que permite que la salida se ajuste al contexto anterior (Domor y otros, 2024).

Entrenamiento: Emplean la retro propagación a través del tiempo (BPTT), la cual amplía la retro propagación convencional para secuencias, añadiendo errores en cada etapa temporal para modificar los pesos comunes.

Limitaciones: Las RNN básicas enfrentan problemas para aprender dependencias a largo plazo debido al problema del gradiente que se extingue o se explota, lo que restringe su habilidad para memorizar información remota en la secuencia.

Redes LSTM (Long Short-Term Memory)

Las LSTM son una variante sofisticada de RNN creada para vencer las restricciones de las RNN convencionales. Incluyen "celdas de memoria" capaces de conservar datos significativos durante extensos lapsos y determinar qué información olvidar o recordar a través de puertas de entrada, salida y olvido. Facilitan la captura de dependencias a largo plazo en secuencias, incrementando notablemente el desempeño en labores con contextos amplios, como la traducción automática o el reconocimiento de voz (Jurafsky, 2025).

RNN Bidireccional (BRNN) y LSTM Bidireccional

Las RNN bidireccionales manejan la secuencia en dos vías: adelante (del pasado al futuro) y atrás (del futuro al pasado). Esto posibilita que el modelo considere tanto el entorno anterior como el subsiguiente para cada etapa de la secuencia. Dos estratos escondidos autónomos atraviesan la secuencia en direcciones contrarias, y sus salidas se fusionan para crear una representación más profunda y contextual (Graves, 2005).

LSTM Bidireccional:

Fusiona la capacidad de memoria a largo plazo de las LSTM con el procesamiento bidireccional, incrementando la exactitud en labores donde el contexto futuro tiene la misma relevancia que el pasado, como en el estudio de texto o el reconocimiento de voz (Graves, 2005).

Tabla 6

Comparación RNN / LSTM

Característica	RNN Simple	LSTM	BRNN / LSTM Bidireccional
Manejo de dependencias	Restringido a corto plazo	Captura dependencias largas	Captura contexto pasado y futuro
Arquitectura	Estado oculto simple	Celdas de memoria con puertas	Dos estados ocultos, adelante y atrás
Problemas comunes	Gradiente desvaneciente	Mitigación del gradiente	Mejor exactitud por contexto completo
Aplicaciones típicas	Series estacionales simples	Traducción, voz, texto	Afirmación avanzada, PLN, secuencias complejas

Nota: Características concretas sobre RNN/LSTM

2.2.5.4.Técnicas de regularización: Dropout, L2, Early Stopping

Las técnicas de regularización son procedimientos empleados para evitar el sobreajuste en modelos de aprendizaje automático, particularmente en redes neuronales, potenciando de esta manera su habilidad para extenderse a datos inéditos. A continuación, se detallan tres métodos fundamentales: Dropout, L2 y Stopping Prematuro (Imrus & Kang, 2023).

Tabla 7

Comparación Dropout- L2- Early Stopping

Característica	Dropout	Regularización L2 (Weight Decay)	Early Stopping
Mecanismo	Disipa aleatoriamente neuronas mientras ocurre el entrenamiento para impedir dependencia excesiva.	Sanciona la magnitud de los pesos añadiendo la suma de sus cuadrados a la función de pérdida.	Estanca el entrenamiento cuando la medida de validación deja de mejorar.

Objetivo principal	Disminuye co-adaptación de neuronas y evadir sobreajuste.	Conservar pesos pequeños para facilitar el modelo y evadir sobreajuste.	Evitar que el modelo se ajuste en exceso a los datos de entrenamiento.
Implementación	Se aplica en el entrenamiento, con posibilidad de apagado (ej. 20-50%).	Se aumenta un término de penalización inspeccionado por un parámetro λ (lambda).	Se vigila la pérdida o métrica en validación y se para el entrenamiento.
Ventajas	Favorece la robustez y generalización; posible de combinar con otras técnicas.	Menora la complejidad del modelo; mejora estabilidad y generalización.	Simple de implementar; evita sobre entrenamiento sin cambiar su estructura
Desventajas	Puede desacelerar el entrenamiento; pretende el ajuste del parámetro de dropout.	No excluye características considerables; requiere ajuste fino de λ.	Puede interrumpir el entrenamiento apresuradamente si no se ajusta bien la paciencia.
Efecto en el modelo	Entrena múltiples subredes implícitas, inicia redundancia.	Penaliza pesos grandes, dando soluciones más simples.	Restringe el número de épocas para evitar memorizar datos de entrenamiento.
Uso típico	Redes neuronales profundas, principalmente en capas ocultas.	Modelos con pesos ajustables, como redes neuronales y regresión.	Modelos supervisados con conjuntos de validación utilizables.

Nota: Comparación entre los diferentes mecanismos, objetivos, implementación, ventajas, desventajas, efecto en el modelo y uso típico de Dropout, Regularización L2 (Weight Decay), Early Stopping.

2.2.5.5. Optimización: Adam y SGD

Adam (Ajuste Adaptativo de Momentos) es un algoritmo de optimización muy utilizado en aprendizaje automático y aprendizaje profundo, que modifica de manera adaptativa la velocidad de aprendizaje para cada parámetro del modelo. Incorpora los beneficios de los

procedimientos Momentum y RMSprop, empleando medias móviles exponenciales de primer y segundo orden de los gradientes para actualizar los pesos de forma eficaz y constante, promoviendo una convergencia rápida incluso en funciones de pérdida complejas. SGD (Descenso del Gradiente Estocástico) es un método de optimización tradicional que actualiza los parámetros del modelo mediante el uso del gradiente calculado en un subconjunto aleatorio (mini-batch) de datos. Emplea un ritmo de aprendizaje establecido o programado para todos los parámetros y es sencillo, sin embargo, puede necesitar modificaciones meticulosas para prevenir oscilaciones o convergencia gradual (Gabbard, 2018).

Adam resulta particularmente beneficioso para el entrenamiento de redes neuronales profundas y modelos complejos, en contraste con SGD, que es ampliamente utilizado y puede proporcionar una mayor generalización en algunas situaciones si se ajusta adecuadamente.

2.2.5.6. MLP vs LSTM en análisis de sentimientos

Las LSTM sobrepasan a los MLP en el estudio de las emociones gracias a su habilidad para gestionar secuencias y contexto, alcanzando una mayor exactitud y una mejor comprensión del lenguaje natural. Por otro lado, los MLP pueden resultar beneficiosos en contextos con textos breves o cuando se cuenta con representaciones vectoriales estáticas como TF-IDF, proporcionando soluciones más sencillas y veloces (Graves, 2005).

Tabla 8

Características MLP/LSTM

Característica	MLP (Perceptrón Multicapa)	LSTM (Long Short-Term Memory)
Tipo de arquitectura	Red neuronal feedforward con capas completamente conectadas.	Red recurrente con memoria a largo plazo para secuencias.

Manejo de secuencias	No captura dependencias transitorios o contextuales en el texto.	Captura dependencias a largo plazo y contexto secuencial.
Preprocesamiento típico	Entrada basada en vectores estáticos como TF-IDF o embeddings fijos.	Entrada secuencial de vectores de palabras o embeddings.
Precisión en análisis de sentimiento	Resultados aceptables (~76% en algunos estudios), pero menor que LSTM.	Mejor rendimiento, con precisiones reportadas hasta 88-89%.
Capacidad para manejar contexto	Limitada, no modela bien ambigüedades o contexto complejo.	Alta, puede descifrar matices y contexto en oraciones largas.
Limitaciones	No captura bien relaciones temporales ni contexto semántico.	Puede ser menos eficiente en textos muy cortos o datos escasos.

Nota: Características de MLP y LSTM.

2.2.6. Transfer Learning en PLN

2.2.6.1. Embeddings preentrenados

Los embeddings de palabras son representaciones vectoriales llenas de datos que capturan las relaciones semánticas y sintácticas del idioma. En contraposición a representaciones dispersas como el Bag of Words o el TF-IDF, los embeddings codifican las palabras en espacios de menor tamaño donde la proximidad entre vectores muestra analogías orales. Esta habilidad es particularmente útil para modelos de aprendizaje profundo, que necesitan aportes numéricos con información semánticamente semántica (Cachay, 2024).

En el estudio de las emociones, la utilización de embeddings ya entrenados facilita el uso de conocimientos lingüísticos extraídos de grandes volúmenes de texto, lo que potencia de manera notable la generalización del modelo, especialmente en situaciones con cantidades de datos escasas o diversos vocabularios. A continuación, se muestran algunos de los métodos y modelos más significativos para el aprendizaje del español (Cachay, 2024):

Word2Vec: Según Mikolov y colaboradores (2013), Word2Vec aprende representaciones de palabras a través de métodos de predicción contextual (CBOW o Skip-gram). A pesar de ser sensible a la morfología del idioma, se ha usado extensamente en PLN por su sencillez y eficacia.

GloVe (Global Vectors for Word Representation): GloVe, desarrollado por Stanford, fusiona estadísticas de coocurrencia a nivel mundial con aprendizaje local. Sus inserciones son deterministas y han probado su eficacia en tareas de similitud semántica.

FastText: Sugirido por Facebook AI Research, Word2Vec mejora al tomar en cuenta subpalabras o n-gramas, lo que facilita su manejo de palabras poco comunes o no vistas (OOV). Es particularmente beneficioso para idiomas con riqueza morfológica como el español.

BETO: Es un modelo fundamentado en la arquitectura BERT (Representations Bidirectional Encoder from Transformers), que se ha entrenado únicamente con corpus en español. En contraposición a los modelos previos, BETO es contextual, lo que significa que produce varios vectores para una misma palabra en función del contexto en el que se presenta. Esto lo transforma en un instrumento potente para tareas complicadas de interpretación de texto y estudio de emociones.

2.2.6.2. Fine-tuning de embeddings para tareas específicas

El fine-tuning de embeddings es un método que implica ajustar un modelo de embeddings ya entrenado para adecuarlo a una tarea o dominio específico, incrementando de esta manera su exactitud y pertinencia en ese contexto específico. Se basa en un modelo de embeddings previamente entrenado (como Word2Vec, FastText, o modelos fundamentados en transformers). Adicionalmente, se capacita utilizando un conjunto de datos particular de la tarea, modificando los vectores para que capten de manera más efectiva las características semánticas y

contextuales pertinentes. Este procedimiento facilita que el modelo diferencie con mayor precisión entre términos relevantes y contextos específicos, mejorando su rendimiento en usos como la recuperación de datos, la clasificación o el análisis semántico (Kunjal & Kumbong, 2024).

Proceso típico de fine-tuning

1. **Carga del modelo preentrenado:** Se ocupa una arquitectura y pesos base ya entrenados en grandes corpus.
2. **Preparación del dataset específico:** Se determinan datos representativos de la tarea, agaregando ejemplos positivos y negativos (por ejemplo, consultas y pasajes importantes o irrelevantes).
3. **Entrenamiento supervisado:** Se modifican los parámetros del modelo reduciendo una función de pérdida (como la entropía cruzada), potenciando la habilidad del embedding para desglosar datos significativos.
4. **Evaluación continua:** Se supervisa el progreso en métricas concretas para prevenir un sobreajuste y garantizar la generalización.

2.2.7. Interpretabilidad de Modelos de Deep Learning

2.2.7.1. Importancia de la explicabilidad en IA

En inteligencia artificial (IA), la explicabilidad se refiere a la habilidad de un modelo de proporcionar explicaciones claras y entendibles sobre cómo realiza sus proyecciones, toma decisiones o emite sugerencias. Esta característica es esencial para asegurar la transparencia, la confiabilidad y la responsabilidad en los sistemas de Inteligencia Artificial (Willie, 2024).

Importancia de la explicabilidad en IA

- ***Comprensión de fortalezas y limitaciones:*** Facilita a programadores y usuarios

comprender cómo y por qué un modelo opera, simplificando la detección de fallos y la optimización constante del desempeño.

- ***Transparencia y confianza:*** Al entender el proceso interno, los usuarios depositan mayor confianza en los resultados y muestran mayor predisposición a adoptar tecnologías fundamentadas en Inteligencia Artificial.
- ***Cumplimiento normativo:*** En áreas vitales como las finanzas, la salud o la justicia, es necesario tener explicabilidad legal para respaldar decisiones automatizadas que impactan a individuos.
- ***Ética y justicia:*** Asiste en la identificación y reducción de prejuicios, fomentando elecciones más equitativas y responsables, previniendo la persistencia de prejuicios.
- ***Responsabilidad y trazabilidad:*** Facilita el seguimiento y la auditoría de decisiones, esencial en sectores donde las repercusiones son relevantes.

2.2.7.2. Técnicas de interpretabilidad

Las estrategias de interpretación en inteligencia artificial (IA) facilitan la comprensión y explicación de las decisiones de modelos, en particular los complejos o "cajas negras". Principalmente se segmentan en dos enfoques principales: modelos que son inherentemente interpretables y técnicas post-hoc que detallan modelos ya entrenados (Cate, 2025).

Modelos inherentemente interpretables: Son modelos simples y claros por diseño, fáciles de entender. Ejemplos:

- Modelos lineales generalizados (regresión lineal/logística).
- Árboles de decisión.
- Clasificadores bayesianos ingenuos.

- k-vecinos más próximos (k-NN).

Técnicas destacadas post-hoc (Cate, 2025)***:***

LIME (Local Interpretable Model-agnostic Explanations). Elabora un modelo local sencillo (como la regresión lineal) que acerca la predicción del modelo complejo en una vecindad próxima a un caso concreto. Facilita comprender qué rasgos impactaron en esa predicción específica.

SHAP (SHapley Additive exPlanations). En base a la teoría de juegos, atribuye a cada elemento un valor que refleja su aporte a la predicción, teniendo en cuenta todas las combinaciones potenciales de estos elementos. Ofrece explicaciones consistentes y a nivel global o local.

Gráficos de Dependencia Parcial (PDP). Presentan la correlación media entre una o dos variables y la proyección del modelo, facilitando la comprensión de los efectos marginales. Las explicaciones textuales crean descripciones fácilmente entendibles para los seres humanos que sintetizan las causas detrás de una predicción mientras que las visualizaciones consisten en mapas térmicos, diagramas de relevancia de variables o ilustraciones de activaciones neuronales para comprender qué componentes del ingreso afectan la salida.

2.2.7.3. SHAP: fundamentos y aplicación en análisis de sentimientos

SHAP (SHapley Additive exPlanations) es un método de interpretación fundamentado en la teoría de juegos que atribuye a cada elemento de entrada (como una palabra en un texto) un valor que simboliza su aporte individual a la predicción de un modelo de aprendizaje automático. Estos valores provienen de la idea de los valores de Shapley, que calculan la contribución marginal de cada "jugador" (característica) al resultado global del "juego" (predicción del modelo) (Dewi y otros, 2022).

Las propiedades clave de SHAP son:

- **Aditividad:** La totalidad de las aportaciones de las características equivalente a la diferencia entre la proyección del modelo para un caso y el valor base (predicción media).
- **Coherencia:** Si se incrementa la aportación de una característica en un modelo, su valor SHAP no se reduce.
- **Interpretabilidad local y global:** Facilita la explicación de tanto las predicciones personales como la conducta global del modelo.

Aplicación en Análisis de Sentimientos. En el estudio de emociones, SHAP se emplea para interpretar y aclarar cómo cada palabra o atributo textual afecta la categorización de un texto como positivo, negativo o neutral. El procedimiento habitual es el siguiente (Dewi y otros, 2022):

Entrenamiento del modelo: Se capacita un modelo de aprendizaje automático (como la regresión logística, Random Forest, redes neuronales) en un grupo de textos marcados con emociones.

Cálculo de valores SHAP: La librería SHAP se emplea para determinar la relevancia de cada palabra o elemento en la proyección de cada texto. Por ejemplo, en un modelo de regresión logística destinado al estudio de emociones, los valores SHAP señalan qué palabras incrementan o reducen la posibilidad de que un texto se categorice como positivo o negativo.

Visualización: Los resultados pueden ilustrarse a través de esquemas de barras o gráficos de violín, evidenciando qué términos son más relevantes en la elección del modelo y cómo fluctúa su impacto entre ejemplos personales.

SHAP vs. LIME. LIME es un método frecuentemente empleado para la interpretación local de modelos de Inteligencia Artificial. Similar a SHAP, LIME produce explicaciones para predicciones individuales, pero lo hace generando un modelo que se pueda entender alrededor de la predicción concreta que se desea explicar. A pesar de que LIME es más veloz en su cálculo, posee varias restricciones en comparación con SHAP (Dewi y otros, 2022):

Consistencia: SHAP asegura que las explicaciones sean coherentes, en otras palabras, si un modelo modifica de tal forma que una característica ejerce un mayor efecto, SHAP mostrará ese cambio de forma correcta. Por otro lado, LIME puede generar diversas interpretaciones para la misma predicción al basarse en la creación de muestras aleatorias alrededor del punto de interés.

Fundamento teórico: SHAP se basa en los principios de Shapley, los cuales se apoyan firmemente en la teoría de los juegos y aseguran una distribución justa de las contribuciones. LIME carece de este robusto cimiento teórico y, en consecuencia, puede producir explicaciones menos fiables en determinadas circunstancias.

CAPITULO 3:

3. DESARROLLO

3.1. Desarrollo del trabajo.

3.1.1. Preprocesamiento de datos

La preparación de los datos para un proyecto de machine learning es un proceso complejo, debido a que cada conjunto de datos es diferente y específico para cada caso, sin embargo, se han podido identificar suficientes aspectos en común para delinear un proceso que permita procesar los datos de manera adecuada, por esta razón, se depuró al conjunto de datos inicial para que el desempeño de los algoritmos sea mayor. El objetivo del presente trabajo fue analizar las emociones expresadas en tuits en ingles en el dataset “Emotions” ubicado en el repositorio en línea de Kaggle, para categorizar las emociones expresadas en las publicaciones de los usuarios.

El conjunto de datos utilizado se encuentra en el idioma inglés y tiene 4 campos y 416.809 registros, los 4 campos son los siguientes: Índice, texto donde se encuentran los tuits de los usuarios, etiqueta y emoción.

La codificación del campo “etiqueta” es la siguiente:

- 0: 'sadness',
- 1: 'joy',
- 2: 'love',
- 3: 'anger',
- 4: 'fear',

- 5: 'surprise'

Se realizó un análisis exploratorio de donde se verificaron y removieron los valores nulos y duplicados, además se analizó la distribución de clases para determinar la proporción de cada emoción en el conjunto de datos. A continuación, se analizó la distribución de la cantidad de palabras y caracteres del campo “texto” para entender la naturaleza del lenguaje para cada emoción.

Después del análisis exploratorio de datos se eliminaron las “stopwords” que son en esencia palabras muy frecuentes y de manera general poco informativas, como artículos, preposiciones, conjunciones, pronombres entre otras. Es importante remover las “stopwords” porque así se reduce el ruido y la dimensionalidad en el modelo, además se mejora la generalización y se facilita la interpretabilidad, un factor importante también tiene que ver con la reducción del costo computacional en el entrenamiento Cdel modelo de machine learning. Como resultado de este procesamiento se obtuvo una versión depurada del conjunto de datos original (Palomino & Aider, 2022).

3.1.2. Desarrollo del Modelo

Para esta sección se trabajó con la versión depurada del conjunto de datos original debido a que el rendimiento fue significativamente superior al obtenido utilizando los datos en bruto. Por esta razón, todos los procedimientos descritos a continuación se realizaron exclusivamente con esta versión optimizada del conjunto de datos.

El conjunto de datos fue dividido en tres subconjuntos: un 80 % se destinó al entrenamiento del modelo, mientras que el 20 % restante fue dividido equitativamente entre validación y prueba (10 % cada uno). Esta segmentación mantuvo la proporción de categorías de

la variable objetivo en cada subconjunto, permitiendo asegurar que el modelo aprendiera de una distribución representativa y pudiera ser evaluado con criterios consistentes.

Para transformar el texto en una forma adecuada para el modelo, se construyó un vocabulario limitado (de tamaño de 10 000 o 25 000 palabras), y se estableció una longitud fija para las secuencias (20, 50 o 100 tokens). Esto permitió estandarizar las entradas textuales y prepararlas para su procesamiento por la red neuronal.

Con el objetivo de enriquecer la representación del lenguaje, se incorporaron vectores de palabras preentrenados en inglés conocidos como FastText. Estos vectores, con 300 dimensiones, fueron desarrollados a partir de grandes corpus multilingües como Common Crawl (Bojanowski y otros, 2017)

A diferencia de las codificaciones numéricas simples, los embeddings permiten capturar similitudes y relaciones entre palabras, proporcionando al modelo un conocimiento lingüístico previo que mejora su comprensión semántica. Se evaluaron configuraciones utilizando 50 000 y 100 000 vectores. Cuando una palabra del vocabulario no estaba disponible en los vectores cargados, se le asignó un vector aleatorio generado de forma controlada. A partir de esta información, se construyó una matriz de representaciones vectoriales que sirvió como base para la red neuronal.

La arquitectura del modelo combinó mecanismos secuenciales y densos, incluyendo una capa de representación semántica ajustable, seguida por una unidad LSTM bidireccional (de 32, 64 o 128 unidades) que permitió capturar relaciones contextuales tanto hacia adelante como hacia atrás en las secuencias de texto. Este tipo de arquitectura resulta especialmente útil para tareas como el análisis de sentimientos, donde el significado de una palabra puede depender tanto del contexto anterior como del posterior (Prechelt, 1998) . A continuación, se incorporó una

capa densa (de 32, 64 o 128 unidades) con activación no lineal (ReLU), acompañada de técnicas de regularización mediante penalizaciones L1 y L2. Estas tasas de regularización no fueron fijas, sino que se exploraron en tres niveles distintos (0.0, 0.01 y 0.001), como parte del proceso de ajuste fino del modelo. También se incluyó una capa de abandono (*dropout*), del 30% o 60% para mitigar el sobreajuste y mejorar la capacidad de generalización. Finalmente, la arquitectura se cerró con una capa de salida con activación *softmax*, adecuada para la clasificación multiclase.

Para el entrenamiento del modelo, se consideraron dos algoritmos de optimización ampliamente utilizados en aprendizaje profundo: Adam y RMSprop. Ambos optimizadores fueron evaluados con dos tasas de aprendizaje distintas (0.01 y 0.001), permitiendo analizar la sensibilidad del modelo frente a diferentes dinámicas de ajuste de pesos.

Dado que el problema abordado corresponde a una clasificación multiclase, fue necesario emplear una función de pérdida compatible con este tipo de tareas, como la entropía cruzada categórica, que penaliza con mayor intensidad las predicciones incorrectas. Asimismo, se utilizó una capa de salida con activación softmax, la cual permite asignar una distribución de probabilidad sobre las distintas clases posibles, facilitando así la interpretación de la predicción como una medida de confianza del modelo.

Durante el proceso de entrenamiento se monitorizó continuamente el rendimiento del modelo en el conjunto de validación. Si el error de validación no mejoraba durante tres iteraciones consecutivas, se detenía el entrenamiento y se restauraban los pesos del modelo correspondientes al mejor desempeño observado. Esta estrategia de *early stopping* evitó el sobreajuste y redujo el tiempo computacional innecesario (Prechelt, 1998).

Además, para mejorar la eficiencia del proceso, se trabajó con conjuntos de datos optimizados que incorporan técnicas como mezcla aleatoria reproducible, agrupamiento por lotes

y lectura anticipada en memoria. Esto permitió aprovechar mejor los recursos computacionales disponibles, haciendo más eficiente el entrenamiento en cada época.

Entonces, con el objetivo de encontrar la mejor configuración del modelo, se realizó una exploración sistemática de hiperparámetros. Las combinaciones evaluadas fueron las siguientes:

- Tamaño del vocabulario: 10 000 y 25 000
- Longitud de secuencia: 20, 50 y 100 tokens
- Número de vectores FastText cargados: 50 000 y 100 000
- Tipo de optimizador: Adam y RMSprop
- Número de unidades LSTM: 32, 64 y 128
- Número de unidades densas: 32, 64 y 128
- Regularización L1 y L2: 0.0, 0.01 y 0.001
- Tasa de aprendizaje: 0.01 y 0.001
- Tasa de abandono (*dropout*): 0.3 y 0.6

En total, se evaluaron 2592 combinaciones distintas de estos hiperparámetros. Para cada una, se entrenó un modelo independiente y se registró su desempeño en el conjunto de validación. La configuración final seleccionada fue aquella que alcanzó el menor valor de pérdida, garantizando así un balance entre capacidad de aprendizaje y generalización.

Una vez entrenado el modelo final, se procedió a evaluar su desempeño sobre el conjunto de prueba, empleando métricas como precisión y exactitud, así como la matriz de confusión, lo cual permitió cuantificar su rendimiento de forma objetiva.

Finalmente, se aplicaron técnicas de interpretabilidad basadas en el enfoque SHAP (*SHapley Additive exPlanations*) con el fin de comprender cómo el modelo tomaba sus decisiones (Lundberg & Lee, 2017). Para ello, se seleccionaron 80 muestras aleatorias de cada

una de las seis categorías de sentimiento, todas con exactamente 12 tokens, lo que permitió realizar un análisis detallado de la importancia relativa de cada posición en la secuencia textual. Este análisis se llevó a cabo tanto de forma agregada, considerando la totalidad de las muestras, como dividido por clase, con el objetivo de explorar si el modelo otorga un peso diferencial a ciertas posiciones dependiendo del tipo de emoción.

Adicionalmente, se realizó un análisis local por clase, seleccionando una muestra representativa para cada categoría emocional (*sadness*, *joy*, *love*, *anger*, *fear* y *surprise*). En cada una de estas muestras se identificaron los cinco tokens individuales con mayor impacto sobre la predicción del modelo. Esta segunda aproximación permitió observar de forma puntual cómo ciertas palabras específicas influyen en la clasificación final, y cómo estos patrones pueden variar entre clases.

3.1.3. Implementación de interfaz grafica

La etapa de la implementación de una interfaz gráfica a partir de la herramienta seleccionada QT6, fue considerada a partir de una estructura modular para poder separar la estructura de procesamiento lógico de la estructura de la etapa gráfica.

El esquema estructural de la implementación esta basado en

- Predict_emotion.py (script de control de interfaz gráfica)
- Form.ui (estructura visual de la interfaz)
- Predict_emotion.py (módulo de procesamiento del modelo y resultados)
- Keras/ (contiene el modelo y archivos de procesamiento)

La división permite mantener una estructura limpia y reutilizable para la carga de otros modelos.

3.1.3.1.Carga de modelos

Se contempló disponer del modelo exportado en ".*keras* " para poder considerar la inclusión de pesos, optimizador y métricas de configuración del entrenamiento, a fin de evitar la disposición de múltiples directorios.

Las librerías principales para su ejecución fueron "tensorflow", "shap", "nltk", "joblib", "matplotlib" permitiendo de esta manera el procesamiento y análisis de la data obtenida como resultado del input del modelo entrenado

3.1.3.2.Procesamiento y análisis

Limpieza de texto. A partir de la misma lógica considerada durante el preprocesamiento del dataset, mismo que fue utilizado para el entrenamiento del modelo, se consideró la realización del mismo filtrado de datos tal que se pueda disponer un input similar, a fin de que el patrón de reconocimiento no se vea afectado por elementos que puedan resultar ambiguos durante el análisis, de esta manera, el texto ingresado pasa por un filtrado de *"stop words"* tal que se eliminen palabras que son o nada relevantes para la obtención del resultado deseado

Extracción de Embeddings. Se obtiene el embedding del texto de la segunda capa del modelo para su empleo en el análisis de la distribución de activaciones neuronales por token, estos vectores se encargan de la interpretación subyacente de cada palabra, logrando así un procesamiento más efectivo, de esta manera se traduce cada palabra a un lenguaje matemático para el sistema entrenado donde vectores similares provocan contextos similares dando una interpretación semántica de la expresión

Resultados. En consideración del input introducido al modelo se dispone de la impresión del resultado en base a la emoción obtenida y el porcentaje de reconocimiento que le atribuye el modelo

Análisis gráfico

Probabilidades por clase. Muestra el nivel de porcentaje obtenido del resultado para cada clase.

Atención por token. Muestra la influencia de cada token en la decisión de la clasificación similar a SHAP.

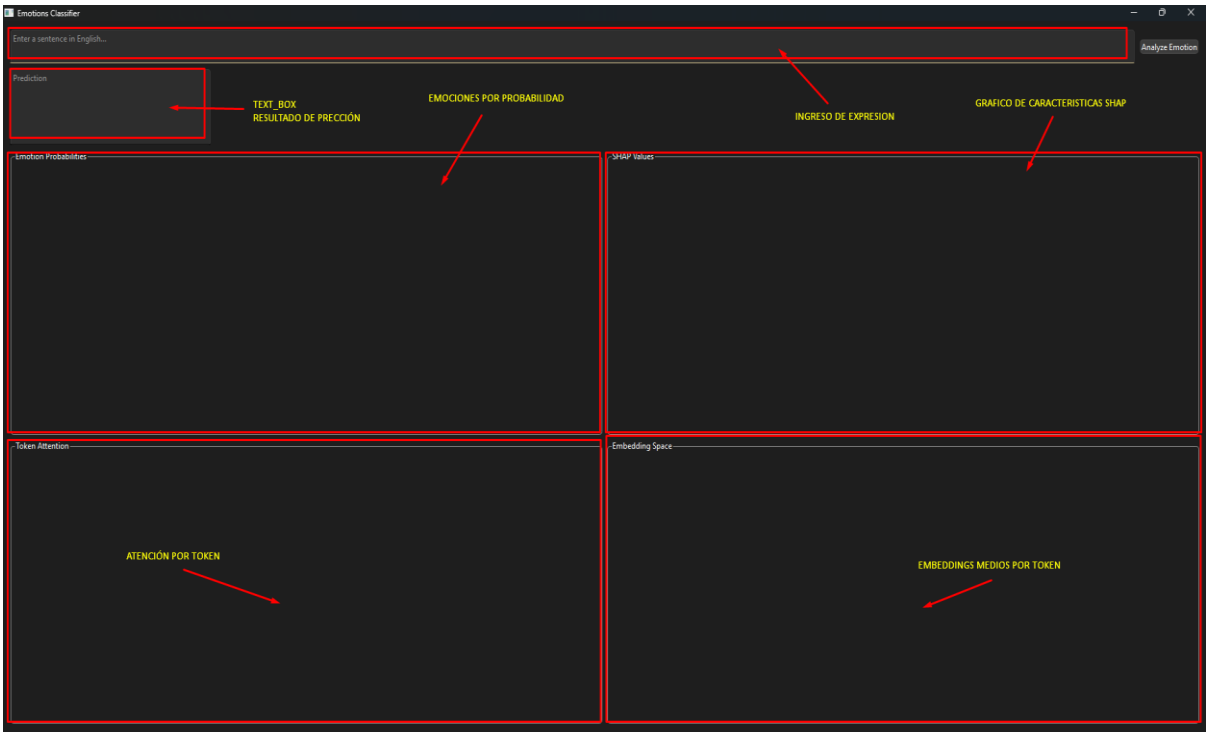
Gráfico de importancia de características SHAP. Demuestra la importancia de cada toquen y su porcentaje de interacción ante el resultado obtenido en comparación con los demás tokens su dependencia sobre cómo interactúan los rasgos al potenciarse o neutralizarse entre sí, captando la relación no lineal entre variables.

Gráfico de vectores de embedding medios. Demuestra un perfil numérico de la frase, dando a conocer la densidad semántica de la misma, en el eje x están representados los componentes dimensionales del embedding y en el eje y el valor medio de cada token.

En la Figura 1 se puede apreciar la distribución de los elementos de la interfaz gráfica permitiendo la fácil visualización de los resultados y su análisis gráfico

Figura 1

Interfaz gráfica de aplicación de reconocimiento de emociones



3.2.Marco Teórico

El estudio de emociones en plataformas como Twitter se ha transformado en un instrumento esencial para entender las emociones de los usuarios y adquirir una comprensión más detallada de cómo perciben diferentes asuntos. Esta técnica, además de facilitar la identificación de puntos de vista positivos o negativos, permite la identificación de una variedad más extensa de emociones, como la alegría, la tristeza, el enojo y el temor. Este estudio tiene usos pertinentes en campos como la sociología, la psicología, el marketing y la política, y ayuda a producir percepciones útiles para la toma de decisiones fundamentadas (García & López, 2020).

Aprendizaje Automático y Procesamiento del Lenguaje Natural (PLN)

El estudio de las emociones se basa principalmente en las técnicas de Procesamiento de Lenguaje Natural (PLN), que permiten que las máquinas entiendan, interpreten y generen lenguaje humano de forma que resulte entendible para los usuarios. No obstante, el PLN se topa con múltiples retos propios del lenguaje humano, en particular cuando se trata de textos informales y cortos, como los tuits. Elementos como la incertidumbre semántica, las abreviaturas, los términos técnicos y los emoticonos pueden dificultar el proceso de análisis. Estos retos complican la labor de extraer emociones de tuits sin estructuración (Pérez & Gómez, 2019)

Por otro lado, el aprendizaje automático ha revolucionado este ámbito al incorporar algoritmos capaces de detectar patrones y correlaciones en grandes cantidades de información textual. Los modelos convencionales de clasificación, que únicamente distinguen entre polaridades elementales (positivo, negativo, neutral), no facilitan una interpretación detallada de las emociones que se encuentran en los mensajes. Por esta razón, en esta investigación se aplica un método más exhaustivo que categoriza las emociones en grupos más concretos, como la alegría, la tristeza, el enojo y el temor, lo que facilita una interpretación más integral de las opiniones manifestadas en los tuits (Martínez & Sánchez, 2021).

Modelos de Aprendizaje Autodidacta en el Estudio de las Emociones

Las Redes Neuronales Artificiales (RNA) se encuentran entre las arquitecturas más empleadas en el estudio de las emociones, gracias a su habilidad para adquirir representaciones complejas de los datos. En este escenario, se ha comprobado que el Perceptrón Multicapa (MLP), una red neuronal profunda, es efectivo en labores de clasificación en diferentes dominios. No obstante, en el tratamiento del lenguaje natural, los modelos fundamentados en

Redes Neuronales Recurrentes (RNN) han demostrado un desempeño destacado. Las RNN se han creado específicamente para manejar series de datos, lo que resulta vital para el análisis de textos (Fernández & Ruiz, 2020)

En las Redes Neuronales Recurrentes con Memoria a Largo Plazo (LSTM) bidireccionales resultan particularmente beneficiosas para el estudio de emociones en tuits. Estas redes no solo manejan los textos de forma secuencial (de izquierda a derecha), sino también en el sentido contrario (de derecha a izquierda), lo que les facilita la captura del contexto integral de los mensajes. Esta habilidad para conservar información a largo plazo es crucial, dado que numerosos tuits incluyen dependencias temporales y contextuales que necesitan ser entendidas en su totalidad (Torres & Ramírez, 2021).

Interpretabilidad en Modelos Automáticos de Aprendizaje: SHAP

Uno de los elementos cruciales en el estudio de las emociones es la capacidad de los modelos para ser interpretado. Aunque las redes neuronales profundas exhiben un alto desempeño en cuanto a exactitud, su carácter de "caja negra" complica la interpretación de cómo y por qué el modelo realiza elecciones concretas. Para resolver este asunto, se utilizan instrumentos como SHAP (SHapley Additive exPlanations), que facilitan la comprensión de cómo características particulares, como palabras o expresiones, afectan la predicción final del modelo. La implementación de SHAP en el estudio de las emociones incrementa la claridad y la seguridad en los resultados, ofreciendo una perspectiva precisa de los elementos que influyen en la categorización de las emociones. Esto simplifica la verificación e interpretación de los modelos, aspecto crucial para que tanto los investigadores como los usuarios puedan tomar decisiones fundamentadas en las emociones identificadas en los tuits (Gómez & Hernández, 2022).

CAPITULO 4:

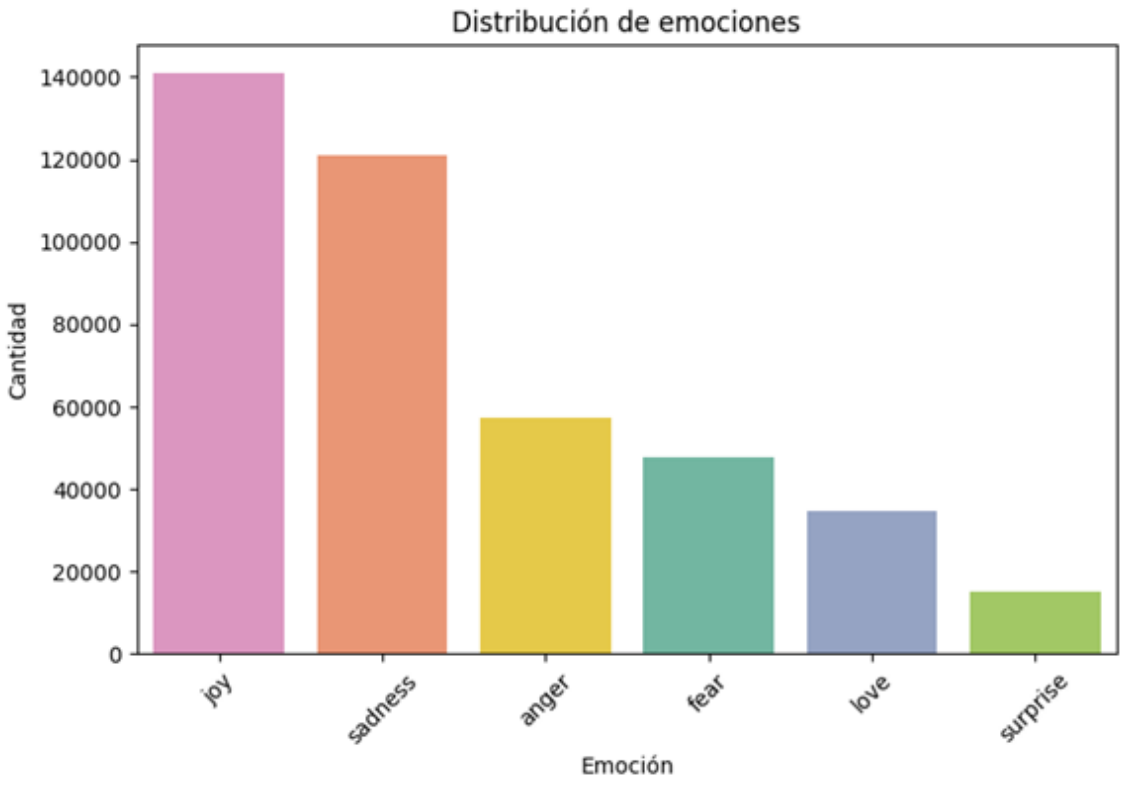
4. DESARROLLO

4.1. Pruebas de Concepto

En la etapa de procesamiento de datos se identificó la distribución de las emociones a lo largo del conjunto de datos, como muestra la Figura 2 la emoción “joy” tiene 140.779 registros, se aprecia claramente que la distribución de clases está desbalanceada, por lo que es posible que el modelo pueda sesgarse a las clases mayoritarias y que clases minoritarias como “surprise” o “love” puedan ser mal clasificadas.

Figura 2

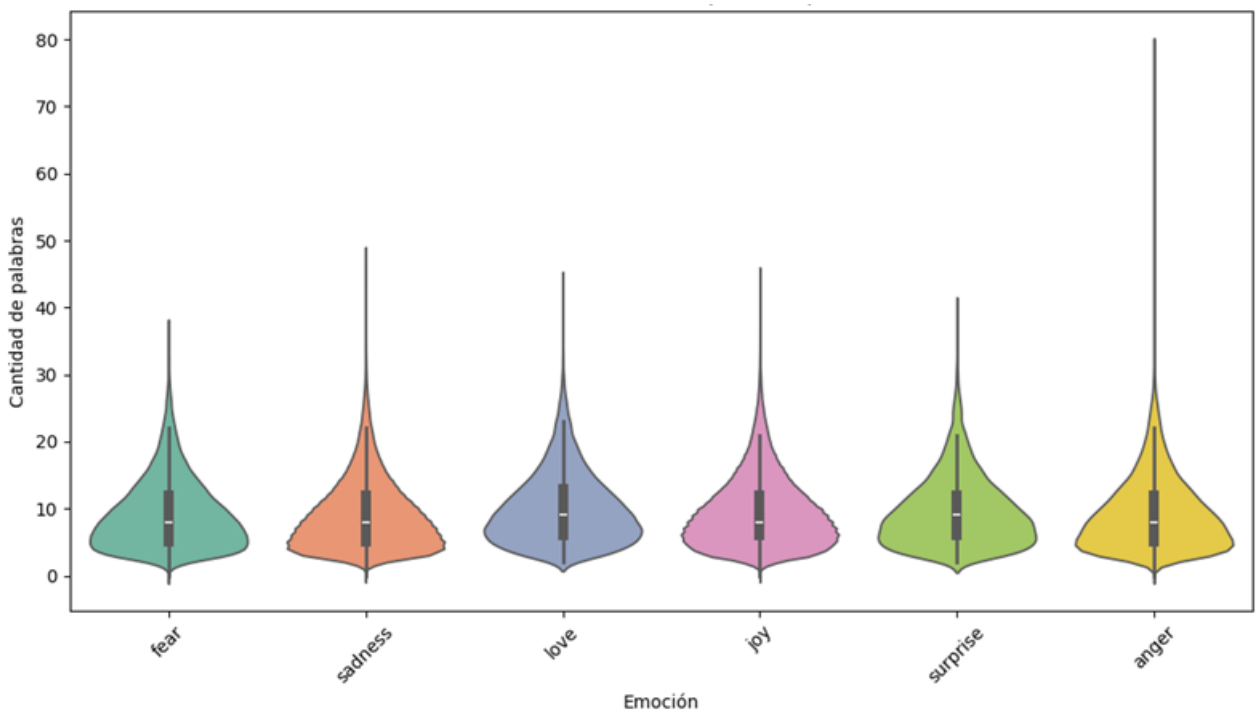
Distribución de emociones en el conjunto de datos.



Con el objetivo de reducir el ruido, optimizar la carga computacional y mejorar el rendimiento, se eliminaron las “stopwords” y como se muestran en la Figura 3 se obtuvo una distribución de la cantidad de palabras por emoción. Se puede notar que la mediana se encuentre entre 5 a 10 palabras para la mayoría de las emociones, sin embargo se puede evidenciar que emociones como “anger” presentan una mayor dispersión con colas largas, esto puede influir en la forma en la que el modelo se entrena.

Figura 3

Distribución de la cantidad de palabras por emoción



Al realizar un análisis intercuartílico de la cantidad de palabras podemos notar lo siguiente:

- El 25% de los textos en el conjunto de datos tienen una longitud de 5.0 palabras o menos
- El 75% de los textos en el conjunto de datos tienen una longitud de 12.0 palabras o menos

- El rango en el que se encuentra el 50% de los datos es 7.0

La mayoría de los textos son bastante cortos. El 50% de los textos más "típicos", los que están entre el 25% y el 75% de la distribución, tienen entre 5 y 12 palabras. Una vez completada la etapa de preprocesamiento del conjunto de datos, se procedió a la implementación del modelo, a continuación se presentan los resultados obtenidos a partir de dos ejes fundamentales: por un lado, la exploración cuantitativa del espacio de hiperparámetros, enfocada en la minimización de la función de pérdida; y por otro, el análisis del comportamiento del modelo final tanto en términos de desempeño predictivo como de interpretabilidad.

La primera parte se centra en la evaluación de las 2,592 combinaciones de configuración posibles, con el objetivo de identificar los conjuntos de parámetros que ofrecieron el mejor compromiso entre generalización y precisión. Posteriormente, se reportan las métricas de clasificación obtenidas sobre el conjunto de prueba, junto con un análisis más profundo de la distribución de errores y la interpretabilidad de las predicciones mediante el uso de técnicas de explicación basadas en SHAP.

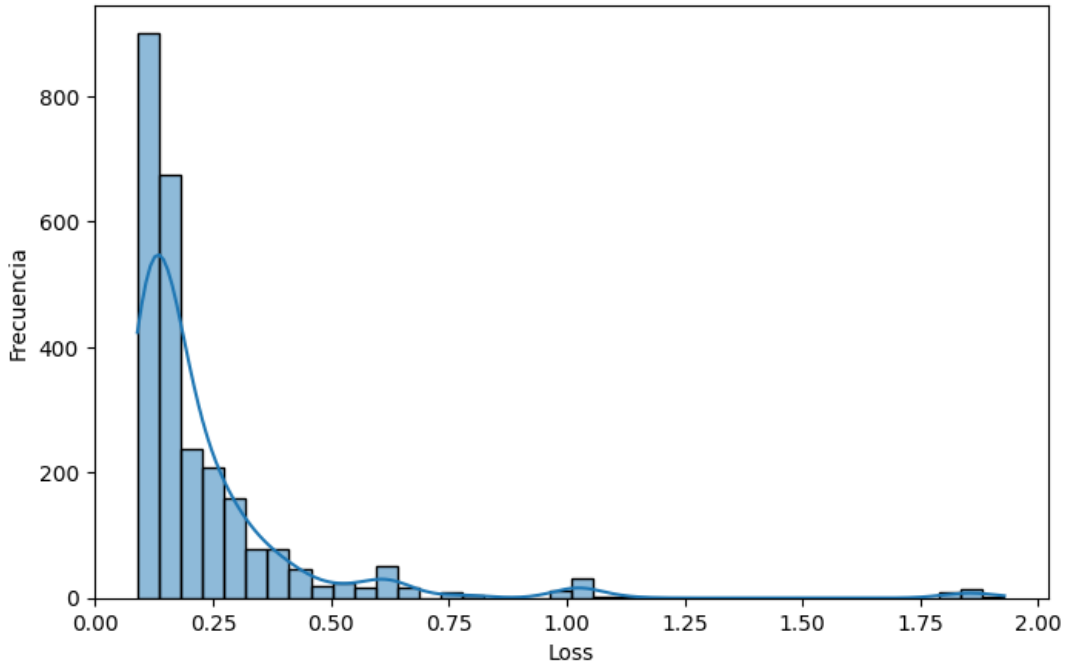
Evaluación de combinaciones e impacto en la función de pérdida. Como parte del proceso de ajuste fino del modelo, se entrenaron un total de 2,592 combinaciones distintas de hiperparámetros, variando aspectos como el tamaño del vocabulario, la longitud de la secuencia, la arquitectura interna, el optimizador y la tasa de regularización. Para evaluar objetivamente el desempeño de cada configuración, se utilizó la función de pérdida (*categorical crossentropy*) como métrica principal.

El histograma de los valores de pérdida obtenidos durante el entrenamiento muestra una distribución asimétrica con cola a la derecha, extendiéndose desde aproximadamente 0.08 hasta 1.93 como se muestra en la Figura 4. Esta forma indica que, si bien algunas combinaciones

generaron resultados considerablemente menos eficientes, la mayoría de los modelos logró un desempeño aceptable, con una fuerte concentración de valores en la región baja de la distribución.

Figura 4

Distribución de la función de pérdida en las combinaciones



En particular, se observa que la mayoría de las combinaciones se agrupan en los dos primeros intervalos del histograma, lo que sugiere que existe una alta densidad de configuraciones efectivas dentro del espacio de búsqueda considerado. Esto refleja tanto la robustez de la arquitectura utilizada, como la calidad del preprocesamiento —especialmente el uso de datos depurados y vectores preentrenados.

Tras el análisis comparativo, se identificó la configuración óptima que produjo el menor valor de pérdida, con un loss final de 0.0895. Los mejores hiperparámetros encontrados fueron los siguientes:

- Tamaño del vocabulario: 25 000
- Longitud de secuencia: 50
- Número de vectores FastText cargados: 100 000
- Tipo de optimizador: RMSprop
- Número de unidades LSTM: 64
- Número de unidades densas: 64
- Regularización L1 y L2: 0.0
- Tasa de aprendizaje: 0.01
- Tasa de abandono (*dropout*): 0.3

Estos valores sirvieron como base para entrenar el modelo final, sobre el cual se realizaron las evaluaciones cuantitativas y los análisis de interpretabilidad presentados en las siguientes secciones.

Desempeño del modelo en el conjunto de prueba. Una vez seleccionado el conjunto óptimo de hiperparámetros, se entrenó el modelo final y se evaluó su rendimiento utilizando el conjunto de prueba, previamente separado y no utilizado durante la etapa de entrenamiento ni validación.

El modelo alcanzó una exactitud global del 94 %, lo que indica un alto nivel de desempeño general como se muestra en la Figura 5 El F1-score ponderado, que considera el tamaño relativo de cada clase, también fue de 0.94, lo que confirma que el modelo logra un equilibrio efectivo entre precisión y recall, incluso en presencia de clases desbalanceadas. No obstante, al observar las métricas a nivel de clase, emergen diferencias importantes en el comportamiento del modelo.

Figura 5

Reporte de clasificación

	precision	recall	f1-score	support
Sadness	0.98	0.97	0.98	12191
Joy	0.92	1.00	0.96	14224
Love	0.99	0.70	0.82	3375
Anger	0.94	0.96	0.95	5612
Fear	0.88	0.94	0.91	4732
Surprise	0.98	0.64	0.77	1479
accuracy			0.94	41613
macro avg	0.95	0.87	0.90	41613
weighted avg	0.94	0.94	0.94	41613

Las emociones Sadness, Joy, y Anger presentan desempeños notablemente altos: *Sadness* obtuvo un F1-score de 0.98, *Joy* alcanzó 0.96, y *Anger* 0.95. Esto sugiere que estas clases están bien representadas en los datos y poseen patrones lingüísticos consistentes que el modelo ha aprendido a identificar con alta confiabilidad.

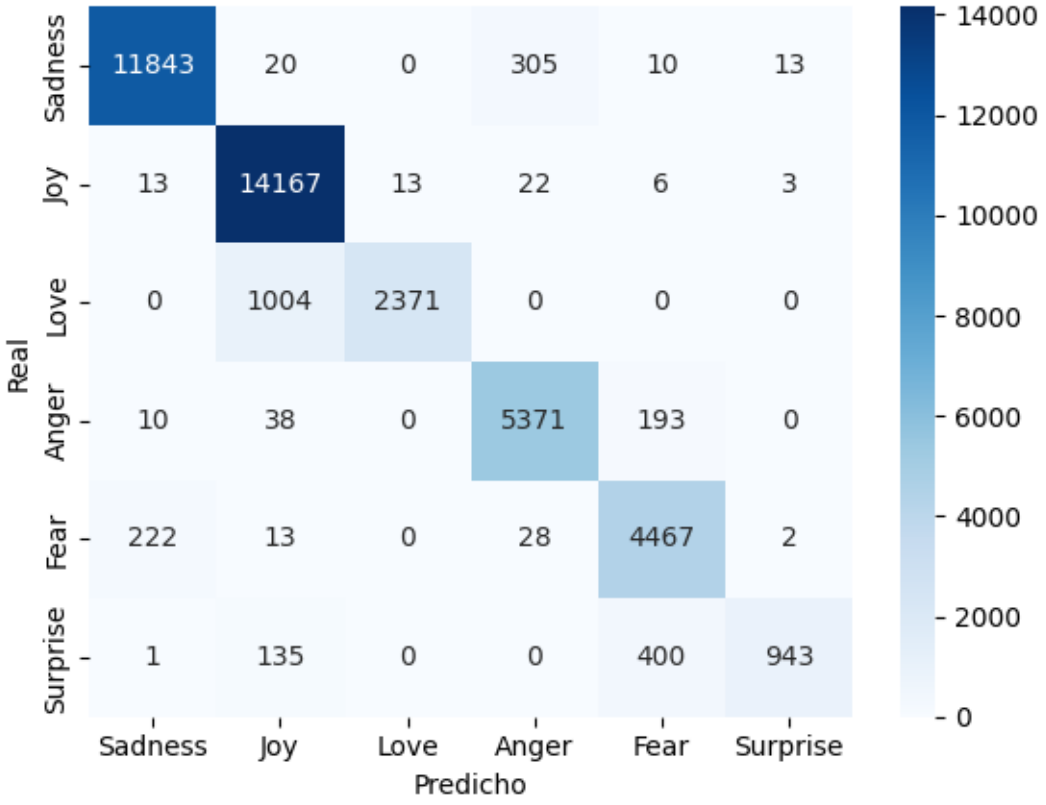
En contraste, las clases Love, Fear y especialmente Surprise muestran un comportamiento menos robusto. *Love*, por ejemplo, si bien tiene una precisión de 0.99 (muy pocas predicciones positivas incorrectas), su recall cae a 0.70, lo que significa que el modelo no logra detectar aproximadamente el 30 % de las muestras verdaderas de esta categoría. Algo similar ocurre con *Surprise*, que aunque presenta una precisión de 0.98, tiene un recall de apenas 0.64. Esto podría indicar que estas emociones están subrepresentadas o presentan mayor ambigüedad lingüística, lo que dificulta su identificación incluso en presencia de indicadores precisos.

La matriz de confusión proporciona mayor detalle sobre los errores de clasificación. Se observa que las principales confusiones ocurrieron entre clases con cierta cercanía semántica así

como lo muestra la Figura 6, por ejemplo, *sadness* fue ocasionalmente confundida con *anger* (305 casos), mientras que *love* fue confundida en un número significativo de ocasiones con *joy* (1004 casos), lo cual es consistente con el solapamiento emocional y lingüístico entre ambas categorías. Por su parte, *fear* fue clasificada erróneamente como *sadness* en 222 casos, y *surprise* confundido con *fear* en 400 casos, lo cual puede reflejar la ambigüedad de expresiones que comparten rasgos afectivos similares.

Figura 6

Matriz de confusión



El análisis macropromediado muestra un F1-score de 0.90, reflejando una ligera penalización por el desempeño desigual entre clases, especialmente aquellas con menor soporte. A pesar de ello, estos resultados siguen siendo altamente competitivos para tareas de

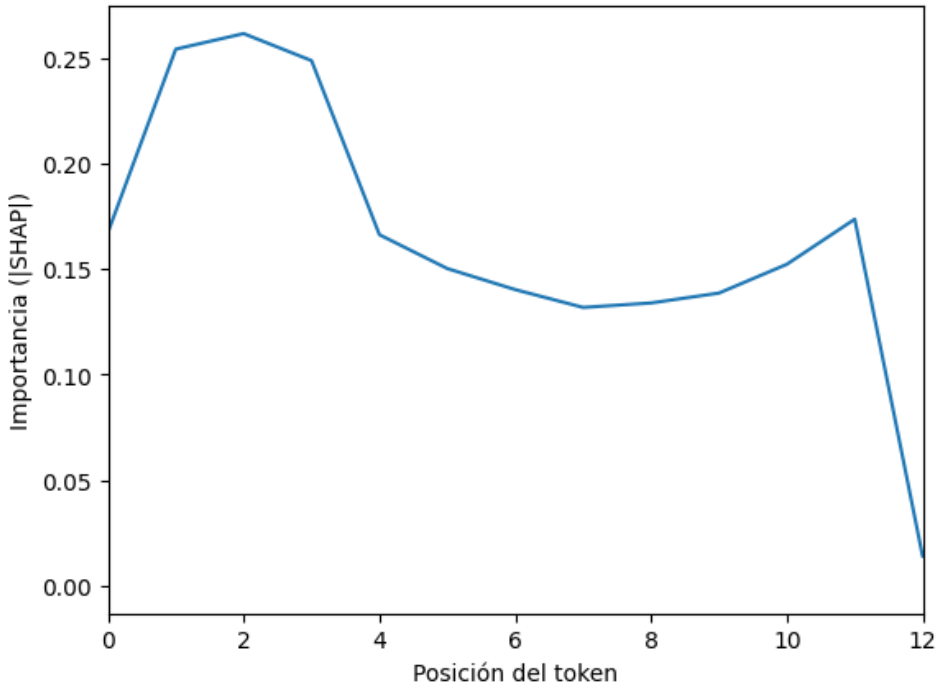
clasificación multiclase en procesamiento de lenguaje natural, especialmente considerando la complejidad emocional y semántica involucrada.

4.2. Análisis de resultados

En la Figura 7 se puede notar la importancia promedio por posición, en primer lugar, se examinó la importancia promedio de cada posición del token dentro de las secuencias de entrada, tomando un subconjunto de 480 muestras (80 por cada clase), todas con una longitud fija de 12 tokens. En la Figura 7 se evidencia que las posiciones iniciales —especialmente las ubicadas entre el segundo y cuarto token— presentan los valores de importancia más altos, alcanzando un valor SHAP cercano a 0.26. A partir de la posición 5 en adelante, la contribución tiende a estabilizarse y disminuir, con un leve repunte en las posiciones 10 y 11, y una caída marcada en la última posición.

Figura 7

Importancia promedio por posición del token



Este patrón sugiere que el modelo tiende a extraer la mayor parte de la información relevante desde las primeras palabras del texto, lo cual puede estar relacionado con la forma en que los enunciados emocionales están redactados: frecuentemente, los tokens iniciales contienen verbos, adjetivos o expresiones que establecen el tono emocional del mensaje.

Importancia promedio por posición según la clase emocional. Con el fin de profundizar en los patrones de atención del modelo, se desagregó el análisis anterior según las seis clases emocionales: *sadness*, *joy*, *love*, *anger*, *fear* y *surprise*. Para cada clase se mantuvo el mismo número de muestras (80), todas con secuencias de 12 tokens, permitiendo así comparar directamente cómo varía la distribución posicional de la importancia entre categorías.

En general, todas las clases comparten una tendencia a presentar mayor peso en los tokens iniciales. No obstante, se evidencian diferencias significativas en el grado y la forma en que esta atención se distribuye, tal y como se puede observar en la Figura 8.

Anger, surprise, joy y love presentan un patrón marcadamente frontal: las primeras cuatro posiciones concentran prácticamente el mayor peso (en *surprise*, incluso se observan valores cercanos a 0.20 en la posición 3), lo que sugiere una estructura lingüística más directa o explosiva en este tipo de emociones.

Fear muestra un perfil más inestable, con picos de importancia en posiciones intermedias y finales, lo cual podría indicar que el modelo necesita capturar mayor contexto para identificar correctamente esta emoción.

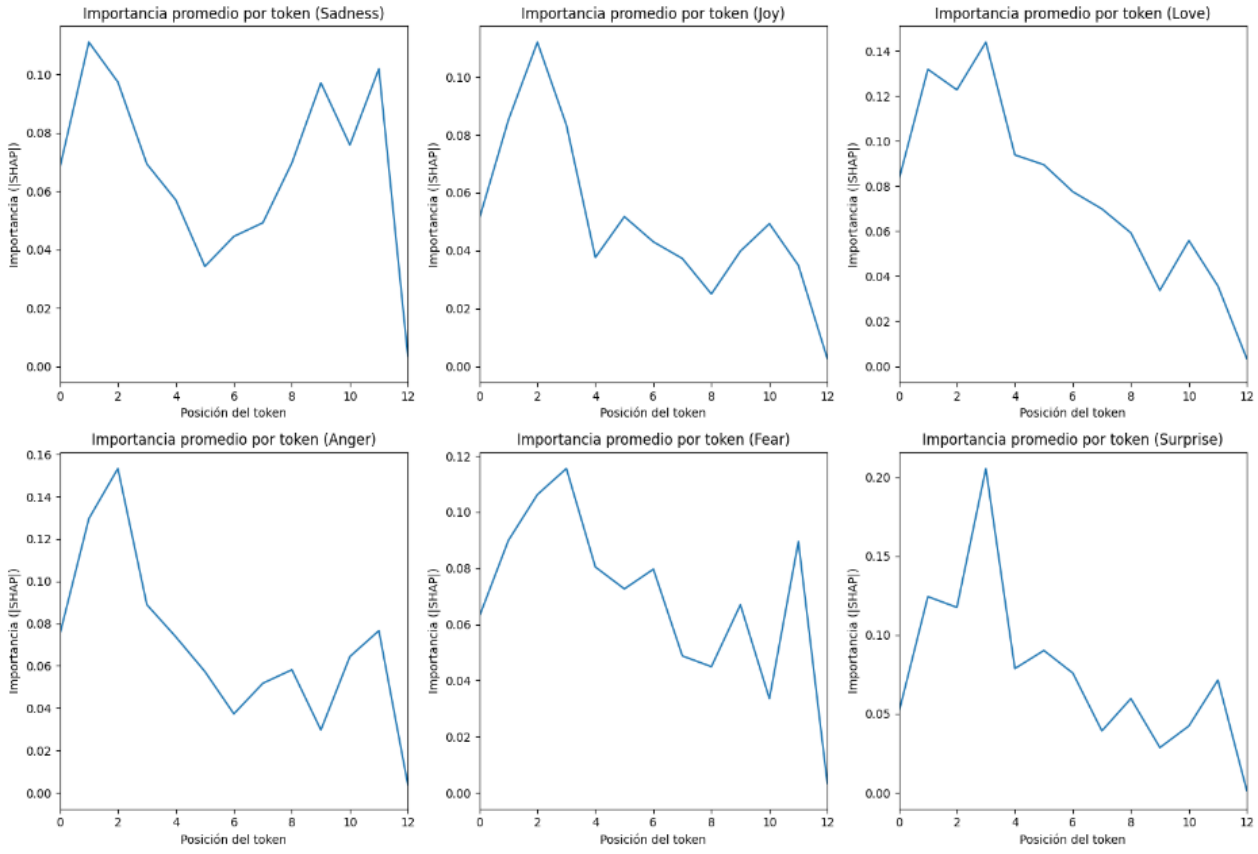
Sadness destaca por un patrón atípico: si bien los tokens iniciales son relevantes, también se observa una recuperación significativa de la importancia en posiciones finales, especialmente en la posición 11. Esto sugiere que las expresiones de tristeza en el corpus utilizado tienden a

cerrarse con palabras claves que el modelo ha aprendido a reconocer como indicativas de dicha emoción.

En resumen, aunque las primeras posiciones muestran una alta importancia en general, el patrón de distribución varía según la clase emocional. Cada emoción exhibe un perfil de atención único, lo que sugiere que el modelo adapta su interpretación de las secuencias en función del tipo de sentimiento. Esta capacidad para ajustar la atención indica que el modelo no sigue reglas predefinidas, sino que aprende representaciones dinámicas que dependen del contexto emocional del texto.

Figura 8

Importancia promedio por posición según la clase

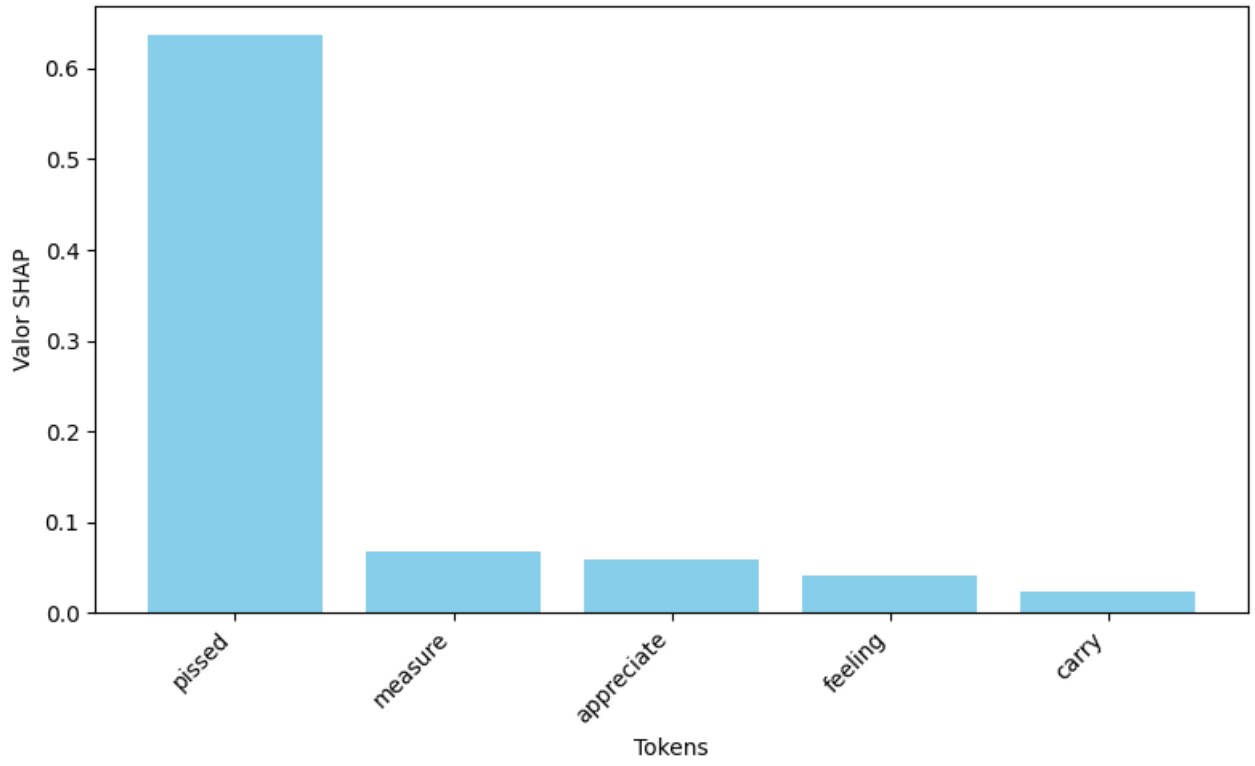


Interpretabilidad local*: tokens más influyentes por clase.* Como complemento al análisis por posición, se identificaron los cinco tokens con mayor valor SHAP en una muestra representativa de cada clase, con el fin de explorar qué palabras individuales contribuyen de manera más significativa a las decisiones del modelo. Estos tokens permiten observar cómo se manifiestan las señales semánticas específicas asociadas a cada emoción. Los resultados se describen a continuación:

Anger. La figura 9 muestra las características más relevantes para esta emoción, el review utilizado fue: appreciate beyond measure still carry guilt feeling pissed ed ness body able.

Figura 9

Top 5 características más importantes de Anger

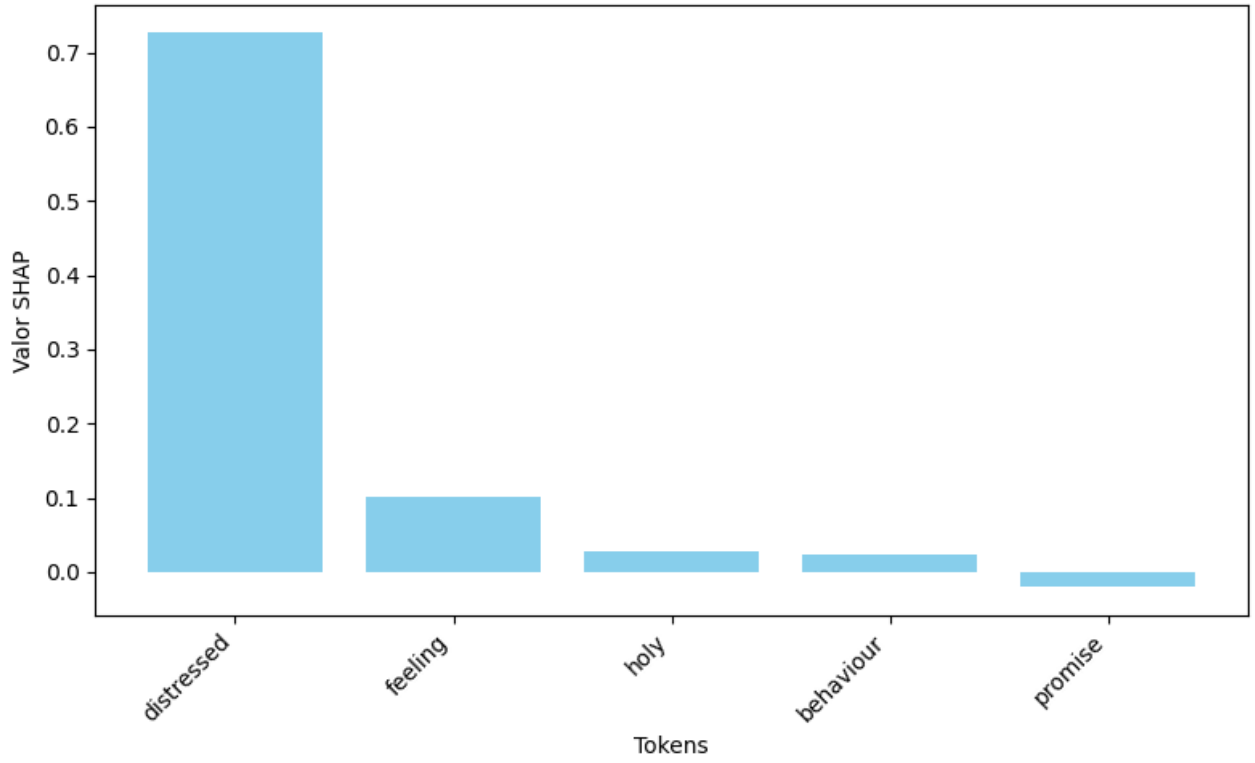


En la clase *Anger*, el token "pissed" destaca con un valor SHAP superior a 0.6, lo que indica una contribución excepcionalmente alta a la activación de esta categoría. Este valor es significativamente mayor que el del resto de tokens de la muestra, lo que sugiere que el modelo asocia este término con un alto grado de certeza emocional. Otros tokens como "measure", "appreciate" y "feeling" tienen contribuciones moderadas, aunque su peso es considerablemente menor. Esto podría reflejar matices lingüísticos que el modelo reconoce como indicadores secundarios de enojo en determinados contextos.

Fear. En la Figura 10 se muestra las características más importantes, el texto utilizado fue: im feeling distressed friends behaviour especially regards past love holy promise psalms

Figura 10

Top 5 características más importantes de Fear

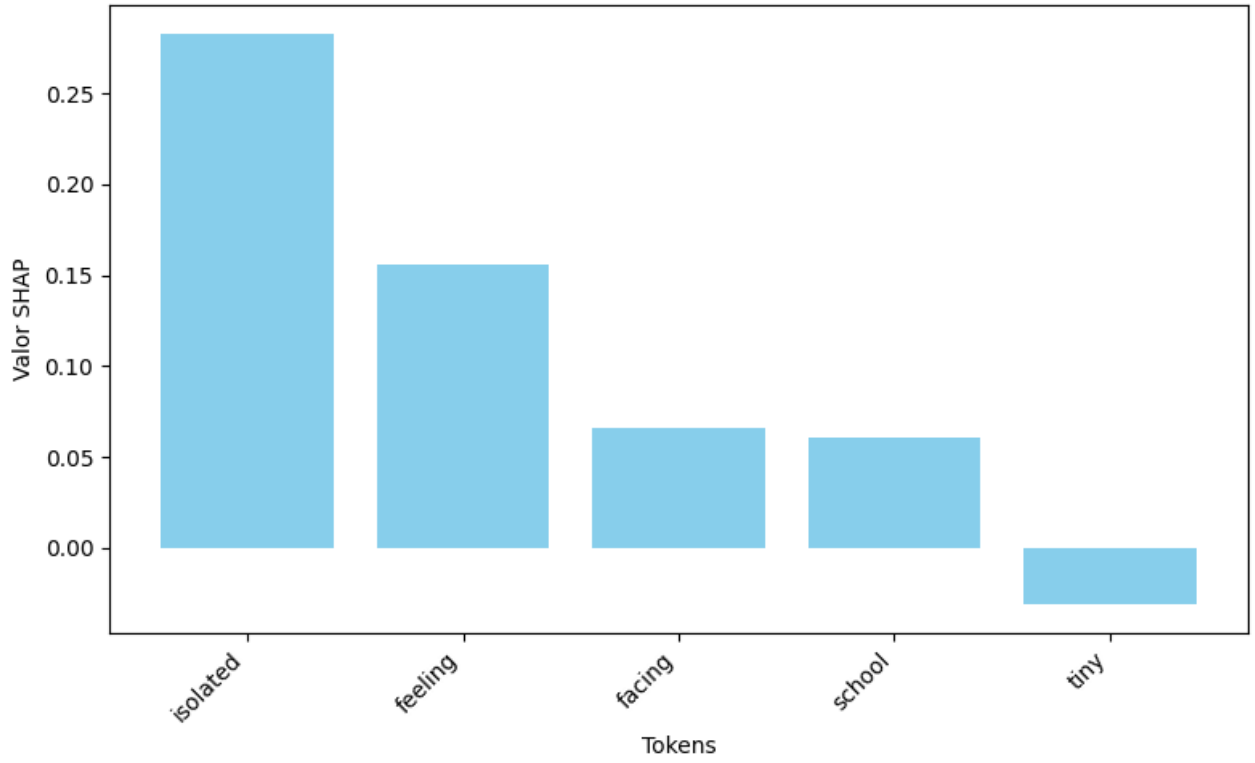


En la clase *Fear*, el token "distressed" sobresale con una importancia superior a 0.7, posicionándose como el principal detonante de la predicción. Palabras como "feeling", "holy", "behaviour" y "promise" presentan contribuciones mucho más bajas, aunque probablemente aporten contexto emocional relevante cuando se combinan. Este patrón sugiere que el modelo ha aprendido a detectar términos clave que comunican vulnerabilidad o amenaza, incluso cuando aparecen en frases complejas.

Sadness. En la Figura 11 se puede observar las características más importantes para esta emoción, el Review utilizado fue: ive going school stuck tiny space facing table pilled books feeling isolated

Figura 11

Top 5 características más importantes de Sadness

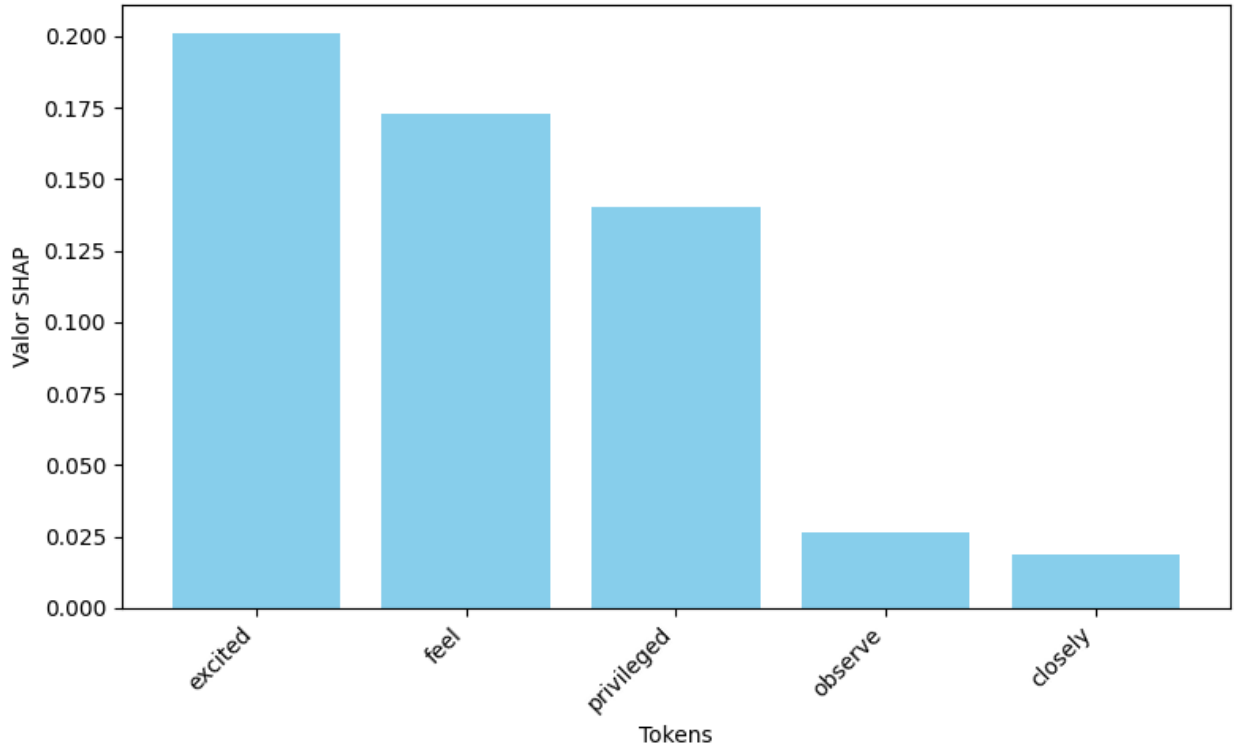


Para la clase *Sadness*, el análisis muestra una distribución más equilibrada de la importancia entre los tokens. El término "isolated" tiene la mayor influencia (≈ 0.28), seguido por "feeling", "facing", "school" y "tiny". A diferencia de las clases anteriores, donde un único token domina la interpretación, aquí se observa una mayor diversidad de palabras relevantes. Esto puede indicar que el modelo reconoce la tristeza como un estado emocional más difuso, donde el sentimiento se construye a partir de una combinación de factores y no solo de palabras emocionalmente cargadas.

Joy. En la Figura 12 se aprecian las características más relevantes, el texto utilizado fue: feel excited privileged observe things closely try portray fleeting moments blink eye

Figura 12

Top 5 características más importantes de Joy

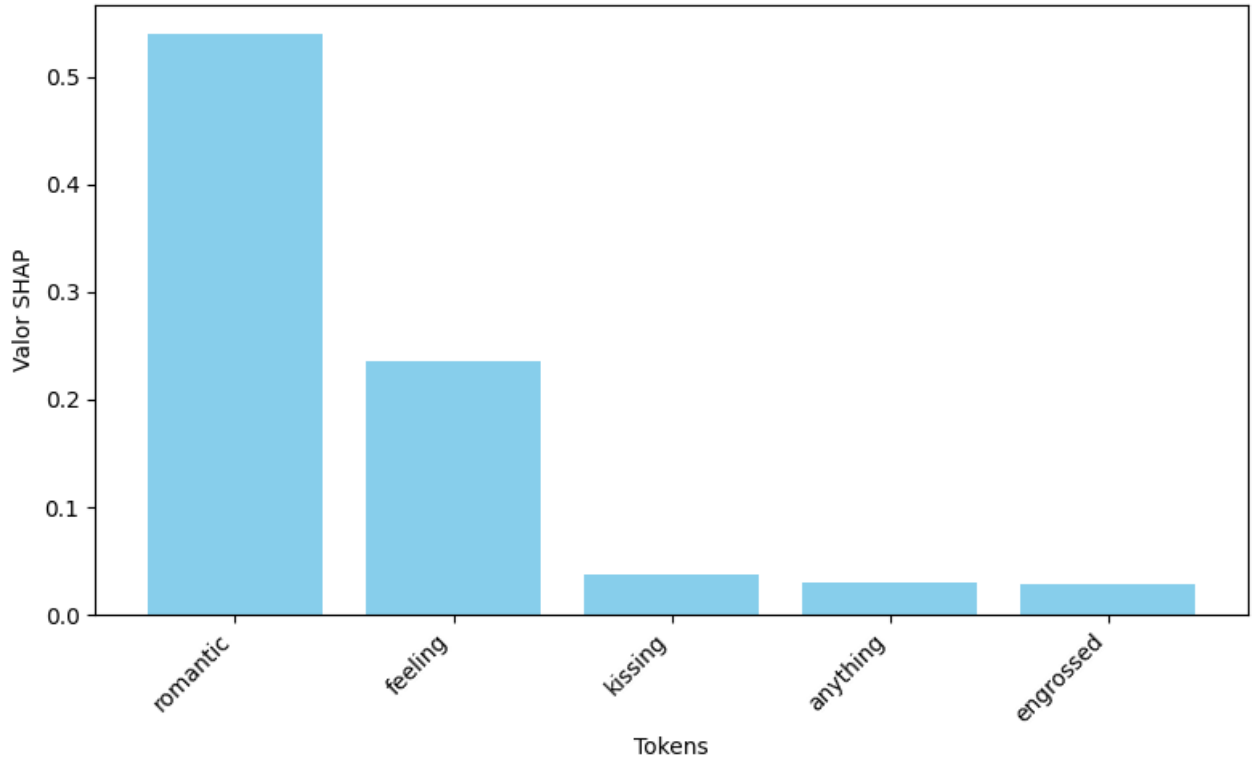


En la clase *Joy,* las palabras más influyentes fueron *excited*, *feel* y *privileged*, seguidas de *observe* y *closely*. Los tres primeros términos transmiten directamente estados emocionales positivos y subjetivos, muy característicos de la categoría de alegría. Esto sugiere que el modelo ha aprendido a asociar patrones léxicos directamente vinculados con experiencias positivas o estados de entusiasmo.

Love. Para esta emoción las palabras más relevantes se pueden apreciar en la Figura 13, el Review utilizado fue: hear anything deeply engrossed sighting could sense feeling kissing cropping romantic head

Figura 13

Top 5 características más importantes de Love



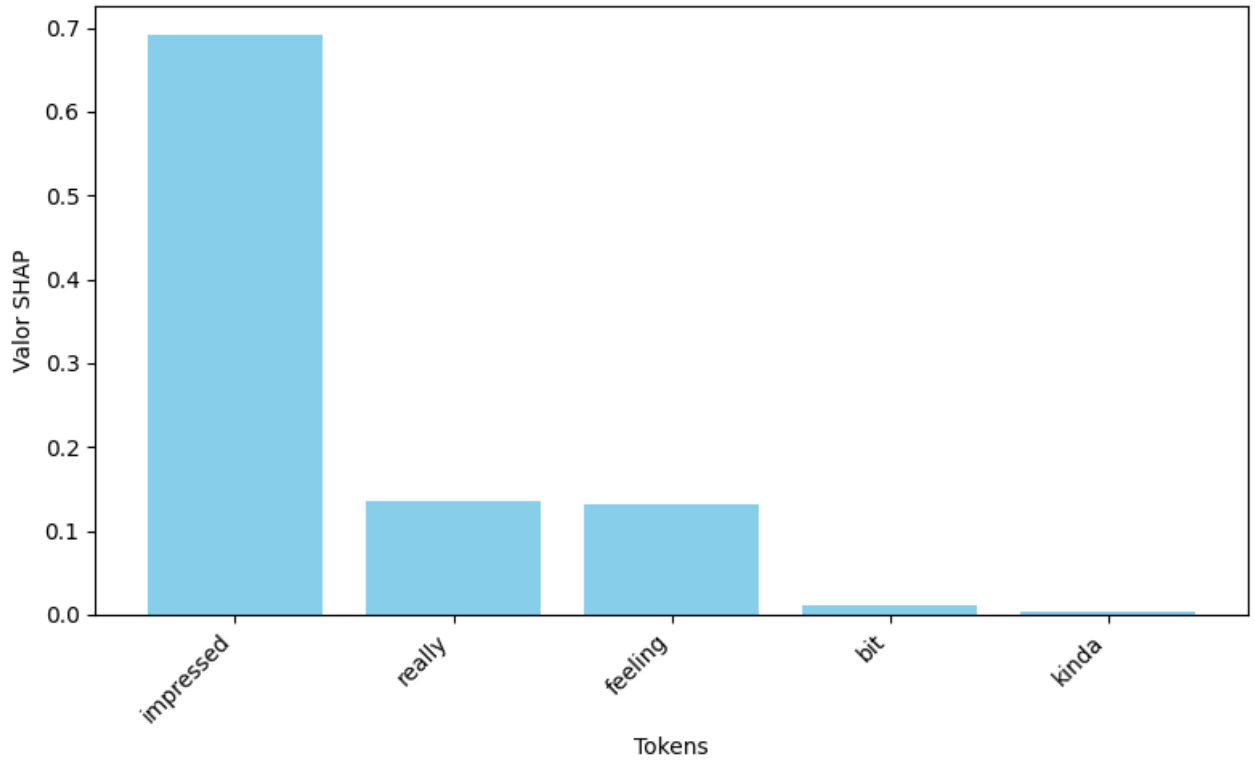
En la clase *Love*, el token con mayor impacto fue *romantic*, acompañado de *feeling*, *kissing*, *anything* y *engrossed*. El predominio de palabras como *romantic* y *kissing* refleja una

conexión directa con el lenguaje afectivo, mientras que *feeling* y *engrossed* también aportan matices emocionales subjetivos. Esto refuerza la idea de que el modelo logra capturar señales afectivas profundas para distinguir esta categoría.

Surprise. En la Figura 14 se puede notar las características más importantes para esta emoción, el Review utilizado fue: remember feeling really impressed living situation bit depressed since dorms kinda suck

Figura 14

Top 5 características más importantes de Surprise



En la clase *Surprise,* los tokens *impressed*, *really*, *feeling*, *bit* y *kinda* fueron los más relevantes. En este caso, el término *impressed* dominó ampliamente el valor SHAP, lo que sugiere que el modelo relaciona directamente esta palabra con la noción de asombro. Las

palabras *bit* y *kinda* aportan matices que suelen acompañar expresiones de sorpresa en contextos informales, mostrando sensibilidad del modelo a patrones contextuales sutiles.

Este análisis cualitativo permite observar cómo el modelo no solo identifica palabras con fuerte carga emocional, sino que también incorpora moduladores y matices propios del lenguaje natural en inglés. El hecho de que *feeling* aparezca de forma recurrente entre las clases sugiere que hay ciertas estructuras comunes que el modelo interpreta como indicadores generales de emoción, mientras que los tokens más específicos permiten la diferenciación entre categorías.

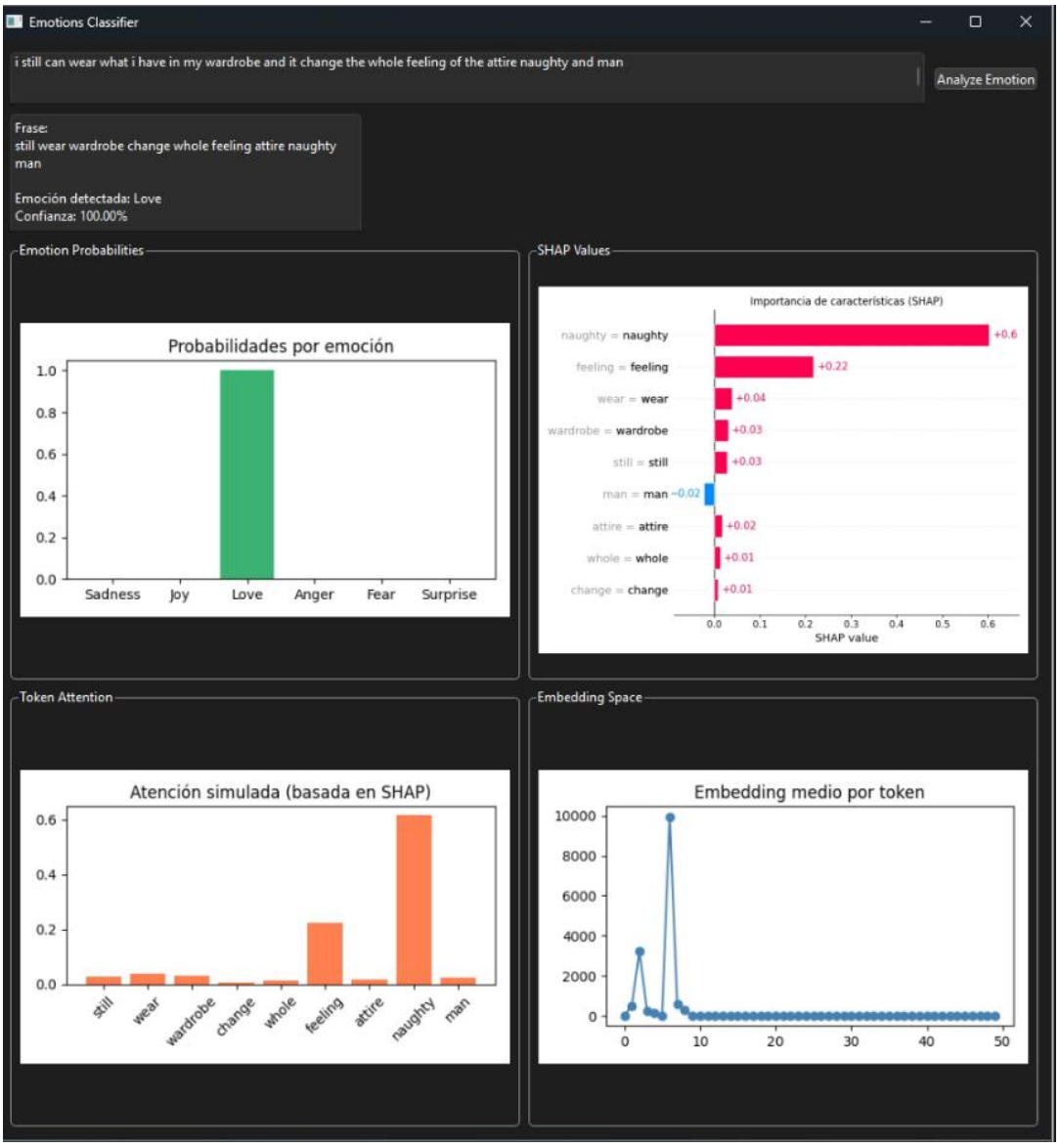
4.2.1.1. Ejecución de interfaz gráfica y pruebas

En el contexto de pruebas realizadas y consolidando la información pertinente en la interfaz gráfica, se ha realizado en detalle con 3 ejemplos a continuación descritos.

Como se muestra en la Figura 15 se obtiene ante el ingreso de texto "i still can wear what i have in my wardrobe and it change the whole feeling of the attire naughty and man” ((Nelgiriyewithana, 2023)

Figura 15

Test 1 de modelo ante input

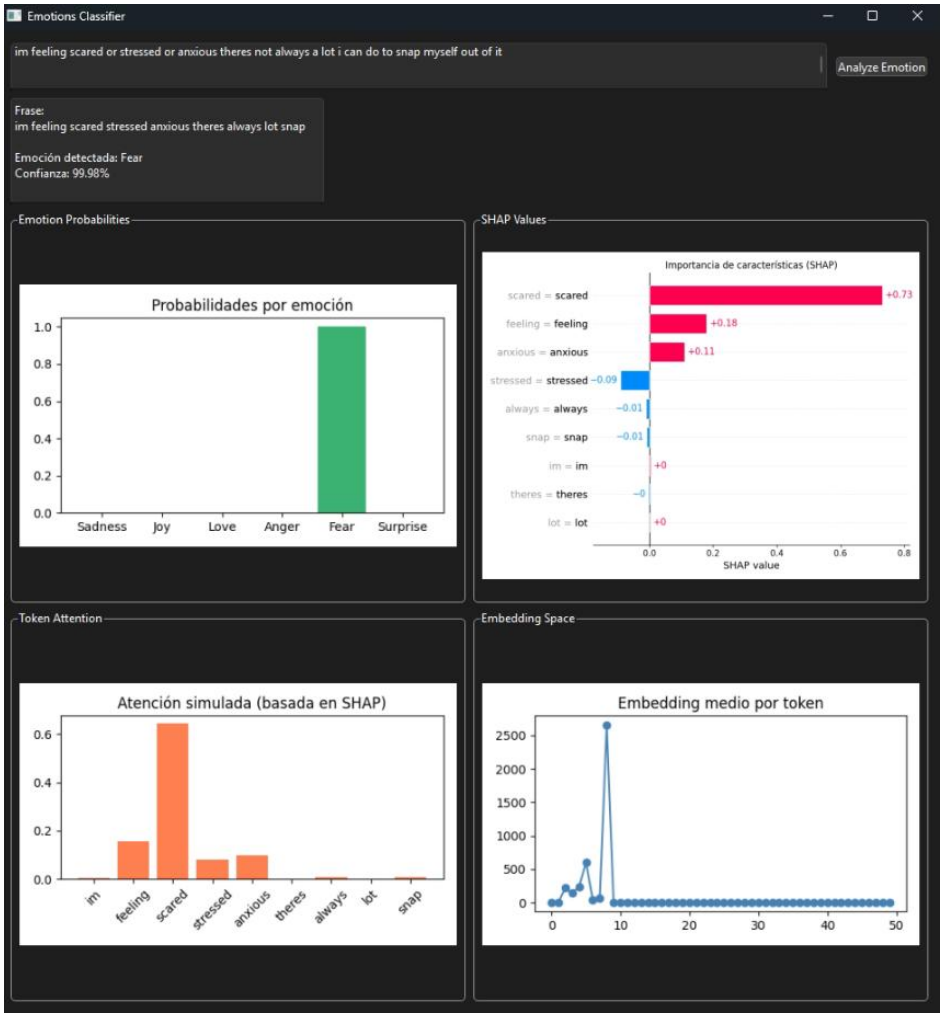


Se obtiene luego de la eliminación de las stop-words, obteniendo “still wear wardrobe change whole feeling attire naughty man”, tal que el modelo procesa el texto resultante y deduce el porcentaje de probabilidad de 100% para la emoción “AMOR (love)”, se visualiza a su vez la influencia de cada token ante la decisión de la clase mediante la gráfica de características SHAP

y la interacción de la importancia de cada token y donde resalta Naughty con gran influencia semántica en los embeddings

En la Figura 16 se puede apreciar ante el ingreso de la expresión "im feeling scared or stressed or anxious theres not always a lot i can do to snap myself out of it" (Nelgiriyewithana, 2023)

Figura 16 *Test 2 de modelo ante input*



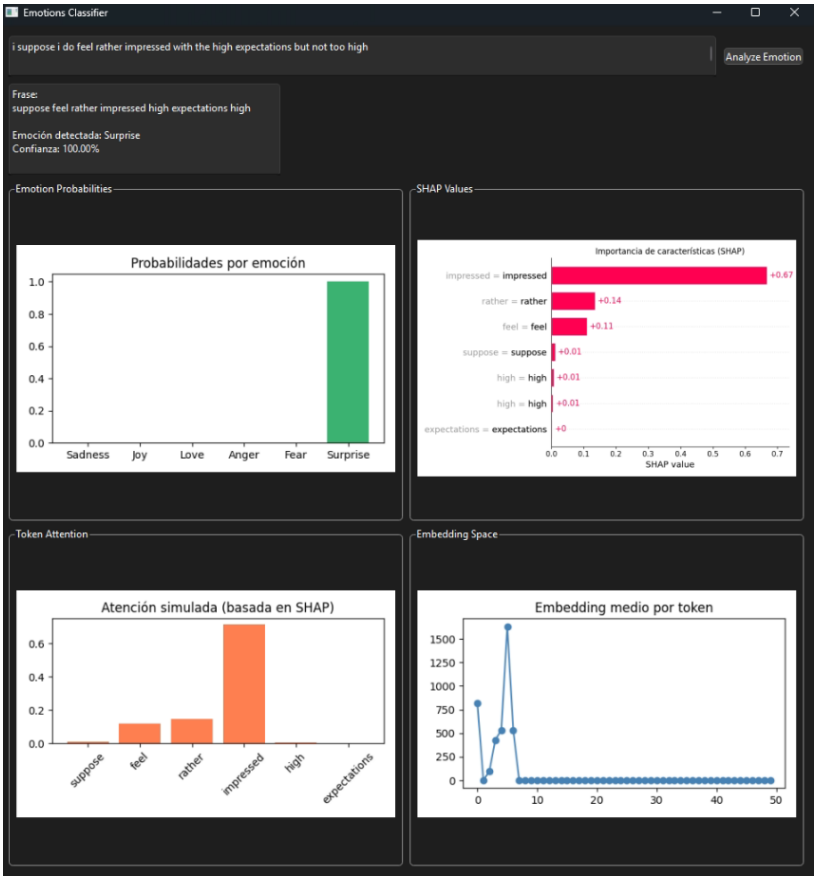
Luego del preprocesamiento del input, se obtiene “im feeling scared, stressed, anxious. There's always a lot. Snap”, una vez clasificado por el modelo, se obtiene con una confianza del 99.98% el tipo de sentimiento “Miedo (fear)”, como resultado se denota dentro de la

interpretación de características SHAP la importancia del token “scared” y su peso sobre la decisión obtenida, así como su influencia en la interacción de los embeddings

En la Figura 17 se puede apreciar ante el ingreso de la expresión " i suppose i do feel rather impressed with the high expectations but not too high” (Nelgiriyewithana, 2023) Una vez preprocesado, se obtiene “suppose feel rather impressed high expectations high”, determinando así la clase con mayor probabilidad “Sorpresa (surprise)” tal que mediant el análisis gráfico se deduce la importancia del token “impressed”. El gráfico de embeddings muestra los token tratando de representar el contexto de la expression incluso antes de llegar a la identidad emocional.

Figura 17

Test 3 de modelo ante input



CAPITULO 5:

5. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

El presente estudio abordó el problema del análisis de sentimientos multiclase en textos en inglés, utilizando un enfoque basado en redes neuronales recurrentes (LSTM bidireccional) y técnicas modernas de interpretabilidad como SHAP. La implementación de un proceso sistemático de búsqueda de hiperparámetros permitió explorar exhaustivamente 2 592 configuraciones distintas, lo cual facilitó la identificación de la arquitectura óptima del modelo. El mejor desempeño se obtuvo con una precisión general del 94 % en el conjunto de prueba, destacando especialmente las clases *Joy* y *Sadness*, mientras que *Surprise* y *Love* presentaron mayores desafíos, principalmente debido al menor número de muestras y a su ambigüedad semántica relativa.

Desde el punto de vista de la interpretabilidad, el análisis con SHAP reveló patrones significativos tanto en la distribución de la importancia por posición de los tokens como en la relevancia de términos individuales por clase. Se observó una alta importancia en las primeras posiciones del texto, aunque con variaciones significativas entre clases. Además, los tokens con mayor impacto permitieron validar que el modelo aprendió relaciones semánticas coherentes con las emociones objetivo, lo cual fortalece la confianza en sus predicciones.

Entre las principales conclusiones del estudio se destacan:

Importancia de la depuración de datos*:* El uso de una versión limpia del conjunto de datos fue determinante para lograr un rendimiento robusto y consistente del modelo.

Eficiencia del ajuste fino: La evaluación exhaustiva de combinaciones de hiperparámetros resultó clave para alcanzar niveles óptimos de desempeño, demostrando la

necesidad de un proceso riguroso de validación.

Utilidad de embeddings preentrenados: La incorporación de vectores FastText enriqueció la representación semántica y facilitó la generalización del modelo.

Aporte de la interpretabilidad*:* La aplicación de SHAP no solo permitió analizar la lógica interna del modelo, sino también generar explicaciones comprensibles sobre las decisiones, lo cual es crucial en aplicaciones sensibles o críticas.

5.2.Recomendaciones

Con base en los resultados obtenidos, se sugieren las siguientes recomendaciones para futuras investigaciones o implementaciones:

Profundizar en el análisis de clases minoritarias*:* Incorporar técnicas de balanceo de clases (como *oversampling* o generación de texto sintético) podría mejorar el desempeño en categorías con menor representación como *Surprise* o *Love*.

Explorar arquitecturas híbridas*:* La combinación de redes LSTM con mecanismos de atención o transformadores podría mejorar aún más la captura de dependencias complejas y mejorar la precisión.

Ampliar el análisis de interpretabilidad*:* Se recomienda extender el uso de SHAP u otras técnicas (como LIME o attention-based saliency) a contextos de mayor longitud o comparar su efectividad en distintos dominios textuales.

Implementación en entornos reales*:* Para validar la utilidad práctica del modelo, sería conveniente probar su desempeño en aplicaciones concretas como análisis de opiniones, monitoreo de redes sociales o atención al cliente.

Evaluación multilingüe y cultural*:* Dado que las expresiones emocionales varían según el idioma y el contexto cultural, futuras versiones del modelo podrían ser entrenadas y evaluadas

con datos en otros idiomas o regiones.

REFERENCIAS

Blanquicett, L., & Murillo, L. (2022). Machine Learning. *Revisata Sistemas*. https://doi.org/10.29236/sistemas.n165a6

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5, 135–146.

Cachay, A. (2024). Evaluation on embeddings application for spanish automatic text clustering. *Revista chilena de ingeniería*, 1-13. https://doi.org/https://orcid.org/0000-0002-0096-0920

Calvo, A. (2020). *Análisis de sentimientos y emociones en redes sociales usando ML .* Universidad de Valladolid .

Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2012). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, 145–150. https://doi.org/https://doi.org/10.1109/MIS.2013.30

Cardoso, A., Talame, L., Amor, M., & Neil, C. (2019). *Minería de Opniones; Análisis de Sentimientos en una Red Social.* Universidad Nacional de la Plata.

Cate, M. (2025). Interpretability Techniques in Machine Learning. https://doi.org/https://www.researchgate.net/publication/390873263

Cocon, F., Pérez, D., Pérez, A., Zavaleta, P., Barradas, U., & Gomez, R. (2023). Web scraping: Uso de plataformas de extracción de datos aplicadas a un sitio web sobre profesiones en México. *Revista Ibérica de Sistemas y Tecnologías de Información*(52), 61-74. https://doi.org/10.17013/risti.52.61–73

D'Andrea, A., Ferri, F., Grifoni, P., & Guzzo, T. (2015). Approaches, Tools and Applications for Sentiment Analysis Implementation. *International Journal of Computer Applications, 125*(3), 26-35. https://doi.org/10.5120/ijca2015905866

Dauda, H., & Umar, M. (2022). *Clasificación de sentimientos: revisión de métodos de vectorización de texto: Bag of Words, Tf-Idf, Word2vec y Doc2vec.*

Dewi, C., Tsai, B., & Chen, R. (2022). Shapley Additive Explanations for Text Classification and Sentiment Analysis of Internet Movie Database. *Communications in Computer and Information Science*, 2-13. https://doi.org/10.1007/978-981-19-8234-7_6

Domor, I., Swart, T., & Obaido, G. (2024). Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. *MDPI, 15*, 1-34. https://doi.org/https://doi.org/10.3390/info15090517

Fernández, E. A., & Ruiz, J. M. (2020). Redes neuronales artificiales y su papel en el análisis de emociones: Un estudio de caso. *Neural Networks and Natural Language Processing Journal, 3*(9), 77-89. https://doi.org/https://doi.org/10.6789/nnlnp.2020.0092

Gabbard, J. (2018). Machine Learning from Scratch: Stochastic Gradient Descent and Adam Optimizer. https://doi.org/https://www.mit.edu/~jgabbard/assets/18085_Project_final.pdf

García, J. A., & López, M. P. (2020). El análisis de emociones en redes sociales: Un estudio sobre Twitter. *Revista de Psicología Digital y Comportamiento Social, 3*(18), 45-62. https://doi.org/https://doi.org/10.1234/rpdcs.2020.0345

Gómez, P. S., & Hernández, D. L. (2022). Interpretabilidad en redes neuronales: SHAP y su aplicación en el análisis de emociones en redes sociales. *International Journal of Explainable AIInternational Journal of Explainable AI, 2*(3), 123-137. https://doi.org/https://doi.org/10.7890/ijexai.2022.0345

Graves, A. (2005). Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition. *Springer-Verlag*, 799–804.

IMB. (11 de Agosto de 2024). *What is NLP (natural language processing)?* . Obtenido de https://www.ibm.com/think/topics/natural-language-processing#:~:text=NLP%20enables%20computers%20and%20digital,machine%20learning%20and%20deep%20learning.

Imrus, S., & Kang, D.-K. (2023). A Review on Dropout Regularization Approaches for Deep. *MDPI, 12*, 1-23. https://doi.org/https://doi.org/10.3390/electronics12143106

Jawad, E. (2023). THE DEEP NEURAL NETWORK-A REVIEW. *JRDO Journal*. https://doi.org/10.53555/m.v9i9.5842

Jurafsky, D. (2025). *Neural Networks.* https://doi.org/https://web.stanford.edu/~jurafsky/slp3/7.pdf

Kaur, J., & Kaur, P. (2018). A Systematic Review on Stopword Removal Algorithms. *International Journal on Future Revolution in Computer Science & Communication Engineering, 4*(4), 207-2010.

Kumar, P., Mohan, B., & Altalbe, A. (2024). The social media sentiment analysis framework: deep learning for sentiment analysis on social media. *International Journal of Electrical and Computer Engineering (IJECE), 14*(3), 3394-3405. https://doi.org/10.11591/ijece.v14i3.pp3394-3405

Kunjal, N., & Kumbong, H. (2024). MT-BERT: Fine-tuning BERT for Downstream Tasks Using Multi-Task Learning. *Stanfor*, 1-9. https://doi.org/https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1234/final-reports/final-report-169730024.pdf

Kyaw, T. (20215). Storage database in cloud processing . *Computer Research and Modeling, 7*(3), 493-498. https://doi.org/10.20537/2076-7633-2015-7-3-493-498

Lazo, A., & Rodas, E. (2024). *Análisis de sentimientos en las redes sociales de las universidades de Cuenca.* Universidad Politécnica Salesiana.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining.* Morgan &. https://doi.org/https://www.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf

Lundberg, S. M., & Lee, S. I. (2017). A Unified Approach to Interpreting Model PredictionsA Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, 30.

Maldonado, E. (2022). *Análisis de sentimientos en la red social Twitter mediante el Procesamiento de Lenguaje Natural.* Universidad Nacional de Chimborazo.

Management Solutions. (2018). *Machine Learning.* Management Solutions.

Martínez, A. R., & Sánchez, L. T. (2021). Aplicación de algoritmos de aprendizaje automático en la clasificación emocional de tuits: Un enfoque integral. *Revista Internacional de Inteligencia Artificial y Computación, 4*(12), 200-218. https://doi.org/https://doi.org/10.9876/riiac.2021.0912

Medina, J. (2020). *Implementación y explotación de un sistema de APIs para el consumo de KPIs.* Universidad de Valladolid.

Muñiz, C. (2018). Deep Learning en el Análisis de Sentimientos. *ResearchGate*(1-30).

Nandwani, P., & Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Springer*, 80-100. https://doi.org/10.1007/s13278-021-00776-6

Nelgiriyewithana. (2023). *Kaggle*. Obtenido de https://www.kaggle.com/datasets/nelgiriyewithana/emotions/data

Palomino, M. A., & Aider, F. (2022). Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis. *Applied Sciences*, 12(17). https://doi.org/https://doi.org/10.3390/app12178765

Pérez, R. S., & Gómez, F. D. (2019). Técnicas de procesamiento de lenguaje natural para el análisis de emociones en plataformas sociales. *Journal of Machine Learning Applications, 2*(7), 123-135. https://doi.org/https://doi.org/10.5678/jmla.2019.0453

Picard, R. W. (1997). *Affective computing.* MIT Press.

Piccioni, L., Guerreiro, M., & Pereira, E. (2023). Multilayer Perceptron. En *Introduction to Computational Intelligence* (pág. 105). IEEE Computational Intelligence Society Open Book.

Prechelt, L. (1998). Early Stopping — But When? (Springer, Ed.) *In Neural Networks: Tricks of the Trade*, 55-69.

Qiu, J. (2024). An Analysis of Model Evaluation with Cross-Validation: Techniques, Applications, and Recent Advances. *Proceedings of Finance in the Age of Environmental Risks and Sustainabilit*, 69-73. https://doi.org/10.54254/2754-1169/99/2024OX0213

Snachez, F. (2019). *Sentiment and emotion analysis in social networks: modeling and linking data, affects and people.* Universidad Politécnica de Madrid.

Syaamantak, D., Kumar, S., & Basu, A. (2019). Mining multiple informational text structure from text data. *Procedia Computer Science, 167*(2020), 2212-2222. https://doi.org/10.1016/j.procs.2020.03.273

Torres, M. J., & Ramírez, V. C. (2021). Redes neuronales recurrentes LSTM para el análisis emocional de tuits: Una comparación con modelos tradicionales. *Revista de Aprendizaje Automático y*

Lenguaje Natural, 1(15), 45-59. https://doi.org/https://doi.org/10.3456/raln.2021.0548https://doi.org/10.3456/raln.2021.0548

Trujillo, A. (2018). *Evaluación de técnicas de procesamiento de lenguaje natural para textos cortos en lenguaje español.* Universidad del Azuay.

Willie, A. (2024). Explainable AI (XAI): Enhancing Transparency and Interpretability in Machine Learning Models. 1-10.

Zachar, P. (2021). The Classification of Emotion and Scientific Realism. *Auburn University Montgomery*, 120-140. https://doi.org/10.1037/h0091270

APÉNDICE. Repositorio del Proyecto

Se pone a disposición el siguiente repositorio de GitHub, donde se encuentra todo el material complementario al presente trabajo, incluyendo:

- El script que contiene el procesamiento, entrenamiento y evaluación del modelo de clasificación emocional.
- El código fuente de la aplicación interactiva, que permite interpretar emociones en tiempo real y visualizar la importancia de cada token en la decisión del modelo.
- **Enlace:** Repositorio del proyecto: https://github.com/TON7T/Multiclass-Sentiment-LSTM-SHAP

Este recurso está disponible para consulta, réplica del experimento y futuras extensiones del proyecto.