Universidad Michoacana de San Nicolás de Hidalgo

Facultad de Ciencias Físico Matemáticas

Detección de sexismo y misoginia en tweets en español con aprendizaje automático

T E S I S

que para obtener el título de:

licenciada en ciencias físico matemáticas

P R E S E N T A :

Kathia Rangel Pompa

ASESORES

Dra. Karina Mariela Figueroa Mora
Dr. Luis Miguel García Velázquez

Ciudad Universitaria, Morelia, Michoacán.
Julio 2024

Agradecimientos

Quiero dar gracias a toda mi familia y amigos por siempre apoyarme en esta etapa de mi vida, la cual habría sido imposible de superar sin ellos a mi lado.

A mis padres, por haberme dado todo lo que necesitaba y más.

A Jorge, por siempre proporcionarme calma en los momentos de estrés y disfrutar junto a mí los momentos felices.

Al Dr. Luis Miguel García, por poner en rumbo este proyecto con su experiencia y consejos.

A mi asesora, la Dra. Karina Figueroa, porque su dedicación por enseñar me llevó a descubrir mi propia pasión, la cual se consolidó junto con este trabajo.

Resumen

Este proyecto tiene como objetivo la creación de un modelo de Aprendizaje Automático capaz de detectar sexismo y misoginia en tweets en español. Además, busca comparar los resultados obtenidos con distintos tipos de modelos comunes y proponer técnicas para trabajar con bases de datos limitadas.

La metodología del trabajo incluyó la recopilación y unificación de diferentes bases de datos existentes de tweets en español, formando así un corpus consolidado. Adicionalmente, se propuso aumentar este corpus añadiendo una base de datos de tweets en inglés relacionados con la detección de sexismo, los cuales fueron traducidos utilizando el API de Google Translator. Finalmente, se aplicaron varios modelos de Aprendizaje Automático a ambas bases de datos (la recopilación inicial y la aumentada con tweets en inglés) para comparar su desempeño.

Conceptos clave: Sexismo, Misoginia, Lenguaje de Odio, Aprendizaje Automático, Procesamiento de Lenguaje Natural.

Abstract

This project aims to create a machine-learning model capable of detecting sexism and misogyny in Spanish tweets. It also seeks to compare the results obtained with different common models and propose techniques for working with limited databases.

The methodology of this work included the collection and unification of different existing databases of Spanish tweets, thus forming a consolidated corpus. Furthermore, it was proposed that this corpus be augmented by adding a database of English tweets related to the detection of misogyny, which were translated using the Google Translator API. Finally, various Machine Learning models were applied to both databases (the initial collection and the augmented one with English tweets) to compare their performance.

Índice general

Capítulo 1

Introducción

1.1. Contexto

El uso de las redes sociales ha propiciado una interacción internacional entre las personas, para poder expresarse y compartir sus opiniones sobre cualquier suceso en el mundo. Sin embargo, con millones de usuarios compartiendo mensajes diferentes cada día desde el anonimato, las plataformas se enfrentan a un grave aumento del uso del lenguaje de odio en línea.

Coloquialmente, el lenguaje de odio hace referencia a cualquier discurso ofensivo dirigido a un grupo o individuo debido a características inherentes (raza, género, religión, orientación sexual, etcétera).

El lenguaje de odio ha encontrado una herramienta para propagarse y viralizarse en plataformas como X (antes Twitter), una de las redes sociales más usadas a día de hoy, con 619 millones de usuarios activos en el mundo[12].

Entre los distintos tipos de lenguaje de odio es importante destacar la violencia de género. La definición más aceptada de violencia de género es la propuesta por la ONU en 1995: «Todo acto de violencia sexista que tiene como resultado posible o real un daño físico, sexual o psíquico, incluidas las amenazas, la coerción o la privación

arbitraria de libertad, ya sea que ocurra en la vida pública o en la privada».

Un estudio realizado en 2017 por Amnistía Internacional reveló que el 23 % de mujeres encuestadas en ocho países han experimentado abusos o acoso en Internet, y que el 46 % de las encuestadas que habían sufrido de éstos dijeron que eran de naturaleza misógina o sexista[9]. Estos ataques amenazan directamente contra la libertad de expresión y tienen como objetivo seguir perpetuando conductas machistas, mientras que silencian mujeres haciéndolas sentir inseguras y humilladas, por lo cual, las empresas tienen la responsabilidad de atender a los abusos misóginos en sus plataformas.

"El peligro especial de los abusos en Internet es la rapidez con la que pueden proliferar: un tuit insultante puede convertirse en minutos en un aluvión de odio focalizado. Las empresas de redes sociales deben empezar a tomarse en serio realmente este problema", afirma Azmina Dhrodia, investigadora de Tecnología y Derechos Humanos en Amnistía Internacional[9].

1.2. Problema

Controlar y reducir el discurso de odio, en todas sus formas, es un problema que muchos entornos y plataformas virtuales están enfrentando actualmente, y que se vuelve más difícil a medida que el tamaño de esas plataformas crece constantemente. También es uno de los problemas más difíciles en cuanto al mantenimiento de plataformas en línea, ya que los límites de lo que es y no es discurso de odio son específicos de cada región y dependen de diversos factores sociales, históricos y culturales.

Con el desarrollo del Procesamiento de Lenguaje Natural (PLN), se han realizado múltiples trabajos en la detección de lenguaje de odio en años recientes. El inglés, siendo el idioma más usado internacionalmente, ha sido el foco de atención en la mayoría de modelos de detección de lenguaje de odio, por lo que el desarrollo en otros idiomas es significativamente menor, como en el español.

Las características específicas del lenguaje de odio, como modismos y palabras clave, son diferentes en cada idioma y, mientras los modelos en inglés son muy capaces en su campo, tienen problemas para adaptarse a otros idiomas.

El sesgo de investigación en lenguas distintas al inglés no sólo indica un menor número de sistemas de Procesamiento de Lenguaje Natural, sino también una falta de bases de datos significativa. Esta carencia de bases de datos en español dificulta el desarrollo y la implementación de herramientas eficaces para la detección automática del discurso de odio en este idioma. Las pocas bases de datos existentes a menudo son limitadas en alcance, diversidad y calidad, lo que reduce la efectividad de los algoritmos entrenados con ellas.

1.3. Justificación

La falta de bases de datos adecuadas en español representa un obstáculo significativo en la lucha contra el discurso de odio en comunidades de habla hispana. Por lo tanto, es crucial investigar y abordar las limitaciones actuales de las bases de datos en español para la detección del discurso de odio.

Este trabajo busca combinar distintos modelos de Procesamiento de Lenguaje Natural y clasificación de texto para la detección de sexismo y misoginia en español, así como proponer estrategias para mejorar las bases de datos existentes y aumentar la eficacia de los modelos implementados.

Capítulo 2

Estado del arte

El presente estado del arte tiene el propósito de indicar el panorama general en la investigación de detección y clasificación del lenguaje de odio (en particular misoginia y sexismo) con modelos de Aprendizaje Automático, así como revisar los trabajos recientes en la investigación con bases de datos en español, los cuales fueron de suma importancia para el desarrollo de este proyecto.

2.1. Detección automática del lenguaje de odio

El Procesamiento del Lenguaje Natural es un campo de la Inteligencia Artificial que se enfoca en la interacción entre las computadoras y el lenguaje humano. Su objetivo principal es permitir a las máquinas entender, interpretar y generar lenguaje humano de manera natural.

Por otro lado, el análisis de sentimientos se ha convertido en un área crítica de investigación debido al aumento de contenido ofensivo en plataformas digitales. El PLN juega un papel fundamental en este contexto al proporcionar herramientas y técnicas para identificar la carga emocional negativa asociada con el lenguaje de odio, ayudando a identificar expresiones discriminatorias.

Una de las investigaciones fundamentales para avanzar en la comprensión y la mi-

tigación del lenguaje de odio en redes sociales ha sido el trabajo de Waseem en 2016. En este estudio, exploraron características predictivas para la detección de discurso de odio en Twitter, analizando cómo ciertos símbolos o palabras pueden indicar la presencia de lenguaje discriminatorio. Encontraron que la distribución geográfica de los usuarios y la longitud de las palabras tienen de cero a muy poco efecto positivo en el rendimiento, a excepción del género[24]. Es decir, usar el género de los usuarios como una característica de los datos del corpus lleva a una mejora en el desempeño, pues los hombres tienden a publicar la mayoría de los tweets clasificados como lenguaje de odio. Sin embargo, como Twitter no provee esta información, no es un recurso que sea fácil de añadir a las bases de datos actualmente.

Otro de los trabajos destacados en la detección automática del lenguaje de odio es el de Davidson en 2017, donde remarca la importancia de separar el lenguaje de odio de otras instancias de lenguaje ofensivo[2], ya que una falla clave en gran parte de los trabajos anteriores es que el lenguaje ofensivo se etiqueta erróneamente como discurso de odio debido a una definición demasiado amplia. Es por este mismo motivo que incluso en el etiquetado manual, los términos racistas u homofóbicos son considerados lenguaje de odio, pero las palabras sexistas y discriminantes hacia las mujeres son sólo vistas como ofensivas.

Como panorama general, Jahan en 2023 realizó una revisión sistemática del estado del arte en la detección automática de discursos de odio utilizando Procesamiento del Lenguaje Natural y tecnologías de Aprendizaje Profundo. La revisión analiza la literatura de los últimos diez años, enfocándose en la terminología, los métodos empleados y las arquitecturas de Aprendizaje Profundo más relevantes.

Los resultados del estudio muestran que el número de publicaciones sobre la detección del lenguaje de odio ha aumentado significativamente en los últimos años y destaca las arquitecturas del Aprendizaje Profundo, como BERT, como un área de oportunidad. De igual manera, señalan que el principal desafío en la detección del

lenguaje de odio es la falta de base de datos que incluyan variabilidad en las distintas culturas y lenguajes[10].

2.2. Enfoques actuales en la detección de sexismo y misoginia en español

La identificación y clasificación automática de la misoginia y el sexismo en redes sociales ha ganado relevancia en la última década, lo que ha llevado a la realización de diversos estudios que han abordado el problema desde diferentes perspectivas metodológicas y con variados enfoques de Procesamiento de Lenguaje Natural y Aprendizaje Automático.

En 2018, S. Téllez et al. desarrollaron Micro Clasificación de Texto (μTC), un sistema innovador para la clasificación de textos con un enfoque minimalista. Abordaron el problema de crear clasificadores de texto que funcionen independientemente tanto del dominio como del idioma, sin nada más que un conjunto de entrenamiento para aprender[22]. El modelo se compone de una serie de transformaciones de texto simples, tokenizadores, un conjunto de esquemas de ponderación, junto con una Máquina de Vectores de Soporte (SVM) como clasificador para producir una clasificación de texto efectiva.

Los autores realizaron una comparación experimental exhaustiva del rendimiento de μTC frente a métodos de vanguardia en 30 conjuntos de datos diferentes, que abarcan tareas como la clasificación de temas, detección de spam, análisis de sentimientos y perfiles de usuarios. En términos de precisión, μTC obtuvo el mejor desempeño en 20 de estos conjuntos de datos y resultados competitivos en los restantes. Esta robustez demuestra su eficacia y flexibilidad en diversas aplicaciones de clasificación de textos, incluso en textos informales y con errores ortográficos.

El trabajo de García-Díaz en 2020 se centra en la identificación automática de misoginia en español. El estudio propone la creación de una nueva base de datos, debido a las limitaciones que posee las bases de datos existentes como AMI 2018 de IberEval[6]. El corpus propuesto se compuso de tweets clasificados en no misóginos y misóginos, cuyo contenido abarcaba tres categorías (violencia contra mujeres relevantes; español europeo vs español latinoamericano; desacreditación, dominancia, acoso sexual y estereotipos).

Para preparar los datos antes de alimentar los modelos de Aprendizaje Automático, se llevaron a cabo varios pasos de preprocesamiento, incluyendo limpieza de texto, tokenización, lematización y estematización, conversión a minúsculas y eliminación de palabras vacías. Una vez procesados los datos, el estudio implementó y evaluó diversos modelos de Aprendizaje Automático, como Random Forest, Sequential Minimal Optimization (SMO) y Lineal Support Vector Machine (LSVM), donde se obtuvieron los mejores resultados con Sequential Minimal Optimization (SMO).

2.2.1. AMI IberEval

La tarea Identificación Automática de Misoginia (AMI) fue propuesta en IberEval 2018, cuyo objetivo principal fue distinguir contenido misógino del no-misógino, categorizar el comportamiento misógino y clasificar el target de un tweet escrito en español o en inglés[5]. El propósito de la competencia constaba de dos tareas:

- **Tarea A:** Identificación de misoginia. Realizar una clasificación binaria de los tweets, en misóginos y no misóginos.
- **Tarea B:** Comportamiento misógino y clasificación de los objetivos. Identificación del tipo de misoginia en el texto y su destinatario. Los tipos de misoginia se dividían en 5 categorías (estereotipo y cosificación, dominación, revictimización, acoso sexual y amenazas de violencia y desacreditación) y dos tipos de targets (activo, si el texto tenía un objetivo específico, y pasivo, si era dirigido a las mujeres en general).

Se emplearon tres enfoques con el fin de recolectar datos en inglés y en español:

- Buscar tweets utilizando un conjunto de palabras clave representativas, como p*rra, p*ta, z*rra.
- Monitorear cuentas de víctimas potenciales, como mujeres feministas públicamente.
- Revisar el historial de misóginos identificados, es decir, que declararon explícitamente odio contra las mujeres en sus perfiles de Twitter (ahora X).

La fase de etiquetado de los datos involucró dos pasos: primero, se compuso un *estándar de oro*[1] que fue etiquetado por dos anotadores expertos, cuyos casos de desacuerdo fueron resueltos por un tercer colaborador experimentado. En segundo lugar, los tweets restantes fueron etiquetados a través de un enfoque de votación mayoritaria por colaboradores externos en la plataforma CrowdFlower[2]. El estándar de oro se utilizó para el control de calidad de los juicios a lo largo del segundo paso[5].

Para la competencia se proporcionó un corpus en español y otro en inglés, donde cada registro de las bases de datos contenía los siguientes campos:

- **id**: Indica un identificador único del tweet.
- **text**: Representa el texto del tweet.
- **misogynous**: Define si el tweet es misógino o no. Toma el valor 1 si el tweet es misógino y 0 si no lo es.
- **misogyny_category**: Denota el tipo de comportamiento misógino, tomando los siguientes valores:
    - *stereotype*: Indica la categoría "Estereotipo y Cosificación".

[1] Un *estándar de oro* es un conjunto de datos de referencia etiquetado manualmente por expertos, utilizado para evaluar y validar la precisión de los modelos de Aprendizaje Automático.

[2] CrowdFlower (más tarde Figure Eight) era una empresa de Aprendizaje Automático e Inteligencia Artificial con humanos en el circuito con sede en San Francisco.

  - *dominance*: Indica la categoría "Dominación".
  - *derailing*: Indica la categoría "Revictimización".
  - *sexual_harassment*: Indica la categoría "Acoso Sexual y Amenazas de Violencia".
  - *discredit*: Indica la categoría "Desacreditación".
  - *0*: Si el tweet no es misógino.

- **target**: Denota el sujeto del tweet misógino, tomando los siguientes valores:
  - *active*: Indica un objetivo específico (individual).
  - *passive*: Indica posibles receptores (genérico).
  - *0*: Si el tweet no es misógino.

En esta competencia, los enfoques ganadores incluyeron Máquinas de Vectores de Soporte (SVM) y Bolsa de Palabras (Bag of Words).

2.2.2. MeTwo

Rodríguez-Sánchez en 2020 propuso una nueva tarea que tiene como objetivo comprender y analizar cómo se expresa el sexismo, desde el odio o la violencia explícitos hasta expresiones sutiles, en conversaciones en línea. Con este fin, desarrollaron el primer conjunto de datos de expresiones y actitudes sexistas en Twitter en español (MeTwo)[20].

Para iniciar el conjunto de datos, se recopilaron una serie de expresiones y términos populares comúnmente utilizados para subestimar el papel de las mujeres en nuestra sociedad, fomentar el acoso hacia ellas o limitar su libertad de expresión[20]. Por cada término seleccionado, se recolectaron 150 tweets, que podían ser sexistas o no sexistas, dando como resultado un corpus final compuesto de 3600 tweets. Cada entrada de la base de datos estaba compuesta de la siguiente manera:

- **status_id:** Identifica de manera única el tweet.

- **text:** Contiene el texto del tweet.
- **categoría:** Identifica la expresión sexista y comportamiento del tweet. Toma los valores:
  - *sexist:* Tweets que subestiman a las mujeres como resultado de su género, independientemente de la faceta de las mujeres que se critique, y sin importar la intencionalidad y la violencia.
  - *non-sexist:* Tweets sin connotaciones sexistas. En esta categoría, se pueden encontrar tweets xenófobos u ofensivos, pero que no subestiman a las mujeres por razón de su género.
  - *doubtful:* Tweets que podrían ser sexistas dependiendo del contexto, el cual no puede inferirse del texto en el tweet.

A diferencia de investigaciones previas que se han enfocado principalmente en la detección de misoginia explícita, este trabajo propone una tarea más amplia que incluye la detección de sexismo en sus diversas formas, desde expresiones explícitas de odio, como lo es la misoginia, hasta estereotipos sutiles que, aunque frecuentemente pasados por alto, son extremadamente dañinos para las mujeres y la sociedad en general[20].

Además, los resultados del estudio demuestran que es posible detectar estas conductas utilizando enfoques de Aprendizaje Profundo y que la misoginia y el odio explícito son más fáciles de identificar que el sexismo sutil debido a su menor dependencia del contexto, por lo cual, modelos que han sido entrenados en una base de datos que incluyen diversas formas de sexismo podrían ser más capaces de generalizar a otras expresiones, como lo es la misoginia.

2.2.3. EXIST IberLEF

En 2021, IberLEF desarrolló la tarea compartida sEXism Identification in Social neTworks (EXIST)[8]. Su objetivo fue la detección del sexismo en un sentido amplio,

desde la misoginia explícita hasta otras expresiones sutiles que implican conductas sexistas implícitas. Con este fin, propusieron una nueva categorización del sexismo y crearon una base de datos utilizando publicaciones de Twitter y la red social Gab en inglés y español[21]. La competencia se dividió en dos tareas:

- **Tarea 1:** Identificación del sexismo. Clasificación binaria de los tweets en sexistas o no sexistas.
- **Tarea 2:** Categorización del sexismo. Categorizar el mensaje según el tipo de sexismo (ideología y desigualdad, estereotipos y dominancia, cosificación, violencia sexual, misoginia y violencia no sexual).

Para la creación de la base de datos, recopilaron una serie de expresiones y términos populares, tanto en inglés como en español, que se utilizan comúnmente para subestimar el papel de las mujeres en la sociedad. Estos términos se han extraído de diferentes fuentes:

- Trabajos previos en el área.
- Cuentas de Twitter y hashtags utilizados para recopilar frases y expresiones que las mujeres reciben en su día a día.
- Expresiones extraídas del proyecto Everyday Sexism Project[3].

Los conjuntos de datos obtenidos fueron etiquetados mediante un enfoque de votación mayoritaria por parte de contribuidores externos en la plataforma Amazon Mechanical Turk[4] (MTurk), involucrando diferentes etapas.

La base de datos completa EXIST contiene 11345 entradas etiquetadas, de las cuales 5644 son en inglés y 5701 son en español. Cada registro de la base de datos estaba compuesto de la siguiente manera:

- **test case**: Contiene la cadena "EXIST2021" necesaria para la herramienta de evaluación EvALL.

[3] `https://everydaysexism.com/`
[4] `https://www.mturk.com/`

- **id**: Identifica de manera única el texto.
- **source**: Indica la fuente de datos. Toma los valores "“twitter” o “gab”.
- **language**: Especifica el idioma del texto. Toma los valores “en” (inglés) o “es” (español).
- **text**: Contiene el texto real.
- **task1**: Define si el texto es sexista o no. Toma los valores “sexist” (sexista) y “non-sexist” (no sexista).
- **task2**: Define el tipo de sexismo (si corresponde). Toma los valores:
  - *ideological-inequality*: Denota la categoría “Ideología y desigualdad”.
  - *misogyny-non-sexual-violence*: Denota la categoría “Misoginía y violencia no sexual”.
  - *objectification*: Denota la categoría “Cosificación”.
  - *sexual-violence*: Denota la categoría “Violencia sexual”.
  - *stereotyping-dominance*: Denota la categoría “Estereotipos y dominancia”.
  - *non-sexist*: Indica que el texto no expresa ningún comportamiento o discurso sexista.

Al igual que Rodríguez-Sánchez, el objetivo del conjunto de datos EXIST es cubrir el sexismo en un sentido amplio desde la misoginia explícita hasta otras expresiones sutiles que implican conductas sexistas implícitas.

Capítulo 3

Marco Teórico

3.1. Conceptos clave

En el marco de este estudio, es fundamental comprender una serie de conceptos clave que forman la base de la investigación sobre el lenguaje de odio, la violencia de género y sus manifestaciones en el contexto digital. Estos conceptos no solo definen el ámbito de estudio de este trabajo, sino que también ayudan a situar la relevancia y la urgencia de abordar estos fenómenos en las redes sociales y en la sociedad en general.

3.1.1. Lenguaje de odio

El lenguaje de odio se refiere a cualquier forma de expresión que difame, degrade o discrimine a una persona o grupo basado en atributos como raza, religión, origen étnico, género, orientación sexual, entre otros[1]. Este tipo de lenguaje tiene el potencial de causar un daño significativo tanto a nivel individual como comunitario, perpetuando prejuicios y fomentando un entorno de hostilidad y exclusión.

3.1.2. Violencia de género

La violencia de género se define como cualquier acto de violencia que se dirige contra una persona debido a su género, con el propósito de perpetuar la desigualdad

de poder entre los géneros. Esta forma de violencia se fundamenta en relaciones de poder desiguales y está relacionada con la discriminación sistemática y la opresión de las mujeres en diversas esferas de la vida social, económica, y política[17]. Es una manifestación crítica de la desigualdad de género y tiene profundas implicaciones para los derechos humanos y el bienestar de las personas afectadas.

3.1.3. Sexismo

El sexismo se refiere a la discriminación o prejuicio basado en el sexo o género de una persona, que a menudo resulta en la marginación o desventaja de un género respecto al otro. El sexismo perpetúa estereotipos de género y mantiene estructuras de poder desiguales entre hombres y mujeres[19]. El sexismo puede manifestarse de múltiples maneras, desde actitudes y creencias personales hasta políticas y prácticas institucionales que refuerzan la desigualdad.

3.1.4. Misoginia

La misoginia es la aversión, desprecio o prejuicio contra las mujeres, basado en la creencia de que son inferiores a los hombres. Esta forma de odio es un tipo extremo de sexismo que perpetúa desigualdades de género y violencia contra las mujeres[14]. La misoginia no solo afecta a las mujeres a nivel individual, sino que también sustenta estructuras sociales y culturales que desvalorizan y marginan a las mujeres sistemáticamente.

3.1.5. Cibersexismo

El cibersexismo es la expresión de prejuicios, privilegios y poder en espacios en línea y a través de la tecnología como medio[18]. En un mundo cada vez más digitalizado, el cibersexismo representa una extensión de las desigualdades de género tradicionales al ámbito digital, donde puede tomar formas como el acoso en línea, la violencia verbal, y la diseminación de contenido misógino y sexista.

3.2. Modelos de aprendizaje automático

El **Aprendizaje Automático** (Machine Learning) es la ciencia (y arte) de programar computadoras tal que puedan aprender a partir de datos[7]. Los modelos de Aprendizaje Automático se entrenan usando un conjunto de datos de entrenamiento y se utilizan para hacer predicciones en nuevos datos o clasificarlos. Durante la fase de entrenamiento se usa otro conjunto independiente (datos de prueba) para probar la eficacia del modelo.

Aprendizaje supervisado: Es un tipo de Aprendizaje Automático en el que los datos de entrenamiento que recibe el modelo incluyen las soluciones deseadas, llamadas labels (etiquetas).
Algunos de los modelos más relevantes de aprendizaje supervisado son:

- Regresión Lineal
- Regresión Logística
- Máquinas de Vectores de Soporte (SVM)

3.2.1. Regresión Lineal

La Regresión Lineal es un modelo que hace una predicción simplemente calculando una suma ponderada de las características de entrada, más una constante llamada término de sesgo (también llamado término de intercepción)[7].

$$\hat{y} = h_\theta(x) = x^T\theta, \tag{3.1}$$

donde:

- $\hat{y}$ es la variable que se quiere predecir.
- θ es el vector de parámetros del modelo, que contiene el término de sesgo θ_0 y los pesos de las características θ_1 a θ_n.

- x^T es el vector traspuesto de características de la instancia, que contiene x_0 a x_n, con $x_0 = 1$.
- $x^T\theta$ es el producto punto de los vectores θ y x^T, que es igual a $\theta_0 x_0 + \theta_1 x_1 + \cdots + \theta_n x_n$.
- h_θ es la función hipótesis, que usa los parámetros del modelo θ.

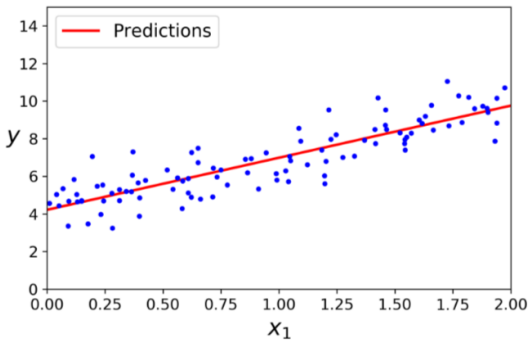


Figura 3.1: Modelo de regresión lineal [7]

En la figura 3.1, el eje X_1 representa los datos de entrenamiento, mientras que el eje y son las etiquetas reales de cada dato. Así, la recta roja representa el modelo lineal.

3.2.2. Regresión Logística

La Regresión Logística es un clasificador binario, usado comúnmente para estimar la probabilidad de que una instancia pertenezca a una clase particular. Si la probabilidad estimada es mayor al 50 %, entonces el modelo predice que pertenece a esa clase (clase positiva), de lo contrario, predice que no (clase negativa). Al igual que la Regresión Lineal, calcula una suma ponderada de las características de entrada (más un término de sesgo), pero en lugar de generar el resultado directamente, genera la logística de este resultado.[7]

$$\hat{p} = h_\theta(x) = \sigma(x^T\theta). \tag{3.2}$$

La logística (denotada σ) es una función sigmoide (con forma S), que devuelve un número entre 0 y 1. Se define como sigue:

$$\sigma(t) = \frac{1}{1 + exp(-t)}. \tag{3.3}$$

Una vez que el modelo de Regresión Logística ha estimado la probabilidad de que una instancia x pertenezca a la clase positiva, entonces puede hacer la predicción $\hat{y}$ fácilmente:

$$\hat{y} = \begin{cases} 0 & \text{si } \hat{p} < 0.5 \\ 1 & \text{si } \hat{p} \geq 0.5 \end{cases} \tag{3.4}$$

Notemos que un modelo de Regresión Logística predice 1 si $x^T\theta$ es ≥ 5 (clase positiva), y 0 si es < 5 (clase negativa).

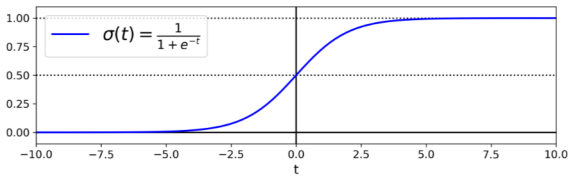


Figura 3.2: Modelo de regresión logística [7]

En la figura 3.2 vemos la gráfica de la función logística, con su forma sigmoidal.

3.2.3. Máquinas de Vectores de Soporte

Una Máquina de Vectores de Soporte (SVM, del inglés Support Vector Machines) es un modelo de Aprendizaje Automático muy potente y versátil, capaz de realizar clasificación lineal o no lineal, regresión e incluso detección de valores atípicos. Las Máquinas de Vectores de Soporte son particularmente adecuadas para la clasificación de conjuntos de datos complejos pero de tamaño pequeño o mediano[7].

Un clasificador lineal de SVM es un modelo de aprendizaje supervisado utilizado para resolver problemas de clasificación binaria. La idea principal de un modelo de SVM con núcleo lineal, es encontrar la línea que separa dos clases, pero que permane-

ce lo más alejada posible de las instancias de cada clase que están más cercanas a la otra clase (llamadas *support vectors*). A esto se le llama *clasificación de gran margen.*

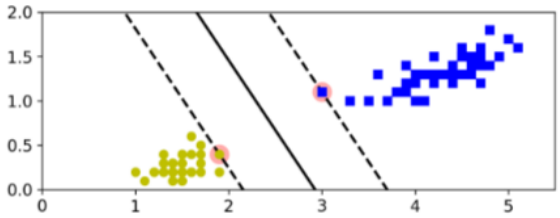


Figura 3.3: Clasificación de gran margen [7]

En la figura 3.3 podemos ver a las instancias seleccionadas de cada clase (los puntos resaltados) y el modelo lineal (la recta continua) de la SVM que hace que las distancias entre las líneas punteadas sea la mayor.

Una Máquina de Vectores de Soporte con núcleo RBF (Radial Basis Function) es un modelo de aprendizaje supervisado utilizado para tareas de clasificación y regresión. El uso del núcleo RBF (también llamado núcleo Gaussiano) permite que la SVM maneje datos que no son linealmente separables en el espacio original de características al proyectarlos a un espacio de características de mayor dimensión. La función RBF es una función de similaridad que mide cuánto se parece cada instancia a un punto de referencia en particular[7] y se define como sigue:

$$\phi_\gamma(x, l) = exp(-\gamma||x - l||^2). \tag{3.5}$$

Es una función con forma de campana que varía entre cero (muy alejado del punto de referencia) y uno (en el punto de referencia).

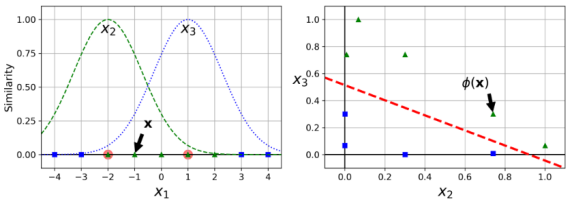


Figura 3.4: Similaridad usando la función Gaussiana RBF[7]

En la figura 3.4, los puntos de referencia seleccionados son $x_2 = -2$ y $x_3 = 1$. La instancia $x = -1$ está localizada a una distancia de 1 del primer punto de referencia (x_2), y a una distancia de 2 del segundo (x_3). Así, si definimos $\gamma = 0.3$, sus nuevas características son $x_2 = exp(-0.3 \times 1^2) \approx 0.74$ y $x_3 = exp(-0.3 \times 2^2) \approx 0.30$. La gráfica en la parte derecha de la figura 3.4 muestra la base de datos transformada (con las nuevas características). Como se puede observar, ahora las clases se pueden separar con un modelo lineal (la recta punteada).

3.2.4. Aprendizaje Profundo y Redes Transformers

El Aprendizaje Profundo es una subárea del Aprendizaje Automático que se basa en el uso de redes neuronales artificiales con múltiples capas (capas profundas) para modelar y comprender datos complejos. Esta técnica ha ganado popularidad debido a su capacidad para superar a otros métodos en tareas como el reconocimiento de voz, la visión por computadora, el Procesamiento de Lenguaje Natural y muchas otras áreas[13].

Vaswani en 2017 introdujo una nueva arquitectura llamada Transformer, la cual utiliza el mecanismo de "self-attention" para procesar secuencias de datos. Este mecanismo evalúa la importancia de cada palabra en una secuencia en relación con todas las demás palabras, permitiendo capturar dependencias contextuales de manera más eficiente que los enfoques tradicionales[23].

BERT (Bidirectional Encoder Representations from Transformers) es un modelo de lenguaje profundo desarrollado por Google que se basa en la arquitectura de Transformers. A diferencia de los modelos de lenguaje tradicionales que procesan el texto de manera unidireccional (de izquierda a derecha o de derecha a izquierda), BERT tiene la capacidad de procesar el texto de manera bidireccional. Esto significa que considera el contexto completo de una palabra en una oración, tanto desde la izquierda como desde la derecha, lo que permite una comprensión más rica y precisa del significado[3].

3.3. Medidas de desempeño

Las medidas de desempeño de un modelo de Aprendizaje Automático son métricas utilizadas para evaluar la eficacia y precisión de un modelo en realizar su tarea[7]. Estas métricas ayudan a determinar qué tan bien se desempeña el modelo con respecto a los datos de entrenamiento y prueba, y son cruciales para comparar diferentes modelos y técnicas. A continuación se describen algunas de las medidas de desempeño más comunes:

3.3.1. Matriz de confusión

La matriz de confusión es una tabla que describe el rendimiento de un modelo de clasificación mostrando las verdaderas predicciones contra las predicciones realizadas por el modelo. Por ejemplo, si un modelo entrenado para detectar misoginia clasifica un texto como misógino, se considera que fue positivo, de lo contrario negativo. En la figura 3.5 vemos los valores de una matriz de confusión.

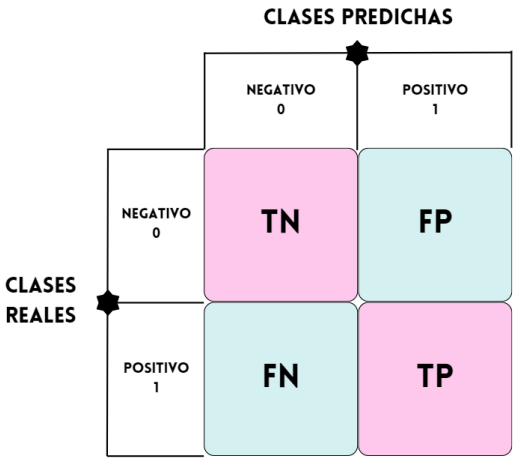


Figura 3.5: Matriz de confusión.

Donde:

- Verdaderos negativos (TN): Número de predicciones correctas para la clase negativa.

- Verdaderos positivos (TP): Número de predicciones correctas para la clase positiva.
- Falsos negativos (FN): Número de predicciones incorrectas para la clase negativa.
- Falsos positivos (FP): Número de predicciones incorrectas para la clase positiva.

3.3.2. Exactitud

La exactitud es la proporción de predicciones correctas realizadas por el modelo en relación con el total de predicciones. Se calcula como:

$$\text{Exactitud} = \frac{\text{Predicciones correctas}}{\text{Total de predicciones}} \tag{3.6}$$

3.3.3. Precisión y Sensibilidad

La precisión mide la proporción de verdaderos positivos entre el total de predicciones positivas realizadas por el modelo. Es especialmente útil cuando el costo de los falsos positivos es alto.

$$\text{Precisión} = \frac{\text{Verdaderos positivos (TP)}}{\text{Verdaderos positivos (TP)} + \text{Falsos positivos (FP)}} \tag{3.7}$$

La sensibilidad, también conocida como *recall*, mide la proporción de verdaderos positivos detectados correctamente por el modelo respecto al total de positivos reales.

$$\text{Recall} = \frac{\text{Verdaderos positivos (TP)}}{\text{Verdaderos positivos (TP)} + \text{Falsos negativos (FN)}} \tag{3.8}$$

3.3.4. Especificidad

La especificidad mide la proporción de verdaderos negativos que son correctamente identificados por el modelo. Se calcula de la siguiente manera:

$$\text{Especificidad} = \frac{\text{Verdaderos negativos (TN)}}{\text{Verdaderos negativos (TN)} + \text{Falsos positivos (FP)}} \tag{3.9}$$

3.3.5. Área bajo la curva (AUC-ROC)

La AUC (Area Under Curve) de la curva ROC (Receiver Operating Characteristic) mide la capacidad de un modelo para distinguir entre clases. Cuanto más cerca esté el valor del AUC a 1, mejor será el rendimiento del modelo. La curva ROC grafica la tasa de verdaderos positivos (sensibilidad) contra la tasa de falsos positivos (1 − especificidad).

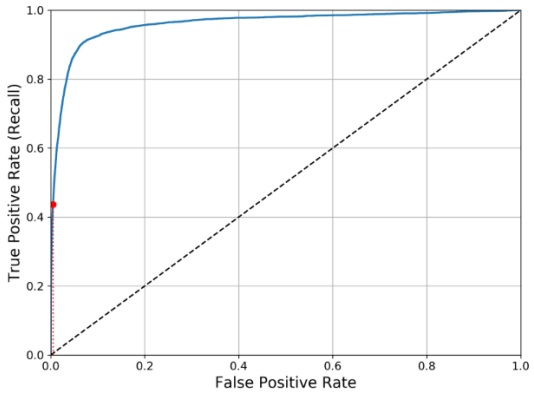


Figura 3.6: Curva ROC [7]

La figura 3.6 muestra la curva ROC de un modelo de Aprendizaje Automático (la línea azul) contra la curva ROC de un clasificador aleatorio (la línea punteada); un buen clasificador se mantiene lo más alejado posible de esa recta (hacia la esquina superior izquierda)[7].

3.4. Métodos de evaluación

Como hemos mencionado, una forma de medir el desempeño de nuestro modelo es entrenarlo con una parte de nuestra base de datos y posteriormente evaluarlo con la

otra parte reservada. Sin embargo, al hacerlo de esa manera nos arriesgamos a obtener resultados poco precisos, pues la eficacia del modelo varía de acuerdo a los conjuntos de entrenamiento y de prueba seleccionados. Es por eso que existen distintos métodos de evaluación, que retratan de forma precisa el rendimiento del modelo.

3.4.1. Validación cruzada

La validación cruzada es un método estadístico para evaluar el rendimiento de generalización que es más estable y exhaustivo que el uso de una división en un conjunto de entrenamiento y un conjunto de prueba[16]. La versión más comúnmente utilizada de la validación cruzada es la validación cruzada k-fold, donde k es un número especificado por el usuario, generalmente 5 o 10.

En la validación cruzada de cinco pliegues, los datos se dividen primero en cinco partes de tamaño aproximadamente igual, llamadas pliegues. El modelo se entrena utilizando el primer pliegue como conjunto de prueba, y los pliegues restantes (2-5) se utilizan como conjunto de entrenamiento. El modelo se construye utilizando los datos en los pliegues 2-5, y luego se evalúa en el pliegue 1. Este proceso se repite utilizando los pliegues 2, 3, 4 y 5 como conjuntos de prueba y los restantes como datos de entrenamiento, respectivamente. Para cada una de estas cinco divisiones de los datos en conjuntos de entrenamiento y prueba, calculamos la precisión (o la métrica que hayamos elegido), y el promedio de los resultados obtenidos será el rendimiento general del modelo. El proceso se ilustra en la figura 3.7:

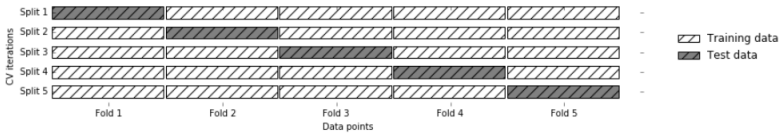


Figura 3.7: División de los datos en la validación cruzada de cinco pliegues [16]

3.4.2. Valicación cruzada estratificada

Dividir el conjunto de datos en k pliegues, donde tomamos los k-ésimos datos contiguos como recién describimos puede no ser una buena idea, pues si las clases de la base de datos no están balanceadas, existe el riesgo de entrenar nuestro modelo con datos de sólo una de las clases y evaluarlo con clases diferentes, lo que resultaría en un mal desempeño.

Es por eso que existen diversas variaciones de la validación cruzada, una de ellas es la validación cruzada estratificada. En la validación cruzada estratificada, dividimos los datos de manera que las proporciones entre las clases sean las mismas en cada pliegue como lo son en todo el conjunto de datos[16]. Así, en cada iteración estamos entrenando y evaluando el modelo en conjuntos representativos de la base de datos completa.

3.5. Técnicas de Procesamiento de Lenguaje Natural

Para que los modelos de Aprendizaje Automático puedan trabajar eficazmente con datos de texto, es necesario aplicar diversas técnicas de preprocesamiento. Estas técnicas aseguran que los datos sean consistentes, relevantes y adecuadamente formateados para el análisis, lo que resulta en modelos más precisos y eficientes.
A continuación se describen algunas de las técnicas más comunes utilizadas en el PLN:

3.5.1. Tokenización

La tokenización es el proceso de dividir un texto en unidades más pequeñas llamadas tókens, que pueden ser palabras, frases o caracteres individuales[11]. Esto ayuda a descomponer el texto en elementos manejables que pueden ser analizados por los modelos de Aprendizaje Automático.

3.5.2. Eliminación de palabras vacías

Las palabras vacías (también llamadas *stop words*) son palabras comunes como "el", "la", "y", "de", que no aportan mucho significado al análisis de texto[22]. La eliminación de estas palabras ayuda a reducir el ruido en los datos y a centrarse en las palabras más importantes para el análisis.

3.5.3. Lematización y Estematización

La lematización es el proceso de reducir las palabras a su forma base o lema[11]. Por ejemplo, la palabra "corriendo" se transforma en "correr".

La estematización es el proceso de reducir las palabras a su raíz[11]. Por ejemplo, las palabras "amigos", "amiga" y "amigable", se reducen a "amig".

3.5.4. Bolsa de Palabras

El modelo de Bolsa de Palabras es una técnica ampliamente utilizada en el Procesamiento de Lenguaje Natural para la representación de texto.

El enfoque principal del modelo es crear una matriz de vocabulario de toda la base de datos, donde cada fila es un texto de nuestro corpus y cada columna es una palabra del vocabulario[11]. En cada columna aparecerá el valor de ocurrencias de esa palabra en el texto de la fila correspondiente.

3.5.5. Frecuencia de Término-Frecuencia Inversa

La Frecuencia de Término-Frecuencia Inversa de Documento (TF-IDF) es una técnica utilizada en el Procesamiento de Lenguaje Natural para evaluar la importancia de una palabra en un documento dentro de un corpus de documentos[15]. La técnica pondera las palabras basándose en dos métricas:

- **Frecuencia de Término (TF):** Mide cuántas veces aparece un término en un

documento. Se calcula como el número de veces que una palabra aparece en un documento dividido por el número total de palabras en ese documento.

- **Frecuencia Inversa de Documento (IDF):** Mide la importancia de un término en el corpus. Se calcula como el logaritmo del número total de documentos dividido por el número de documentos que contienen la palabra.

El resultado es que las palabras comunes, que aparecen en muchos documentos, tienen un peso más bajo, mientras que las palabras raras, que aparecen en pocos documentos, tienen un peso más alto. Esto ayuda a resaltar palabras significativas y relevantes para cada documento, mejorando la efectividad en tareas como la búsqueda de información y la clasificación de textos.

3.5.6. Incrustaciones de Palabras

Las incrustaciones (embbedings) son representaciones densas y continuas de palabras en un espacio de alta dimensión, que capturan relaciones semánticas entre ellas[11]. Modelos populares para generar incrustaciones incluyen Word2Vec, GloVe y FastText. Estas representaciones permiten que los modelos de Aprendizaje Automático comprendan mejor las relaciones contextuales entre las palabras.

Capítulo 4

Propuesta

El presente capítulo se enfoca en detallar la metodología propuesta y los experimentos realizados para abordar el problema de la detección de sexismo y misoginia en tweets en español. Dada la creciente presencia de discurso sexista y misógino en redes sociales, es crucial desarrollar modelos de Aprendizaje Automático que puedan identificar y clasificar estos contenidos de manera eficaz. Sin embargo, uno de los desafíos más significativos en esta tarea es la limitada disponibilidad de bases de datos etiquetadas en español, lo que afecta la capacidad de los modelos para generalizar y ofrecer resultados precisos.

Primero se describen las bases de datos utilizadas, señalando los problemas asociados a su tamaño reducido y representatividad limitada. Luego, se detallan los diferentes experimentos realizados con varios modelos de Aprendizaje Automático, incluyendo Máquinas de Vectores de Soporte Lineal, Regresión Logística y el modelo preentrenado SaBERT. Cada uno de estos modelos se evaluó usando validación cruzada estratificada para determinar su eficacia en la detección de sexismo y misoginia en el corpus de tweets en español.

Finalmente, se presenta la propuesta de aumentar las bases de datos mediante la inclusión de tweets en inglés traducidos al español utilizando el API de Google Translator. Esta ampliación tiene como objetivo mejorar la representatividad del corpus y,

en consecuencia, el rendimiento de los modelos. Esta propuesta busca no solo mejorar los resultados actuales, sino también proporcionar una estrategia viable para futuras investigaciones en el área de detección de lenguaje de odio en redes sociales.

4.1. Bases de datos

Debido a las limitaciones individuales de las bases de datos existentes de tweets en español, se ha realizado una recopilación de tres de ellas. Ya que la misoginia es una manifestación específica del sexismo, se determinó unificar los siguientes bases de datos:

- **AMI IberEval:** La base de datos en español consta de 4138 tweets, de los cuales 2064 (49.9 %) son misóginos y 2074 (50.1 %) son no misóginos.
- **MeTwo:** Contiene 3600 tweets, de los cuales 1152 (32 %) son sexistas, 2181 (60.6 %) son no sexistas y 267 (7.4 %) son dudosos (doubtful).
- **EXIST IberLEF:** La base de datos EXIST en español consta de 5701 entradas, de las cuales 2864 (50.2 %) son sexistas y 2837 (49.8 %) son no sexistas.

Así, como el corpus de AMI IberEval es una clasificación de misoginia y los dos restantes, MeTWo y EXIST IberLEF, abarcan distintas formas de sexismo más sutil, al unirlas obtenemos un corpus amplio y representativo de sexismo y misoginia.

Para poder unir las bases de datos, necesitamos hacerlas compatibles entre sí mediante un cambio en las etiquetas. Como el objetivo es obtener un modelo capaz de generalizar la detección en distintas formas de sexismo, es crucial estandarizar las etiquetas y categorías utilizadas en los tres conjuntos de datos. Esto nos permitirá entrenar un modelo más robusto y adaptable a diferentes contextos y expresiones de sexismo.

Se decidió por reducir el problema a un problema de clasificación binaria, así cada entrada de la base de datos posee los siguientes atributos:

- **text:** Contiene el texto del tweet.
- **klass:** Identifica el comportamiento del tweet. Toma los valores:
  - *sexist:* Si el tweet tiene un contenido sexista o misógino.
  - *non-sexist:* Si el tweet no tiene connotaciones sexistas.

Para adaptar las bases de datos a este nuevo formato, se realizaron las siguientes transformaciones:

- **MeTwo:** Se eliminó la columna de "status_id"; se desecharon los registros cuya categoría era "doubtful"; se cambió el nombre de la columna "categoría" por "klass".
- **AMI IberEval:** Se eliminaron las columnas "id", "misogyny_category" y "target"; en el campo "misogynous", las etiquetas con valor 1 se cambiaron a "sexist" y las etiquetas con valor 0 se cambiaron a "non-sexist"; se cambió el nombre de la columna "misogynous" por "klass".
- **EXIST IberLEF:** Se eliminaron las columnas "test case", "id", "source", "language" y "task2"; se cambió el nombre de la columna "task1" por "klass".

Finalmente, la base de datos unificada se compuso de 13172 tweets, de los cuales 6080 (46.2 %) son sexistas y 7092 (53.8 %) son no sexistas. A continuación se presenta una tabla que muestra la composición final de nuestra base de datos:

	AMI	**MeTwo**	**EXIST**	**Unificada**
sexist	2064	1152	2864	6080
non-sexist	2074	2181	2837	7092
total	4138	3333	5701	13172

Tabla 4.1: Distribución de las bases de datos

4.2. Metodología experimental

A continuación, se detallará el enfoque experimental utilizado para abordar el problema de la detección de sexismo y misoginia en tweets en español. La metodolo-

gía incluye el preprocesamiento de los datos, la elección de modelos de Aprendizaje Automático, el procedimiento experimental seguido, y las métricas de evaluación empleadas para evaluar el desempeño de los modelos.

4.2.1. Preprocesamiento de los datos

Para poder alimentar la base de datos a nuestro modelo, es crucial someter los datos a una serie de modificaciones conocidas como preprocesamiento. Esta fase es fundamental en nuestro estudio para asegurar la calidad y la coherencia de los textos antes de ser utilizados en los modelos de Aprendizaje Automático.

Para este proceso, se ha utilizado el modelo de texto propuesto por S. Téllez en 2018 incluido en Micro Clasificación de Texto (μTC). Dicho modelo recibe una lista que contiene los datos y le aplica las siguientes transformaciones[22]:

- **Normalización de caracteres:** El texto recibido se modifica de la siguiente manera:

  1. Remueve diacríticos.
  2. Elimina letras duplicadas.
  3. Agrega " ∼ " al inicio, al final y los espacios entre las palabras se reemplazan por " ∼ ".
  4. Transforma todas las letras a minúsculas.
  5. Cambia los números por "_num".
  6. Cambia los usuarios por "_usr".
  7. Cambia las URL por "_url".
  8. Cambia los emojis y emoticonos por "_pos" (positivos) o "_neg" (negativos), dependiendo de la emoción que transmitan.

- **Tokenización:** Divide el texto en fragmentos más pequeños llamados tókens dependiendo de los parámetros que reciba:

1. *Parámetro negativo:* Si recibe un entero negativo n, entonces separa el texto en grupos de n palabras, llamados n-gramas de palabras.
2. *Parámetro positivo:* Si recibe un entero positivo n, entonces separa el texto en grupos de n caracteres, llamados q-gramas de caracteres.
3. *Duplas positivas:* Si recibe una dupla de enteros positivos (n, m), entonces separa el texto en grupos de n palabras, saltándose m palabras del texto para seleccionar la siguiente.

De esta forma, el espacio de palabras se ordena de forma alfabética, donde primero van los n-gramas de palabras y después los q-gramas de caracteres.

- **Vectorización:** Transforma el texto en un vector utilizando TF-IDF (Frecuencia de Término-Frecuencia Inversa de Documento), de la siguiente manera:

  1. Tokeniza el texto recibido.
  2. Calcula el TF(t), el número de veces que aparece el tóken en el espacio de palabras del corpus.
  3. Calcula el IDF de cada tokén t con la fórmula:

  $$IDF(t) = \log_2 \left(\frac{N}{TF(t)} \right), \tag{4.1}$$

  donde N es el número de documentos en el espacio de palabras.

  4. Calcula el TF-IDF para cada tóken del texto con la fórmula:

  $$TF - IDF(t) = TF(t) \times IDF(t) \tag{4.2}$$

  5. Cada vector tendrá el valor $TF-IDF(t)$ en los documentos donde aparece el tóken, y 0 en donde no aparece ninguno de los tókens del texto.
  6. Normaliza el vector diviendo todas las entradas entre su norma Euclídea,

que se calcula como:

$$norma = \sqrt{\sum_i (TF - IDF(t_i))^2} \tag{4.3}$$

4.2.2. Modelos de Aprendizaje Automático

La elección y comparación de modelos de Aprendizaje Automático juega un papel fundamental en la efectividad y precisión de la detección de sexismo y misognia en redes sociales. Entre los modelos que seleccionamos para la experimentación se encuentran el SVM con núcleo lineal, SVM con núcleo RBF (Radial Basis Function), Regresión Logística y el modelo preentrenado SaBERT.

Máquina de Vectores de Soporte Lineal

Al igual que con el modelo de texto, decidimos seguir con la línea de trabajo de S. Téllez, donde escogieron como modelo de clasificación una SVM con núcleo lineal. Esto con el objetivo de tener una referencia clara para nuestros otros resultados.

Entre sus principales ventajas, tenemos que los SVM lineales son computacionalmente más eficientes en comparación con otros modelos complejos, lo que los hace adecuados para problemas con grandes conjuntos de datos o muchas características, donde otros métodos pueden resultar excesivamente costosos.

Máquina de Vectores de Soporte con núcleo RBF

Hemos decidido emplear un modelo SVM con núcleo RBF como uno de los comparativos de este trabajo debido a su reconocida capacidad para manejar datos no linealmente separables y su robustez en diversas tareas de clasificación. El núcleo RBF permite al SVM capturar relaciones complejas y no lineales en los datos, lo que es esencial para problemas donde las clases no son linealmente separables. Esta flexibilidad mejora el poder predictivo del modelo.

Además, la capacidad del núcleo RBF para ajustar finamente los patrones relevantes sin sobreajustarse al ruido del conjunto de entrenamiento ayuda a encontrar un equilibrio adecuado entre el sesgo y la varianza, mejorando la generalización en datos no vistos.

Regresión Logística

Otro de los modelos a comparar que se eligió fue el de Regresión Logística, debido a su simplicidad, interpretabilidad y eficacia probada en problemas de clasificación binaria, al igual que por ser un modelo con un costo computacional menor, lo cual facilita el manejo de conjuntos de datos grandes y permite iteraciones rápidas en el desarrollo del modelo, optimizando el uso de recursos y tiempo durante el proceso de investigación.

La demostrada eficacia de la Regresión Logística en una amplia gama de problemas de clasificación binaria, su flexibilidad y adaptabilidad a diferentes contextos y tipos de datos, la hacen una herramienta valiosa en la evaluación comparativa de modelos de detección de sexismo. Además, la Regresión Logística proporciona un punto de referencia sólido para comparar el desempeño de modelos más complejos. Esto permite evaluar la ganancia en precisión y justificación de la complejidad adicional de los otros modelos propuestos.

SaBERT

Los modelos basados en transformadores han demostrado ser extremadamente efectivos en tareas de Procesamiento de Lenguaje Natural, como la clasificación de texto. Particularmente, BERT ha establecido nuevos estándares en múltiples tareas de PLN, ofreciendo una referencia sólida y confiable para evaluar la efectividad de nuestros modelos[3].

En este trabajo, utilizaremos SaBERT, una variación de BERT creado para el análisis de sentimientos en español. El modelo fue preentrenado en un conjunto de

datos que contenía 11,500 tweets en español recopilados de varias regiones, tanto positivos como negativos, obteniendo una exactitud de 0.865.

4.2.3. Procedimiento experimental

Para evaluar el desempeño los modelos se llevó a cabo la metodología descrita a continuación. Los dos modelos de Máquinas de Vectores de Soporte y el modelo de Regresión Logística siguieron los mismos pasos, pero el modelo preentrenado SaBERT requirió un enfoque ligeramente diferente, por la naturaleza de su programación.

División de datos

Se dividió la base de datos en dos conjuntos, uno de entrenamiento (80 %) y otro de prueba (20 %), con la función *train_ test_ split* de scikit-learn[1]. Así garantizamos conjuntos con clases balanceadas y con resultados reproducibles y consistentes.

Los conjuntos de entrenamiento y de prueba fueron los mismos para todos los modelos, para poder tener una comparación acertada.

Entrenamiento de modelos

Los modelos de SVM con núcleo lineal y Regresión Logística, fueron entrenados realizando una búsqueda de hiperparámetros y validación cruzada estratificada.

Para la búsqueda de hiperparámetros se utilizó RandomizedSearchCV[2], una técnica muy eficiente cuando el espacio de hiperparámetros es grande. A diferencia de GridSearchCV[3], que prueba todas las combinaciones posibles de los hiperparámetros especificados, RandomizedSearchCV selecciona aleatoriamente un número fijo de combinaciones y las evalúa. Para cada iteración de la validación cruzada estratificada

[1]scikit-learn es una biblioteca de Aprendizaje Automático gratuita y de código abierto para el lenguaje de programación Python.

[2]`https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html`

[3]`https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html`

se compararon diez combinaciones, donde se escoge la que obtiene la mejor exactitud.

El espacio de hiperparámetros a comprobar fue seleccionado eligiendo una lista de los valores más comunes utilizados por la mayoría de modelos de Aprendizaje Automático:

- **C:** 0.001, 0.01, 0.1, 1, 10, 100.
- **max_iter:** 5000, 10000, 15000, 20000, 30000.
- **tol:** 1e-4, 1e-3, 1e-2.

Donde 'C' es el parámetro de regularización, el cual controla el equilibrio entre maximizar el margen y minimizar el error de clasificación; un valor más alto permite que el modelo se ajuste más a los datos de entrenamiento, mientras que un valor más bajo busca un margen más amplio, lo que permite que el modelo generalice mejor.

'max_iter' define el máximo número de iteraciones que el algoritmo debe realizar antes de detenerse; si el algoritmo no converge dentro de este número de iteraciones, se detendrá y retornará el mejor resultado encontrado hasta ese punto.

Finalmente, 'tol' representa la tolerancia para el criterio de parada, la cual controla la precisión de la solución, pues si el cambio en la función objetivo entre iteraciones consecutivas es menor que 'tol', el algoritmo se detendrá; valores más pequeños pueden dar lugar a soluciones más precisas pero pueden requerir más iteraciones.

Debido a las limitaciones del equipo con el que se programaron los modelos, el modelo de SVM con núcleo RBF se entrenó una única vez con el conjunto de entrenamiento completo. Esto porque el costo y tiempo de realizar RandomizedSearchCV era demasiado alto.

El modelo preentrenado SaBERT[4] se entrenó utilizando la biblioteca "transformers" de Hugging Face[5]. El entrenamiento se realizó con una técnica de procesamiento por lotes (batch processing) durante sólo dos épocas, ya que a partir de una tercer época el modelo tiende a sobreajustar los datos de entrenamiento rápidamente.

Evaluación de modelos

Para la validación de los modelos se utilizó validación cruzada estratificada[6], una variación del método de validación cruzada que garantiza que la proporción de las clases en cada iteración sea la misma que en nuestra base de datos original. Se dividió el conjunto de entrenamiento en cinco subconjuntos, donde en cada iteración se entrenaba el modelo en cuatro de ellos y se evaluaba en el restante. Los modelos se evaluaron usando exactitud, promediando las obtenidas en las cinco iteraciones.

Después, evaluamos los modelos de SVM con núcleo lineal y Regresión Logística en el conjunto de prueba, con los mejores parámetros encontrados por RandomizedSearchCV.

Para el modelo de SVM con núcleo RBF graficamos las curvas de precisión y sensibilidad contra los umbrales de decisición y seleccionamos el umbral de decisión que nos dé la mejor exactitud.

Para el modelo preentrenado SaBERT, el modelo se evalúa en el conjunto de prueba, y al finalizar todas las épocas, los mejores parámetros y el modelo final se guardan.

[4] `https://huggingface.co/VerificadoProfesional/SaBERT-Spanish-Sentiment-Analysis`

[5] Hugging Face, Inc. es una empresa estadounidense que desarrolla herramientas para crear aplicaciones utilizando el Aprendizaje Automático.

[6] `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html`

4.2.4. Métricas de evaluación

Para evaluar los modelos, hemos calculado la exactitud en el mismo conjunto de prueba utilizando la función *accuracy_ score* de scikit-learn. A modo de visualización, también mostraremos la matriz de confusión de cada modelo.

Además hemos graficado la curva ROC de cada modelo, así como su área bajo la curva (AUC), para poder analizar qué tan bien distinguen entre las clases del problema.

4.3. Ampliación de la base de datos

Para abordar las limitaciones actuales de nuestra base de datos existente en la detección de sexismo, proponemos aumentar significativamente la cantidad y diversidad de datos disponibles. Esto nos permitirá mejorar la representatividad de las muestras y la cobertura de diferentes tipos de expresiones sexistas en contextos variados.

En el área de Aprendizaje Automático, el concepto de aumento de datos se refiere a un conjunto de estrategias para crear nuevos datos a partir de los ya existentes, con el propósito de mejorar su representatividad y tamaño, de forma que incremente la eficacia del modelo. Mientras que el aumento de datos se ha convertido en una técnica estándar para entrenar redes profundas en el procesamiento de imágenes, no es una práctica común en el entrenamiento de redes para tareas de Procesamiento de Lenguaje Natural[4].

Fadaee en su trabajo en 2017 menciona las ventajas de usar traducción inversa, que se refiere a la traducción de un texto a otro idioma y después nuevamente a su idioma original, para aumentar la incidencia de las palabras menos comunes en la base de datos alimentada a un modelo de traducción de inglés a alemán[4].

Si bien técnicamente no se considera aumento de datos, hemos decidido aprove-

char nuestra disponibilidad de acceso a una base de datos de tweets en inglés para intentar mejorar la eficacia de nuetros modelos, traduciendo los tweets al español para añadirlos a nuestra base de datos actual.

4.3.1. Estrategias de ampliación

Hemos utilizado la base de datos en inglés de EXIST IberLEF, la cual contiene 5644 tweets, de los cuales 2794 (49.5 %) son sexistas y 2850 (50.5 %) son no sexistas. Cada entrada contiene los siguientes campos:

"test case"	"id"	"source"	"language"	"text"	"task1"	"task2"

de los cuales, sólo conservaremos las columnas de "text" y "task1", y las demás las desecharemos. Recordemos que:

- **text:** Contiene el texto del tweet.
- **task1:** Define si el texto es sexista o no. Toma los valores "sexist" si es sexista y "non-sexist" si no lo es.

Para traducir la columna de "text", utilizamos el API de Google Translate, una herramienta que facilita la traducción automática entre diferentes idiomas. Este API nos permitió generar versiones traducidas de los tweets originales para añadirlas a nuestra base de datos.

Finalmente, sólo hizo falta cambiar el nombre de la columna "task1" por "klass" para poder fusionar las bases de datos. Nuestro corpus aumentado contiene 18,816 tweets, de los cuales 8874 (47.2 %) son sexistas y 9942 (52.8 %) son no sexistas. La distribución de los datos con la nueva base de datos traducida se muestra en la tabla 4.2.

Para comprobar que la traducción de la base de datos no afectó a las etiquetas ya asignadas, hemos seleccionado una muestra aleatoria de cien tweets traducidos, de los cuales cincuenta de ellos están clasificados como sexistas y los otros cincuenta

	Unificada	**EXIST Inglés**	**Aumentada**
sexist	6080	2794	8874
non-sexist	7092	2850	9942
total	13172	5644	18816

Tabla 4.2: Distribución de la base de datos aumentada

como no sexistas. Después, fueron analizados manualmente y, en aquellos donde había duda, se usó ChatGPT[7] para clasificarlos como sexistas o no sexistas.

Para los tweets sexistas, se usó ChatGPT para clasificar cuatro de ellos, de los cuales tres los catalogó como no sexistas. Para los no sexistas, se usó ChatGPT en tres ocasiones, de las cuales dos los identificó como sexistas.

Así, podemos generalizar diciendo que de la base de datos traducida, un aproximado del 5 % de los textos podrían haber cambiado su etiqueta orignal, producto del cambio en el idioma. Este es un margen de error aceptable que estamos dispuestos a asumir.

[7] `https://chatgpt.com/`

Capítulo 5

Resultados experimentales

En este capítulo mostraremos los resultados de los modelos con la base de datos unificada y con la propuesta de la base de datos aumentada con la traducción de tweets en inglés. También analizaremos qué nos indican las distintas gráficas y métricas de evaluación utilizadas.

5.1. Resultados preliminares

En esta sección mostraremos los resultados de las curvas ROC, precisión y sensibilidad de cada modelo, así como su exactitud y analizaremos su desempeño con la base de datos unificada.

A continuación mostramos una tabla comparando la exactitud obtenida en el conjunto de entrenamiento y en el conjunto de prueba para los cuatro modelos (tabla 5.1):

	SVM Lineal	**SVM RBF**	**Reg Log**	**SaBERT**
Entrenamiento	0.742	0.75	0.748	0.922
Prueba	0.753	0.757	0.75	0.797

Tabla 5.1: Exactitud de los modelos en los conjuntos de entrenamiento y prueba

Podemos ver que los modelos tuvieron resultados muy similares, aunque aún podemos resaltar peculiaridades de cada uno. Como esperábamos, SaBERT obtuvo el

mejor resultado, con una exactitud de 0.797, sin embargo, es un resultado mucho peor a la exactitud del conjunto de entrenamiento pues este modelo es el más dado a sobreajustar de los cuatro. De los tres modelos restantes, sorprendentemente la SVM con núcleo RBF fue el mejor, a pesar de no haber podido realizar una búsqueda de hiperparámetros.

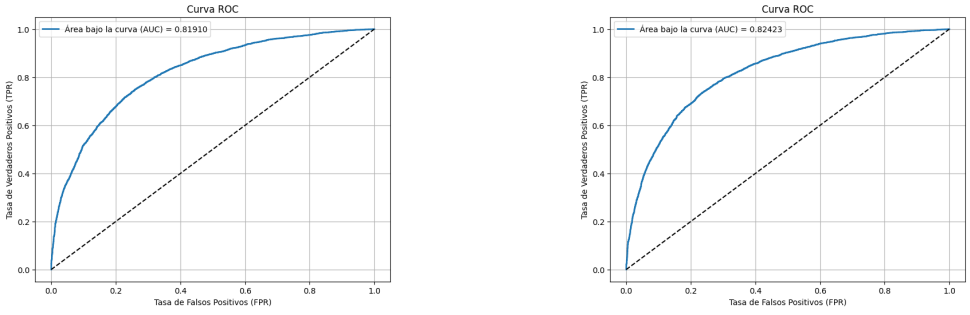
En la siguiente tabla podemos comparar el área bajo la curva ROC de cada uno de los modelos:

	SVM Lineal	**SVM RBF**	**Reg Log**	**SaBERT**
AUC	0.8191	0.8242	0.8164	0.8693

Tabla 5.2: Área bajo la curva ROC de los modelos

Nuevamente, nuestros modelos tienen resultados muy similares, siendo SaBERT el mejor. Sin embargo, todos los modelos parecen tener un buen rendimiento y no muestran mucho problema para distinguir entre los textos sexistas y no sexistas.

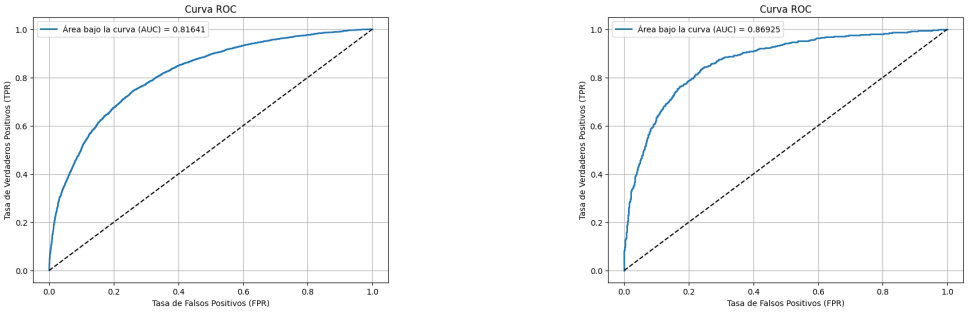
Las figuras 5.1 y 5.2 muestran las curvas ROC de los cuatro modelos. En ellas podemos notar que las curvas se parecen mucho entre sí, como ya habíamos mencionado con el área bajo la curva. Evidentemente, todos los modelos son superiores comparados con un clasificador aleatorio, sin embargo, aún hay espacio para mejorar.



(a) SVM núcleo lineal (b) SVM núcleo RBF

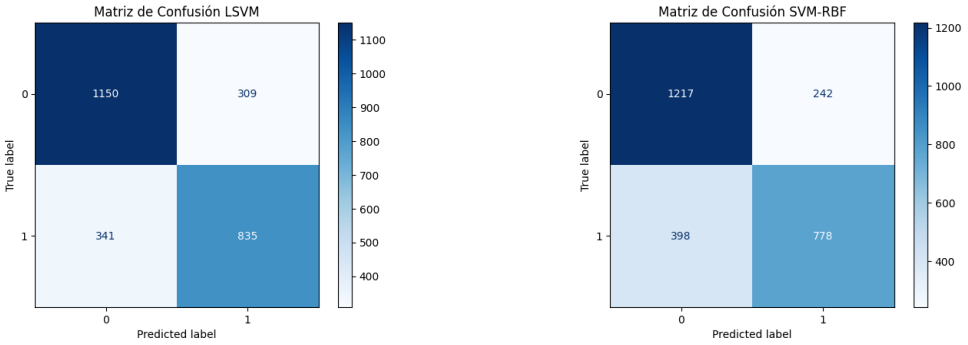
Figura 5.1: Curvas ROC de los modelos de SVM

Ahora analicemos las matrices de confusión de los modelos para poder identificar en cuál de las clases tienden a equivocarse más: comparemos las figuras 5.3 y 5.4.



(a) Regresión Logística (b) SaBERT

Figura 5.2: Curvas ROC de los modelos de Regresión Logística y SaBERT

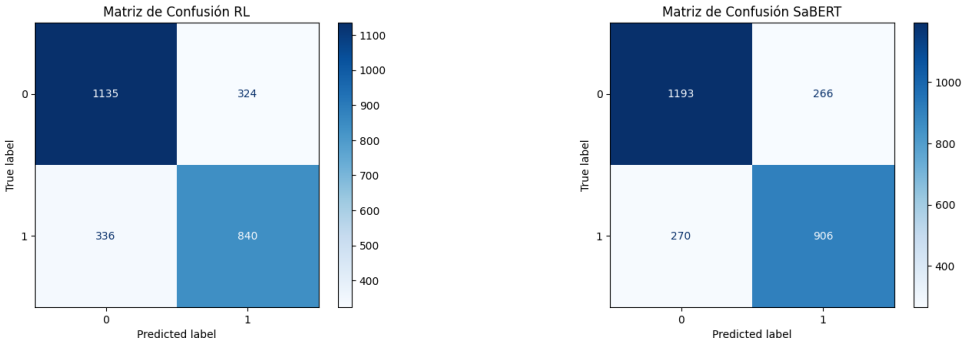


(a) SVM núcleo lineal (b) SVM núcleo RBF

Figura 5.3: Matrices de confusión de los modelos de SVM

Ambos modelos de SVM parece que tienen más dificultades para clasificar los textos sexistas que los no sexistas, incluso parece que al modelo de núcleo RBF le cuesta más que al lineal. Si observamos la sensibilidad de ambos, tenemos que el modelo SVM lineal tiene una sensibilidad de 0.710, mientras que el SVM con núcleo RBF tiene un valor de 0.662, lo cual en el contexto de nuestro problema, podríamos querer una sensibilidad más alta, es decir, preferimos que un tweet no sexista sea clasificado como sexista, sobre clasificar un tweet sexista como no sexista.

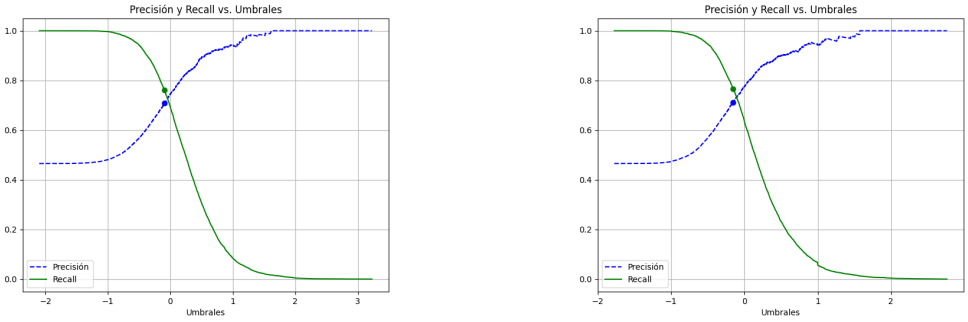
Por otro lado, aunque a SaBERT sigue siendo superior, vemos que el modelo, junto con el de Regresión Logística, son más capaces de clasificar los tweets sexistas correctamente. De hecho, si analizamos su sensibilidad, el modelo de Regresión Logística tiene un valor de 0.714, el más alto después de SaBERT, con 0.770.



(a) Regresión Logística (b) SaBERT

Figura 5.4: Matrices de confusión de los modelos de Regresión Logística y SaBERT

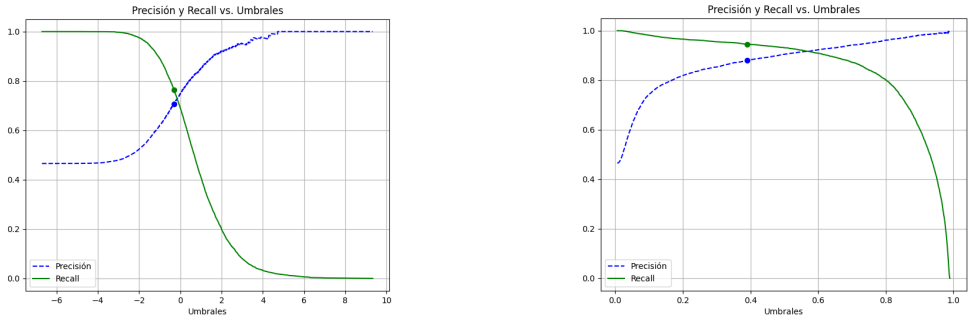
Por último, examinemos las curvas de precisión y sensibilidad contra diversos umbrales de decisión (figuras 5.5 y 5.6).



(a) SVM núcleo lineal (b) SVM núcleo RBF

Figura 5.5: Curvas de Precisión-Sensibilidad de los modelos de SVM

Podemos notar que los modelos de SVM y Regresión Logística tienen curvas de precisión y sensibilidad muy parecidas, esto puede deberse a que los tres fueron entrenados con el mismo espacio de hiperparámetros, por lo que tienen un rendimiento bastante similar. En el modelo SaBERT, observamos que la curva de sensibilidad cae drásticamente a medida que el umbral aumenta, lo que sugiere que el modelo deja de clasificar correctamente muchos de los textos sexistas cuando el umbral es muy alto, así que lo mejor es mantener un umbral ≤ 6 aproximadamente.



(a) Regresión Logística (b) SaBERT

Figura 5.6: Curvas de Precisión-Sensibilidad de los modelos de Regresión Logística y SaBERT

Además, podemos destacar que en todos los modelos se encontró que el mejor balance para la exactitud implica una sensibilidad más alta que la precisión por el contexto del problema que se está tratando, como se había mencionado anteriormente.

5.2. Resultados finales

Con nuestra base de datos aumentada, vamos a reentrenar los modelos siguiendo el mismo procedimiento experimental que ya habíamos explicado, para después analizar los resultados de nuestra propuesta.

Primero comparemos la exactitud obtenida por los modelos en los conjuntos de entrenamiento y de prueba (tabla 5.3).

	SVM Lineal	**SVM RBF**	**Reg Log**	**SaBERT**
Entrenamiento	0.743	0.737	0.740	0.926
Prueba	0.732	0.717	0.732	0.766

Tabla 5.3: Exactitud de los modelos en los conjuntos de entrenamiento y prueba de la base de datos aumentada.

Nuevamente, SaBERT obtuvo los mejores resultados con una exactitud de 0.766, aunque sigue siendo el único modelo con problemas de sobreajuste, parece que continúa desempeñándose bien. Parece ser que el modelo de SVM con núcleo lineal y el

de Regresión Logística tuvieron un desempeño casi idéntico, sin embargo, esta vez la SVM con núcleo RBF tuvo el peor desempeño de los cuatro.

También podemos notar que, pese al incremento en la base de datos, los modelos no tuvieron una mejor exactitud, aunque la diferencia entre los experimentos no fue significativa. Esto puede deberse a varios factores:

- Los datos añadidos no proporcionaron nueva información al modelo.
- Los datos del conjunto de prueba eran más difíciles de clasificar.
- Una mayor cantidad de datos puede llevar a un subajuste, por lo que una búsqueda mayor de hiperparámetros puede ser requerida.

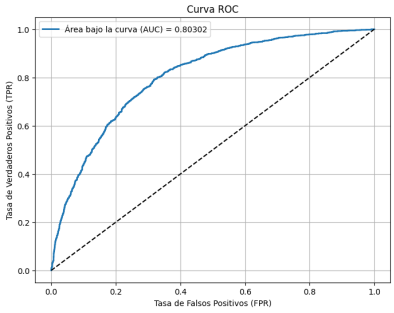
Analicemos ahora los valores del área bajo la curva ROC de los cuatro modelos para poder analizar más a fondo el desempeño de los modelos (tabla 5.4).

	SVM Lineal	**SVM RBF**	**Reg Log**	**SaBERT**
AUC	0.8030	0.7935	0.8051	0.8423

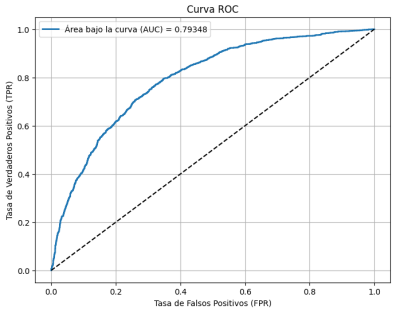
Tabla 5.4: Área bajo la curva ROC de los modelos con la base de datos aumentada.

Al igual que con la exactitud, los modelos tienen un valor ligeramente peor de su área bajo la curva, sin embargo, se sigue considerando un buen desempeño. Como se puede notar, el modelo de SVM con núcleo RBF fue el que disminuyó en mayor medida su eficacia, lo cual se debe a que fue el único al que no se le realizó una búsqueda de hiperparámetros y, mientras más grande sea el conjunto de datos, el modelo irá perdiendo capacidad para identificar los patrones importantes de los datos.

Al observar las curvas ROC de los modelos (figuras 5.7 y 5.8), podemos notar que nuevamente éstas siguen siendo muy similares entre sí, y continúan teniendo un mejor desempeño que un clasificador aleatorio, a pesar de no haber obtenido la mejora que esperábamos.

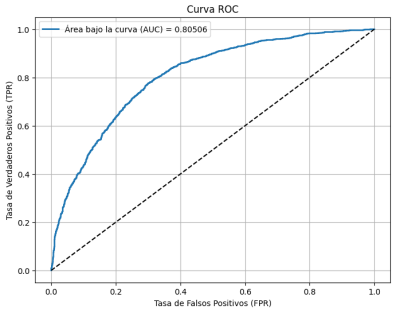


(a) SVM núcleo lineal

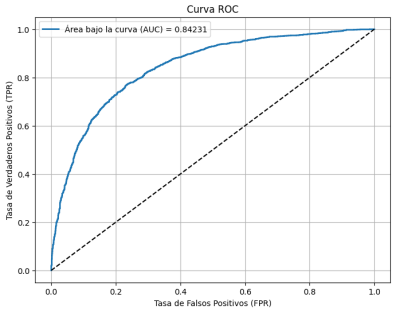


(b) SVM núcleo RBF

Figura 5.7: Curvas ROC de los modelos de SVM con la base de datos aumentada



(a) Regresión Logística



(b) SaBERT

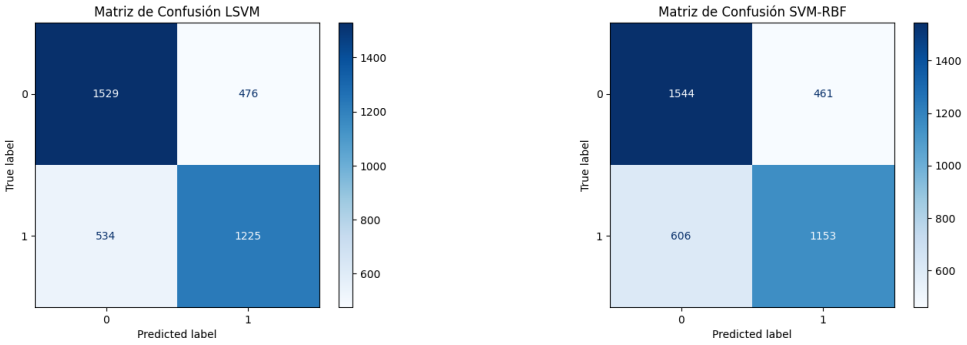
Figura 5.8: Curvas ROC de los modelos de Regresión Logística y SaBERT con la base de datos aumentada

Ahora compararemos las matrices de confusión de los modelos para ver cuál es la clase donde tienden a equivocarse más (figuras 5.9 y 5.10).

Al igual que con la base de datos unificada, los modelos de Máquinas de Vectores de Soporte son los que tienen más problemas para clasificar los textos sexistas correctamente; además, su sensibilidad permanece con valores muy similares a los anteriores, con 0.696 para el núcleo lineal y 0.655 para el núcleo RBF.

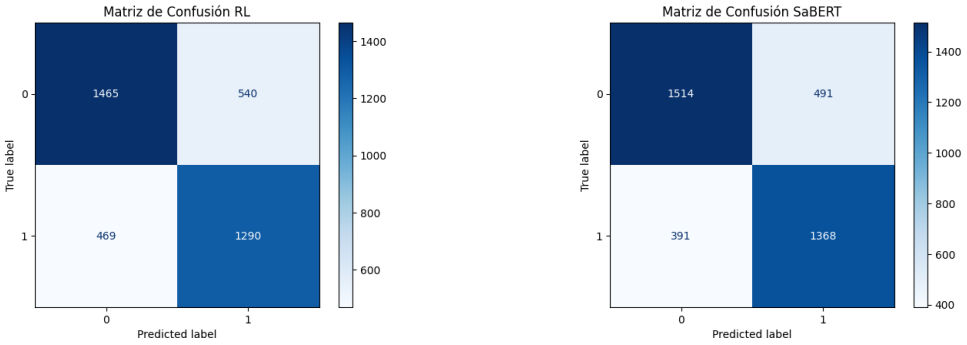
Por otro lado, los modelos de Regresión Logística y SaBERT mejoraron su capacidad de clasificar los tweets sexistas, obteniendo valores de sensibilidad más altos que con la base de datos unificada, con 0.732 y 0.778 respectivamente.

Esto es, en el caso de la Regresión Logística, debido a su simplicidad el modelo



(a) SVM núcleo lineal (b) SVM núcleo RBF

Figura 5.9: Matrices de confusión de los modelos de SVM con la base de datos aumentada.



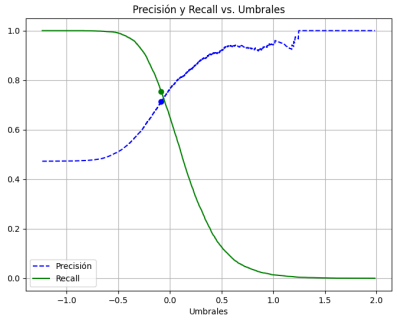
(a) Regresión Logística (b) SaBERT

Figura 5.10: Matrices de confusión de los modelos de Regresión Logística y SaBERT con la base de datos aumentada.

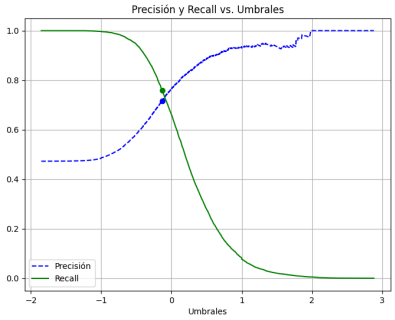
tiende a mejorar con el aumento de datos, pues esto le permite una mayor capacidad de generalización. De manera similar, como ya habíamos mencionado, el mayor problema del modelo SaBERT era su rápida tendencia a sobreajustar, por lo que un mayor número de datos ayuda a regular esto.

Para finalizar, observemos las curvas de precisión y sensibilidad de los modelos contra distintos umbrales de decisión en las figuras 5.11 y 5.12.

Primero, debemos aclarar que la sensibilidad siempre disminuye cuanto más aumenta el umbral de decisión, por lo cual su curva parece más “lisa”; sin embargo, este no es siempre el caso con la precisión, la cual puede tener pequeñas variaciones

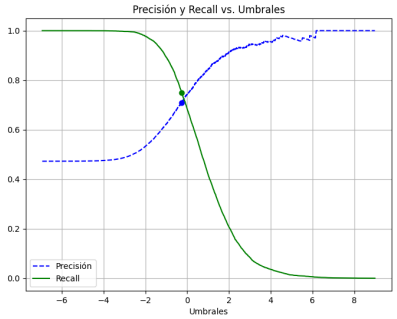


(a) SVM núcleo lineal

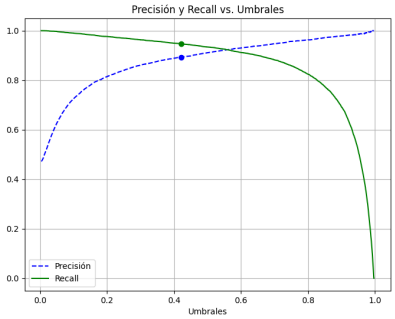


(b) SVM núcleo RBF

Figura 5.11: Curvas de Precisión-Sensibilidad de los modelos de SVM con la base de datos aumentada



(a) Regresión Logística



(b) SaBERT

Figura 5.12: Curvas de Precisión-Sensibilidad de los modelos de Regresión Logística y SaBERT con la base de datos aumentada

en umbrales de decisión cercanos, lo que explica su apariencia ligeramente desigual[7].

Al igual que en los resultados preliminares, las curvas de los modelos de SVM y Regresión Logística son muy similares por la naturaleza de su entrenamiento. Para el modelo SaBERT, parece que aún debemos mantener umbrales por debajo de 0.6 para un mejor desempeño. Nuevamente, para todos los modelos el balance óptimo indica una sensibilidad mayor que la precisión.

Capítulo 6

Conclusiones

En el presente trabajo, se llevó a cabo un estudio exhaustivo sobre la clasificación de tweets sexistas mediante varios modelos de Aprendizaje Automático, siendo estos una SVM con núcleo lineal, SVM con núcleo RBF, Regresión Logística y el modelo preentrenado SaBERT. A lo largo del proceso, se realizaron múltiples experimentos para evaluar el desempeño de cada modelo utilizando métricas de exactitud, precisión, sensibilidad y AUC. Además, se propuso aumentar la base de datos con una traducción de tweets relacionados para aumentar la eficacia de los modelos mencionados.

En los resultados obtenidos, el modelo preentrenado SaBERT demostró ser altamente efectivo en la clasificación de textos sexistas, superando a los modelos tradicionales en términos de precisión y sensibilidad. La SVM con núcleo lineal y la Regresión Logística ofrecieron un desempeño sólido y consistente, siendo buenos puntos de referencia debido a su simplicidad y rapidez de entrenamiento.

Por otro lado, la SVM con núcleo RBF mostró una capacidad superior para manejar relaciones no lineales en los datos, pero su desempeño se vio afectado negativamente cuando se aumentó el tamaño del conjunto de datos. Esto se debió a la limitación de los recursos computacionales a mano, sin embargo, deja lugar al desarrollo del modelo con una búsqueda de hiperparámetros intensiva.

Los modelos complejos como las SVM con núcleo RBF y SaBERT requieren una cuidadosa optimización de hiperparámetros y recursos computacionales significativos. Es esencial encontrar un balance entre la complejidad del modelo y la eficiencia computacional para manejar grandes volúmenes de datos sin sacrificar el rendimiento.

Respecto a nuestra propuesta de aumentar la base de datos con traducciones de textos, podemos concluir que sigue siendo una estrategia potencial de construir bases de datos más grandes y representativas. Debido a la poca existencia de bases de datos relacionadas a la clasificación de sexismo y misoginia en textos, hemos tenido que utilizar un corpus muy similar a uno de los que componían la primera base de datos unificada, por lo que era poco probable que proporcionara otros patrones significativos para la tarea. Sin embargo, el margen de empeoramiento de los modelos no fue significativo, lo que nos indica que la técnica podría resultar beneficiosa si se utilizan datos que hayan tenido otros criterios de recolección.

La inclusión de datos adicionales también resaltó la necesidad de un preprocesamiento riguroso para asegurar que los datos sean relevantes y de calidad, minimizando así el impacto negativo en los modelos de Aprendizaje Automático.

6.1. Trabajo futuro

Para mejorar y extender los resultados obtenidos, es crucial construir bases de datos diversas y de calidad, que recopilen textos con una variedad de lenguaje utilizando distintos enfoques y que abarquen un espectro amplio del sexismo. Para ello, es importante que el proceso sea supervisado por expertos en el tema, con el fin de reducir incosistencias y errores en los datos.

Una mejora en los datos utilizados para el entrenamiento y evaluación de los modelos, permite realizar una optimización más exhaustiva de los hiperparámetros, especialmente para modelos complejos como las SVM con núcleo RBF y los modelos

preentrenados. El uso de técnicas como Randomized Search podría proporcionar mejores resultados[7].

Los resultados obtenidos muestran que, con la optimización adecuada, los modelos de Aprendizaje Automático mantienen una eficacia competitiva con los modelos de Aprendizaje Profundo, por lo que investigar la combinación de modelos tradicionales y modelos preentrenados para aprovechar las fortalezas de ambos enfoques podría ser un área de interés.

Además, es necesario explorar el uso de infraestructuras de computación en la nube y técnicas de paralelización para manejar grandes conjuntos de datos y modelos complejos de manera más eficiente. Esto permitirá una mayor escalabilidad y reducirá los tiempos de entrenamiento, lo que da una mayor oportunidad de investigación a aquellas personas que no poseemos recursos computaciones muy sofisticados.

En resumen, este trabajo proporciona una base sólida para la detección de sexismo y misoginia utilizando diferentes modelos de Aprendizaje Automático y destaca la importancia de considerar tanto la calidad de los datos como la optimización de los modelos. Las oportunidades de trabajo futuro ofrecen un camino claro para mejorar y ampliar los resultados obtenidos, contribuyendo al avance continuo en el campo de la detección del lenguaje de odio.

Bibliografía

[1] Danielle Keats Citron y Helen Norton. «Intermediaries and Hate Speech: Fostering Digital Citizenship for Our Information Age». En: *Boston University Law Review* 91 (2011). URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1764004.

[2] Thomas Davidson et al. «Automated Hate Speech Detection and the Problem of Offensive Language». En: *Proceedings of the 11th International AAAI Conference on Web and Social Media*. 2017, págs. 512-515.

[3] Jacob Devlin et al. «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding». En: *arXiv preprint arXiv:1810.04805* (2019).

[4] Marzieh Fadaee, Arianna Bisazza y Christof Monz. «Data augmentation for low-resource neural machine translation». En: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2017, págs. 567-573.

[5] Elisabetta Fersini, Paolo Rosso y Maria Anzovino. «Overview of the Task on Automatic Misogyny Identification at IberEval 2018». En: *Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018)*. Ed. por Paolo Rosso et al. Vol. 2150. CEUR Workshop Proceedings. co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018), Sevilla, Spain, September 18th, 2018. CEUR-WS.org. 2018, págs. 214-228.

[6] José García-Díaz et al. «Detecting misogyny in Spanish Tweets. An approach based on linguistics features and word embeddings». En: *Future Generation Computer Systems* 114 (ago. de 2020). DOI: 10.1016/j.future.2020.08.032.

[7] Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2018.

[8] IberLEF. *sEXism Identification in Social neTworks (EXIST)*. 2021. URL: http://nlp.uned.es/exist2021/.

[9] Amnistía Internacional. «Amnistía revela alarmante impacto de los abusos contra las mujeres en Internet». En: (2017). URL: https://www.amnesty.org/es/latest/press-release/2017/11/amnesty-reveals-alarming-impact-of-online-abuse-against-women/.

[10] Md Saroar Jahan y Mourad Oussalah. «A systematic review of hate speech automatic detection using natural language processing». En: *Neurocomputing* 546 (2023), pág. 126232. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2023.126232. URL: https://www.sciencedirect.com/science/article/pii/S0925231223003557.

[11] Daniel Jurafsky y James H. Martin. *Speech and Language Processing*. 3.ª ed. Draft version. Pearson, 2023. URL: https://web.stanford.edu/~jurafsky/slp3/.

[12] Simon Kemp. «DIGITAL 2024: GLOBAL OVERVIEW REPORT». En: *Data Reportal* (2024). URL: https://datareportal.com/reports/digital-2024-global-overview-report.

[13] Yann LeCun, Yoshua Bengio y Geoffrey Hinton. «Deep Learning». En: *Nature* 521 (2015), págs. 436-444.

[14] Kate Manne. *Down Girl: The Logic of Misogyny*. Oxford University Press, 2018.

[15] Christopher D. Manning, Prabhakar Raghavan e Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge, England: Cambridge University Press, 2008. ISBN: 978-0521865715. URL: https://nlp.stanford.edu/IR-book/.

[16] Andreas C. Müller y Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*. Sebastopol, CA: O'Reilly Media, 2016. ISBN: 978-1449369415. URL: https://www.oreilly.com/library/view/introduction-to-machine/9781449369880/.

[17] World Health Organization. *Violence against women: Intimate partner and sexual violence against women*. World Health Organization, 2017. URL: https://www.who.int/news-room/fact-sheets/detail/violence-against-women.

[18] Bailey Poland. *Haters: Harassment, abuse, and violence online*. U of Nebraska Press, 2016.

[19] Barbara J. Risman. «Gender as a Social Structure: Theory Wrestling with Activism». En: *Gender & Society* 18.4 (2004), págs. 429-450.

[20] F. Rodríguez-Sánchez, J. Carrillo-de-Albornoz y L. Plaza. «Automatic Classification of Sexism in Social Networks: An Empirical Study on Twitter Data». En: *IEEE Access* 8 (2020), págs. 219563-219576. DOI: 10.1109/ACCESS.2020.3042604.

[21] Francisco Rodríguez-Sánchez et al. «Overview of EXIST 2021: sEXism Identification in Social neTworks». En: *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021)*. Vol. 2943. CEUR Workshop Proceedings. 2021, págs. 195-207. URL: http://ceur-ws.org/Vol-2943/overview_EXIST.pdf.

[22] Eric S. Tellez et al. «An automated text categorization framework based on hyperparameter optimization». En: *Knowledge-Based Systems* 149 (2018), págs. 110-123. ISSN: 0950-7051. DOI: 10.1016/j.knosys.2018.03.003. URL: https://www.sciencedirect.com/science/article/pii/S0950705118301217.

[23] Ashish Vaswani et al. «Attention Is All You Need». En: *Advances in Neural Information Processing Systems* 30 (2017), págs. 5998-6008.

[24] Zeerak Waseem y Dirk Hovy. «Hateful symbols or hateful people? predictive features for hate speech detection on twitter». En: *Proceedings of the NAACL student research workshop*. 2016, págs. 88-93.