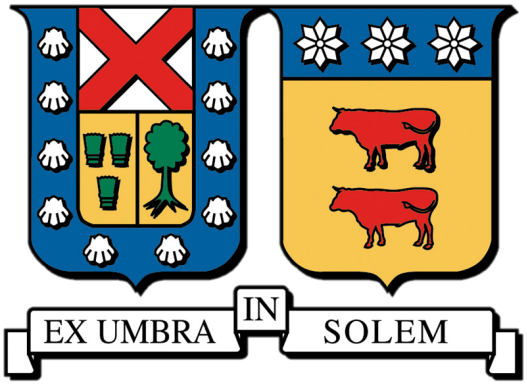
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE INGENIERÍA INFORMÁTICA



DETECCIÓN DE XENOFOBIA Y MISOGINIA EN TWITTER UTILIZANDO REPRESENTACIONES INDEPENDIENTES DEL IDIOMA

SEBASTIÁN ENRIQUE RODRÍGUEZ ORTIZ

Tesis para optar al grado académico de

MAGÍSTER EN CIENCIAS DE LA INGENIERÍA INFORMÁTICA

PROFESOR GUÍA: HÉCTOR ALLENDE OLIVARES

PROFESOR CO-REFERENTE: HÉCTOR ALLENDE CID

ENERO-2024

TITULO DE LA TESIS:

DETECCIÓN DE XENOFOBIA Y MISOGINIA EN TWITTER UTILIZANDO REPRESENTACIONES INDEPENDIENTES DEL IDIOMA

AUTOR:

SEBASTIÁN ENRIQUE RODRÍGUEZ ORTIZ

TRABAJO DE GRADO, presentado en cumplimiento parcial de los requisitos para el Grado de Magíster en Ingeniería Informática de la Universidad Técnica Federico Santa María.

Dr. Héctor Allende O. _____
(Tutor)

Dr. Héctor Allende C. _____
(Co-Tutor)

Dr. Ricardo Ñanculef A. _____
(Evaluador Interno)

Dr. Enrique Cannesa T. _____
(Evaluador Externo)

Dr. Mauricio Solar F. _____
(Presidente Comisión)

Valparaíso, Chile
Enero, 2024

Agradecimientos

En primer lugar, quiero agradecer a mi profesor guía, Dr. Héctor Allende Olivares, por todo su apoyo durante todo el proceso del curso de magíster. Ha sido verdaderamente enriquecedor poder aprender no solo de Machine Learning, sino que también motivarme a aprender sobre otras áreas del conocimiento humano. También quiero agradecer al Dr. Héctor Allende Cid, quien siento que fue una de las primeras personas que vio el potencial en mí para realizar este postgrado. Asimismo, agradecer al Dr. Rodrigo Alfaro por las oportunidades en las cuales él confió en mí.

A mi pareja, Isabel, muchas gracias por todo tu apoyo en lo que han sido estos años. No podría estar en esta etapa final si no fuera por tu paciencia y toda tu ayuda en el día a día. Agradecer también a mi familia, la que constantemente me brindó todo su apoyo, inclusive en los momentos más álgidos.

Finalmente, agradecer a todos mis amigos, los que están y ya no. Partiendo por el grupo INCA, los cuales siempre fueron una motivación para no quedarme atrás y alcanzar el mismo nivel que ellos. Agradecer también al grupo DEEP PUCV, los cuales me han brindado muchas oportunidades para aprender e interactuar con grandes mentes fuera de mi área. Y por último, pero no menos importante, a todos mis amigos, los cuales siempre estuvieron allí para apoyarme ante lo que estuviese pasando.

Resumen

La detección del discurso de odio es un campo de investigación cuyo fin es mitigar el comportamiento malicioso en plataformas en línea. Estas plataformas a su vez generan una gran cantidad de contenido, en el cual moderadores humanos buscan mensajes correspondientes a este tipo de discurso para tomar las acciones correspondientes, pero que no podrían monitorear en su totalidad. Por esta misma razón, herramientas del área del procesamiento de lenguaje natural pueden ser útiles para desarrollar modelos que permitan asistir al proceso de clasificación de los mensajes de forma automatizada.

Si bien el discurso de odio es un problema que afecta a la mayoría de los idiomas que tienen presencia en línea, la gran mayoria de los conjuntos de datos contienen texto en el idioma inglés. Por lo mismo, es importante encontrar una forma de aprovechar dichos recursos, los que pueden ser transferidos a otros idiomas con un mínimo esfuerzo. Para esto se pueden utilizar modelos generadores de vectores de oraciones independientes del idioma. Estas representaciones pueden ser utilizadas como entradas para un modelo de aprendizaje automatizado, entrenado en un conjunto de datos para la detección de discurso de odio en uno o más idiomas. Actualmente, existen modelos basados en redes neuronales profundas, los cuales permiten la generación de estos vectores. Ejemplos de estos pueden ser LASER [1], BERT multilingüe [2] o LaBSE [3].

En esta tesis se propone la utilización de LaBSE como codificador de vectores de oraciones para la tarea de clasificación de discurso de odio en los idiomas inglés y castellano. Este modelo se utilizará junto a otros modelos del estado del arte, con el fin de realizar una comparación del desempeño de los vectores generados para dicha tarea. Además de esto, se realizará la comparación de los modelos presentados en esta propuesta, con el fin de observar la capacidad de clasificación de éstos mediante el ajuste fino de dichas arquitecturas. Se utilizarán dos conjuntos de datos para validar la hipótesis de este trabajo, los cuales consisten en: el conjunto de SemEval2019, tarea 5 [4], y un conjunto de mensajes asociados a la Convención Constituyente en Chile recolectados durante el año 2021. Ambos conjuntos de datos contienen mensajes que provienen de la red social Twitter, y que presentan un contenido misógino y en contra de los inmigrantes.

Palabras clave: Discurso de Odio, Aprendizaje Profundo, Modelos de Lenguaje Multilingües, Clasificación Binaria, Vectores de Oraciones

Abstract

Hate speech detection is a research area whose purpose is to mitigate malicious behavior in online platforms. Additionally, massive amounts of content are generated on these platforms, in which human moderators search for hate speech messages to take proper actions; however, given the sheer number of messages, it is impossible to manually review the entirety of it. Given this, natural language processing tools may be useful to develop classification models to assist the reviewing process in an automated way.

Even though hate speech is a problem that affects most languages with online presence, the majority of the datasets are composed of English texts. Consequently, it is important to find a method to leverage these resources and transfer them to other languages with minimal effort. For this purpose, language independent sentence embeddings can be used. These representations can be used as input for a machine learning model and trained over a hate speech dataset in one or more languages. Currently, there are deep neural network models that are used for generating these sentences embeddings, such as LASER [1], multilingual BERT [2], or LaBSE [3].

In this dissertation, the use of LaBSE as a sentence embedding encoder for hate speech classification tasks in English and Spanish is proposed. The aforementioned model as well as others state-of-the-art models will be used to compare the performance of the vectors generated for this task. In addition to this, the comparison of the models will be performed in an end-to-end approach to observe their binary classification ability. Two datasets are used to validate the hipothesis of this work, which are: SemEval 2019 workshop, task 5 proposed by [4], and messages related to the chilean Constitutional Convention gathered in 2021. Both data sets contain messages that come from Twitter social network, and have misogynistic and xenophobic content.

Keywords: Hate Speech, Deep Learning, Multilingual Language Models, Binary Classification, Sentence Embeddings

Tabla de Contenidos

Lista de Tablas

Lista de Figuras

Nomenclatura

Siglas	Significado
LASER	*Language Agnostic Sentence Embeddings Representations*
BERT	*Bidirectional Embedding Representation from Transformers*
LaBSE	*Language Agnostic BERT Sentence Embeddings*
ONU	Organización de las Naciones Unidas
SVM	Máquina de vectores de soporte
TF-IDF	*Term Frequency - Inverse Document Frequency*
XLM-RoBERTa	*Cross Language Model RoBERTa*
RoBERTa	*Robustly Optimized BERT Pretraining Approach*
infoXLM	*Information Cross Language Model*
LSTM	*Long Short Term Memory*
ALBERT	*A Lite BERT*
XLM	*Cross Language Model*
MUSE	*Multilingual Universal Sentence Embeddings*
PLN	Procesamiento de Lenguaje Natural
ELN	Entendimiento del Lenguaje Natural
MLM	*Mask Language Modeling*
NSP	*Next Sentence Prediction*
mBERT	*multilingual BERT*
TLM	*Translation Language Modeling*
BPE	*Byte Pair Encoding*
BPTT	*Back Propagation Through Time*
D.O.	Discurso de Odio
LR	*Logistic Regression*
LSVM	*Linear Support Vector Machine*
RSVM	*Radial Basis Function Support Vector Machine*
PSVM	*Polynomial Support Vector Machine*
DT	*Decision Trees*
RF	*Random Forest*
ET	*Extremely Randomized Trees*
ROC-AUC	Receiving Operating Characteristic Area Under the Curve

Capítulo 1

Introducción

1.1 Antecedentes y Motivación

El discurso de odio corresponde al acto comunicativo el cual promueve acciones discriminatorias, generando un menoscabo a la dignidad de un grupo de personas. En su mayor parte, estas acciones están basadas en la discriminación según raza, tono de piel, etnicidad, género, orientación sexual, nacionalidad, religión, y otras características de grupos o individuos. Aunque el discurso de odio no es un problema nuevo, sigue siendo relevante en el día de hoy, debido al incremento en su utilización en las plataformas de redes sociales y la anonimidad que éstas proveen, generando un entorno ideal para la proliferación de estas malas prácticas.

Para enfrentar estas acciones discriminatorias, se han desarrollado herramientas automatizadas para la detección de discurso de odio. Para este fin, se han implementado modelos basados en Aprendizaje Automático utilizando algoritmos tradicionales (tales como Bayes Ingenuo, Máquinas de Vectores de Soporte, Árboles de decisión, etc.) y algoritmos basados en redes neuronales profundas. Tradicionalmente, el objetivo de la detección del discurso de odio se ha realizado en un enfoque monolingüe (usando un solo idioma objetivo para su detección). Jahan y Oussalah [5] en su revisión sistemática de la literatura, encontraron que un 51% de los estudios realizados se llevan a cabo para el idioma inglés:

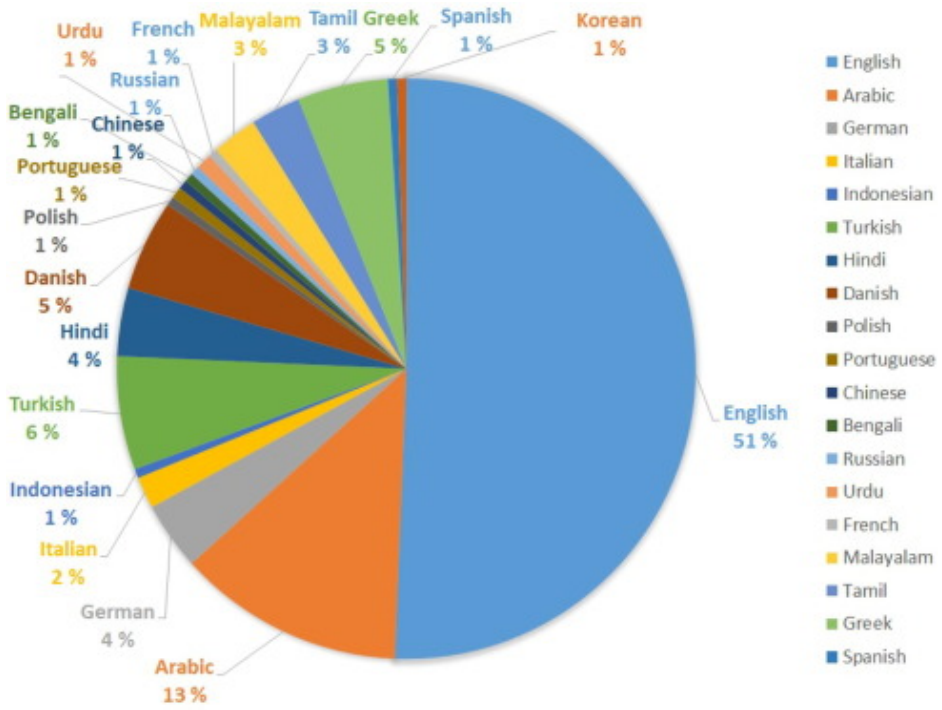


Figura 1.1: Porcentajes de idiomas investigados en la literatura respecto al discurso de odio [5]

Esto es problemático para otros idiomas en los cuales los datos etiquetados son escasos. Sin embargo, existen esfuerzos recientes para proveer set de datos en múltiples idiomas, y múltiples dominios [4], [6]–[8].

Con el desarrollo de los modelos de lenguaje basados en la arquitectura Transformers [2], [9] existen diversas mejoras en tareas relacionadas a la clasificación de texto. Más aún, existen versiones de modelos entrenados en múltiples idiomas a la vez [3], [10], [11], los cuales podrían ser útiles para clasificar textos en varios idiomas. Sin embargo, todavía sería necesario tener textos etiquetados en cada uno de los idiomas para poder hacer el entrenamiento. Para estos modelos, existen versiones entrenadas en múltiples idiomas, los cuales permiten la extracción de representaciones vectoriales en las que una oración traducida en múltiples idiomas debería tener una representación vectorial similar a través de todos los idiomas. Considerando lo anterior, al utilizar estos modelos pre-entrenados en múltiples idiomas, se pueden atacar problemas en un enfoque translingüe (entrenar en varios idiomas, y realizar inferencia en un idioma no presente en el conjunto de datos de entrenamiento). Con estos antecedentes, existe un interés en utilizar las capacidades de estos modelos, los cuales generan representaciones independientes al idioma para utilizarlos en una etapa de pre-procesamiento de los datos de entrada. Esto se realiza con el fin de clasificar texto utilizando otras fuentes de múltiples idiomas en su fase de entrenamiento.

1.2 Definición del problema

Clasificación Automatizada de texto

Para este trabajo se utiliza la definición propuesta por Sebastiani [12] para el proceso de clasificación de texto automatizada. Este proceso corresponde a realizar una asignación de un valor lógico (verdadero o falso) para cada par $\langle d_j, c_i \rangle \in D \times C$, donde D corresponde al dominio de los documentos y $C = \{c_1, \ldots, c_{|C|}\}$ corresponde a un conjunto predefinido de categorías. Si a $\langle d_j, c_i \rangle$ se le asigna un valor de verdadero, indica la decisión de clasificar el documento d_j en la categoría c_i. Por otra parte, asignar el valor falso, indica la decisión de no clasificar d_j bajo c_i.

Ante esto mismo, la clasificación de texto automatizada consiste en la tarea de aproximar la función objetivo desconocida $\Phi : D \times C \rightarrow \{T, F\}$ la cual describe cómo un documento debe ser clasificado. En este caso, la aproximación se lleva a cabo mediante una función $\hat{\Phi} : D \times C \rightarrow \{T, F\}$, la cual corresponde a un clasificador entrenado en datos reales, tal que Φ y $\hat{\Phi}$ coincidan lo más posible.

Por otra parte, las categorías corresponden solo a etiquetas simbólicas, donde no existe un conocimiento adicional de sus significados para ayudar a construir el clasificador. Por esto mismo, solo se utiliza el texto del documento para poder ajustar el clasificador sin utilizar conocimiento exógeno, como por ejemplo el autor del documento, la fecha de publicación, u otro metadato. Para esta tesis, el conjunto de documentos D corresponde a textos obtenidos de la red social Twitter, los cuales pueden ser asignados a dos subgrupos del discurso de odio: misoginia y xenofobia. Adicionalmente, estos textos estarán en dos idiomas: inglés y castellano. Por otra parte, las categorías C corresponden a 2 etiquetas: "Discurso de Odio" y "No Discurso de Odio", configurando así una clasificación binaria.

Representaciones independientes al idioma

Históricamente, la clasificación de texto automatizada se ha realizado con una extracción de características dependientes del idioma y específicamente utilizando un solo idioma en su conjunto de entrenamiento, siendo éste un enfoque monolingüe. Esto implica que, para poder ajustar los clasificadores, se hace una transformación de lenguaje natural a una representación numérica, la cual nos permita obtener la función objetivo. Esta transformación usualmente se realiza generando un vocabulario y luego obteniendo una representación vectorial, la cual puede ser una representación binaria (está o no la palabra); una representación con números enteros (frecuencias de palabras presentes en los textos) o una representación con números reales (frecuencias ponderadas por algún factor). Adicionalmente, a estas representaciones, los enfoques modernos utilizan vectores de palabras u oraciones en conjunto con redes neuronales profundas para la clasificación de textos. El problema que surge con estas representaciones es que al ser dependientes del idioma en el cual

fueron entrenados, estos clasificadores no pueden ser utilizados en otros idiomas, ya que dichos textos quedarían fuera de vocabulario. Una excepción a esto puede presentarse en la traducción de los textos de otros idiomas al idioma en el cual fue entrenado. Sin embargo, esto también conlleva dificultades, ya que es necesario tener un vocabulario suficientemente amplio para poder capturar todas las traducciones, y a su vez, tener suficientes textos para poder entrenar el clasificador utilizando todas las palabras del vocabulario seleccionado. Además, pueden existir problemas de errores introducidos por sistemas automáticos en la traducción.

Actualmente, existen modelos de lenguaje basados en la arquitectura *Transformer* propuesto por [9], los cuales son entrenados en cantidades masivas de textos y actualmente obtienen los mejores rendimientos en diversas tareas. Específicamente estos modelos están basados en BERT (Bidirectional Embeddings Representations from Transformers) [2] los cuales existen en versiones monolingüe (específicas para un idioma) y multilingües. Estas últimas son de crucial interés para este trabajo, ya que nos permiten generar tanto vectores de oraciones, los cuales pueden funcionar para múltiples idiomas, como un modelo de clasificación para la detección de discurso de odio en múltiples idiomas. Para esto, se tendrían que combinar conjuntos de datos en varios idiomas y ajustar un modelo el cual pueda realizar inferencias sobre los idiomas presentes en su etapa de entrenamiento, generando así un modelo con un enfoque multilingüe. Cabe destacar que, existe una posibilidad de que el rendimiento del modelo sea deficiente con ejemplos de idiomas que no estén presentes en la etapa de ajuste. Ya que, si bien el modelo se entrena con múltiples idiomas, no existe una garantía que las representaciones para un mismo texto en distintos idiomas estén en una proximidad en el espacio vectorial. Esto debido a que, en su etapa de preentrenamiento o ajuste fino, no se entrena con una función objetivo explícita que optimice la proximidad de dichas representaciones [13].

Teniendo en cuenta lo anterior, surge el área de la generación de representaciones independientes al idioma. La idea es generar un método de extracción de características el cual permita realizar el ajuste y clasificación, independiente del idioma empleado. Una de las primeras representaciones que se puede utilizar para este proposito, es utilizar una bolsa de grafemas [1] y combinaciones de éstos, para luego generar un vector codificando la frecuencia de aparición de dichas combinaciones de grafemas. A su vez, los enfoques para generar vectores de palabras u oraciones agnósticos al lenguaje consisten en un proceso de alineamiento de los vectores en la función objetivo al momento de hacer el entrenamiento. Con estos antecedentes, existe un modelo en la literatura el cual es conocido como LaBSE (Language agnostic BERT Sentence Embeddings) [3], el cual permite generar vectores de oraciones. Este modelo fue originalmente concebido para la tarea de recuperación de texto en 109 idiomas, obteniendo resultados de vanguardia sobre múltiples conjuntos de datos. En síntesis, en esta propuesta se utilizará LaBSE para generar representaciones indepen-

[1] Unidad mínima de la escritura de una lengua

dientes al idioma sobre mensajes de la red social Twitter. Estos mensajes, los cuales están en dos idiomas: inglés y castellano, previamente son clasificados según la existencia o no de un contenido que puede ser discurso de odio. Luego se utilizarán para explorar si las representaciones generadas por LaBSE y otros modelos permiten la clasificación en enfoques multilingües y translingües.

1.3 Objetivos de la investigación

Objetivos generales

Diseñar e implementar un modelo de LaBSE para la tarea de detección de misoginia y xenofobia en los idiomas inglés y castellano. Este modelo puede ser utilizado en los enfoques de extracción de vectores de características, y en un enfoque de un modelo stand-alone para realizar la clasificación. En estos dos enfoques, se espera obtener mejores resultados que máquinas de aprendizaje tradicionales y máquinas de aprendizaje profundas basados en el uso de vectorizadores de texto tradicionales (frecuencias de n-gramas) y vectores de oraciones.

Objetivos Específicos

- Estudiar el estado del arte para la clasificación de textos en la tarea de la detección del discurso de odio.
- Implementación del algoritmo de LaBSE para la detección de misoginia y xenofobia.
- Implementar los modelos para realizar la comparación con la propuesta.
- Analizar los resultados para las distintas pruebas en enfoques monolingües, multilingües y translingües.

1.4 Alcance de esta investigación e hipótesis

Para esta investigación se utilizará un modelo de red neuronal basado en el modelo de Transfomers llamado Language Agnostic BERT Sentence Embeddings (LaBSE). Este modelo permite generar representaciones independientes del idioma, las cuales tienen potencial para representar textos en múltiples idiomas, con variación mínima en el vector numérico generado a través de los distintos idiomas. Por lo mismo, es de interés efectuar pruebas necesarias con la intención de observar si estas representaciones son capaces de ser utilizadas como datos de entrada para un modelo de aprendizaje supervisado. En este caso, se utilizarán datos de la red social Twitter, específicamente mensajes que son subgrupos del discurso de odio: misoginia y xenofobia. Por esto mismo, se

realizará una comparación del rendimiento de LaBSE con múltiples modelos, los cuales afirman generar representaciones independientes al idioma. Por lo mismo, la hipótesis que se propone para este trabajo es la siguiente:

El uso de LaBSE como modelo de clasificación y generador de representaciones, mejora el desempeño en las métricas utilizadas (Exactitud, Puntaje F1 y área bajo la curva ROC), en enfoques multilingües y translingüe. Para el estudio de comparación, se usarán otros modelos tradicionales y modelos profundos utilizando técnicas tales como: bolsa de palabras, bolsa de grafemas y vectores de oraciones en la tarea de la detección de discurso de odio en inglés y castellano.

1.5 Organización de la tesis

El capítulo 2 provee una revisión de la literatura relacionada con el tema, discutiendo con respecto a la definición del discurso de odio, y los desafíos que existen para la detección monolingüe. Adicionalmente, se revisa la literatura con respecto a la tarea multilingüe y translingüe.

El capítulo 3 presenta el marco teórico, realizando una descripción de las técnicas y arquitecturas que se utilizarán para realizar la detección de xenofobia y misoginia.

El capítulo 4 presenta la metodología, detallando cuáles son los conjuntos de datos utilizados, además de detallar los modelos utilizados para llevar a cabo el contraste de la hipótesis. Finalmente, se presentan las métricas y detalles en las implementaciones de los experimentos a realizar.

En el capítulo 5 se muestran los resultados obtenidos para todos los experimentos detallados en el capítulo 4, seccionados por tarea y evidenciando el desempeño de los modelos.

Finalmente, en el capítulo 6 se presentan las conclusiones, y posible trabajo futuro a realizar relacionado con el tema.

Capítulo 2

Estado del arte

2.1 Definición

Identificar el discurso de odio representa un desafío considerable, dada la ausencia de una definición precisa que permita distinguir claramente qué constituye dicho discurso. Además, cabe señalar que cada persona puede tener una interpretación subjetiva de este fenómeno [14]. Esta incertidumbre se ve influenciada por los sesgos sociales que afectan la percepción y respuesta de los individuos ante el discurso de odio. En consecuencia, una definición precisa debe tener en cuenta las sutilezas lingüísticas y las comunicaciones interpersonales para llevar a cabo una identificación automatizada. La complejidad inherente a este fenómeno y las motivaciones sociales subyacentes deben ser consideradas para comprender por qué resulta desafiante identificar el discurso de odio.

Ante el creciente número de mensajes relacionados con el discurso de odio en plataformas de redes sociales como Twitter[1] y Facebook entre otras, se han implementado esfuerzos para mitigar sus efectos [15]. Las reacciones observadas frente al discurso de odio en estas plataformas pueden atribuirse a diversos factores. Este tipo de discurso presenta un riesgo sustancial y tiene el potencial de causar daño a individuos y comunidades, según el mensaje expresado. Además, estos factores pueden generar un entorno impredecible caracterizado por la hostilidad, disminuyendo el atractivo de la plataforma en donde se alojan los mensajes. Es importante destacar que el discurso de odio va en contra de los valores y principios que defienden las plataformas de redes sociales, las cuales abogan por un espacio inclusivo para todos sus usuarios. Por último, estas redes sociales deben tomar medidas ante posibles implicaciones legales, además de evitar impactos negativos al permitir la difusión libre de este discurso en sus plataformas.

El concepto de discurso de odio ha sido delineado y definido por diversas entidades, investigaciones académicas y debates en línea. Por ejemplo, el Comité de Ministros del Consejo Europeo lo describe como un tipo de expresión que disemina, instiga, avanza o racionaliza la animosidad

[1] Ahora conocido como x.com

racial, xenofobia, anti semitismo, u otro tipo de animosidad enraizada en la intolerancia [16]. Nobata et al. [17], presenta una definicion que engloba expresiones verbales y escritas que muestran hostilidad o desprecio hacia un grupo particular en base a su raza, etnicidad, religión, género, edad, discapacidades, u orientación sexual/identidad de género. Asimismo, las plataformas de redes sociales mencionadas tienen sus propias interpretaciones sobre el discurso de odio. De acuerdo con la política de Twitter, se consideran mensajes con ataques directos a individuos con base en su raza, etnicidad, nacionalidad, afiliación religiosa, orientación sexual, casta, identidad de género, edad o discapacidad estan prohibidos [18]. Por otro lado, Fortuna y Nunes [14] definen el discurso de odio como cualquier lenguaje que ataque o menosprecie, pudiendo incitar a la violencia u odio hacia grupos con base en características específicas tales como apariencia física, religión, ascendencia, origen étnico o nacionalidad, orientación sexual, identidad de género u otras características. Este tipo de discurso puede manifestarse de diversas maneras, incluso de forma sutil o mediante el uso de humor. Dadas estas variaciones en las definiciones, desde el inicio de esta investigación se optó por adoptar la definición de la Organización de las Naciones Unidas (ONU): cualquier tipo de comunicación ya sea oral o escrita, —o también comportamiento— , que ataca o utiliza un lenguaje peyorativo o discriminatorio en referencia a una persona o grupo en función de lo que son, en otras palabras, basándose en su religión, etnia, nacionalidad, raza, color, ascendencia, género u otras formas de identidad [19].

Con respecto a la evolución del discurso de odio en términos de cantidades, Pinker en su libro "En defensa de la Ilustración" [20] muestra métricas asociadas a varios estudios longitudinales con respecto a este tipo de discurso, o en su defecto, a actitudes y fobias relacionadas a posturas poco tolerantes. Ante esto, Pinker utiliza estas métricas como indicadores asociados al discurso de odio y cómo éste va disminuyendo a través del tiempo. A modo de ejemplo, en su libro se presenta cómo las búsquedas de chistes sexistas, racistas y homófobos en la plataforma de Google han ido en descenso desde el 2004 hasta el 2017. Para los chistes sexistas, se presenta una disminución de un 80% a un 20% a la frecuencia del mes, donde más búsquedas relacionadas con esta temática en el año se generó. Asimismo, se presenta que hay una disminución de un 60% a un 10% de chistes racistas, y de un 50% a casi un 0% de chistes homófobos. Otros ejemplos que presenta Pinker son la disminución de delitos de odio en contra de distintas razas, etnias y religiones; la disminución de acciones violentas en contra de mujeres, y también la disminución de opiniones sexistas, homofóbicas y racistas en Estados Unidos. Pinker apunta que esta disminución se debe a múltiples factores, tales como políticas internacionales buscando la igualdad de derecho para mujeres, la despenalización de la homosexualidad y un aumento en la escolaridad de las nuevas generaciones, entre otros.

Finalmente, queda plantear la duda si utilizar una métrica como la disminución de búsquedas en Google con respecto a términos asociados al discurso de odio, sirve como una medida indirecta

para afirmar que el discurso de odio va a la baja. Ante esto mismo, Lupu et al. [21] muestra en su investigación que en un periodo de 1 año y medio, existe una tendencia al alza de mensajes que contienen discurso de odio después de que ocurra un evento polémico. A modo de ejemplo, después de la muerte de George Floyd, en 6 plataformas seguidas por Lupu et al., hay un aumento de casi un 250% en mensajes de odio racistas en las plataformas. Aparte de racismo, los autores comentan que hay un aumento de un 90% en discurso de odio religioso, posteriormente a un asesinato de un general iraní, y también un aumento en mensajes homofóbicos en 100% y un aumento de un 50% en mensajes xenofóbicos posterior a las elecciones de Estados Unidos del año 2020.

2.2 Desafíos en la detección del discurso de odio

El discurso de odio es un problema con efectos que pueden dañar a individuos y a la sociedad. Sin embargo, detectar discurso de odio es una tarea difícil y desafiante debido a razones técnicas, legales y contextuales.

- **Definición:** La definición de discurso de odio puede dificultar la detección debido a las múltiples definiciones propuestas por individuos, organizaciones y plataformas de redes sociales, generando problemas en realizar una estandarización para dicha definición.

- **Matices contextuales:** MacAvaney et al. [23] menciona que discernir si un comentario califica o no como discurso de odio puede ser una tarea difícil, debido al hecho de que la interpretación y propósito de una declaración está vinculada a circunstancias circuncidantes. Un comentario puede ser emitido con una intención de broma o sátira, a diferencia a otro comentario que efectivamente provenga de un origen de odio.

- **Desafíos Tecnológicos**: Sistemas automatizados para detectar discurso de odio se basan en algoritmos de aprendizaje automático, los cuales pueden presentar sesgos si es que fueron entrenados con data sesgada [23]. Adicionalmente, el disacurso de odio puede ser enmascarado utilizando estrategias como errores ortográficos, jergas, o el uso de alternancia de código[2].

- **Calidad del conjunto de datos:** Pueden existir discrepancias en los conjuntos de datos utilizados para el entrenamiento y la evaluación. Estos conjuntos de datos no solo se originan de varias fuentes, sino que también capturan información distinta, inclusive cuando se tratan del mismo fenómeno [23]. Por esta razón puede ser difícil detectar cuáles son las características del discurso de odio presentes en múltiples conjuntos de datos.

- **Obstáculos Legales:** En algunas naciones, la libertad de expresión protege el discurso de odio, por lo que tomar acciones en contra de este tipo de discurso puede ser difícil [19]. Por

[2]La alternancia de código corresponde el empleo alternativo de dos (o más) lenguas o dialectos en un discurso

su parte la globalización y el fácil acceso de internet ha dificultado el combatir el discurso de odio debido a que no hay regulaciones estándar entre los países.

A pesar de estos obstáculos, es importante identificar y oponerse al discurso de odio para prevenir sus efectos dañinos en los individuos y en la sociedad. Para lograr este objetivo, una combinación de sistemas automatizados y supervisión humana es necesaria.

2.3 Enfoques automatizados para la detección del discurso de odio

En años recientes, ha habido un incremento en el uso de enfoques automatizados para la detección del discurso de odio. Esto es debido a principalmente al incremento de dicho discurso en internet, el cual vuelve prohibitivo un enfoque de moderación completamente llevado a cabo por humanos [24]. Muchas de las plataformas de redes sociales prohíben el discurso de odio en sus términos de servicio. Cada reporte tiene que ser revisado manualmente, con el fin de poder hacer cumplir sus reglas. Por lo mismo, métodos automatizados pueden ayudar a acelerar el proceso de evaluación y a no exponer a su personal a mensajes que puedan afectar su salud mental.

Un método propuesto por MacAvaney et al [23], corresponde a un enfoque basado en palabras claves, donde se realiza una búsqueda en los textos mediante el uso de un diccionario de palabras potencialmente ofensivas. Sin embargo, este método tiene sus desventajas, tales como una alta tasa de falsos positivos, y la inhabilidad de poder detectar discurso de odio que no tenga las palabras claves definidas. Por otra parte, métodos que utilizan máquinas de vectores de soporte (SVMs), Bayes Ingenuo, y regresión logística son populares para ser utilizados para la categorización de texto. Asimismo, ademas de estos métodos, el surgimiento en el uso de architecturas de redes neuronales profundas, tales como redes convolucionales, redes recurrentes y redes basadas en Transformers, han progresado el área de la clasificación de texto.

Para identificar discurso de odio, Davidson et al. [24], utiliza un enfoque basado en la extracción de características basadas en el corpus utilizado en el entrenamiento. Estas características corresponden a: Part of Speech, vocabulario ponderado por el metodo TF-IDF, y otros componentes lingüisticos. Con las características definidas, se utilizan SVMs como algoritmo de aprendizaje para hacer el entrenamiento de un clasificador que enfrente múltiples tipos de discurso de odio.

Zimmerman et al. [25] por su parte, definen el uso de un ensamblado de redes neuronales, especificamente de redes convolucionales. La idea tras de la propuesta de Zimmerman, es que múltiples modelos con distintas inicializaciones pueden bajar su error en general, mediante la predicción de múltiples modelos para la tarea de predicción de odio.

2.4 Clasificación de discurso de odio Multilingüe

Si bien existe un interés en proveer de conjuntos de datos monolingues para la detección del discurso de odio que no sean en idioma inglés, tales como: árabe, danes, turco, griego, italiano, francés, castellano, holandés, alemán, portugués, indonesio entre otros; hay que tomar en cuenta que la tarea de construir clasificadores que puedan funcionar con múltiples idiomas a la vez es una tarea reciente. En este caso, uno de los primeros trabajos en esta área corresponde a la propuesta de Ousidhoum et al. [7], el cual presenta el primer conjunto de datos para la detección de discurso de odio en múltiples idiomas. En esta propuesta, se evalúa el uso de múltiples técnicas para atacar el problema desde un punto de vista multilingüe y de multiples tareas para los idiomas inglés, francés y árabe. Por su parte, Ibrohim y Budi [26] investigaron el efecto de aplicar métodos de traducción automatizada asistida por redes neuronales, con el fin de realizar detección de odio en hindi, inglés e indonesio, mediante la comparación de clasificadores que hicieran uso o no de traducciones en su conjunto de entrenamiento.

Ranasinghe y Zampieri[52] utilizaron vectores de palabras translingues, específicamente generados por XLM-RoBERTa (Cross Language Model RoBERTa[10]), para transferir conocimiento de un idioma con altos recursos como el inglés, a un lenguaje de bajos recursos (como bengali, hindi o castellano) para realizar inferencias sobre mensajes en estos idiomas. Para esto, utilizan XLM-RoBERTa como base para entrenar un modelo en inglés, para luego utilizar data de entrenamiento en bengali, hindi y castellano para hacer un ajuste fino del modelo en uno de estos idiomas objetivos. Corazza et al. [27] propuso una arquitectura de red neuronal robusta para identificar el discurso de odio en diferentes idiomas, y evaluó el efecto de distintos tipos de vectorización de los textos, características adicionales y normalización de hashtags y emojis relacionados en el rendimiento de la arquitectura. Vashistha y Zubiaga [28] proponen una arquitecturar jerárquica para redes neuronales profundas para la identificación del discurso de odio en inglés, hindi y una combinación de éstos. Su objetivo era investigar los efectos de las combinaciones de filtros de redes convolucionales o el uso de BERT[2] como entrada para una red recurrente bidireccional basadas en LSTMs.

El workshop OffensEval-2020 [8] es un esfuerzo pionero en analizar el lenguaje ofensivo en un enfoque multilingüe en redes sociales, mediante la disposición de conjunto de datos en 5 idiomas: árabe, danés, inglés, griego y turco. Utilizando el conjunto de datos en inglés, se realizaron anotaciones a tres niveles para identificar si el mensaje tiene un contenido ofensivo, el tipo de ofensa y la audiencia objetivo. En este workshop, distintos participantes contribuyeron a esta tarea mediante la implementación y evaluación de diversos modelos de redes neuronales. Para los idiomas distintos al inglés, la data es anotada solo si tiene un contenido ofensivo. Cabe destacar que para este workshop, más de la mitad de las investigaciones se basaban en el uso de redes

neuronales utilizando la arquitectura Transformers como modelo pre-entrenado base, para luego pasar por el proceso del ajuste fino y técnicas de aumentación de datos para atacar el problema de la detección del discurso de odio. Ante esto mismo, Wang et al. [29] propuso un método multilingüe utilizando XLM-RoBERTa y Ernie para predecir lenguaje ofensivo y el tipo de ofensa subyacente. Wiedemann et al. [30] realizó una evaluación exhaustiva de diferentes modelos basados en la arquitectura Transformers, tales como BERT-base, BERT-large, RoBERTa-base, RoBERTa-large, XLM-RoBERTa y distintas versiones de ALBERT para realizar el ajuste fino de estos modelos para la tarea en el idioma inglés. Uno de sus resultados más notables, consiste en la utilización de un ensamblado de modelos basados en ALBERT para obtener un mejor rendimiento.

2.5 Clasificación de discurso de odio translingüe

En el enfoque translingüe donde hay pocos o nulos conjuntos de datos en el idioma objetivo, es un concepto nuevo en el dominio de la detección del discurso de odio. Algunos de los trabajos más recientes han discutido el uso de modelos translingües, en conjunto con métodos de aprendizaje basados en *few-shot* o *zero-shot* learning, para realizar la identificación del lenguaje ofensivo en un idioma no visto en su etapa de entrenamiento. Stappen et al. [31] propone el uso de una arquitectura para atacar el problema de la detección de odio con enfoques monolingüe y translingüe. En este caso, se realiza la detección entre los idiomas inglés y castellano, utilizando un modelo BERT o XLM como extractor de características. Dada la naturaleza del modelo Transformer, las representaciones obtenidas de estos modelos son contextuales sin la utilización del proceso de ajuste fino. Luego, estas representaciones son utilizadas como entrada para una arquitectura propuesta por los autores, la cual se entrena para la clasificación del discurso de odio en los enfoques anteriormente mencionados.

Aluru et al. [32] analizó el discurso de odio en un enfoque multilingue utilizando 9 idiomas obtenidos de 16 conjuntos de datos públicos relacionados a la tarea de mensajes de odio. En un enfoque basado en aprendizaje *few-shot*, se utilizan $n-1$ idiomas para el conjunto de entrenamiento, y un n-esimo idioma como el idioma objetivo. Luego, se utilizan vectores basados en modelos LASER [1] y BERT, utilizando un enfoque incremental para incluir muestras del idioma objetivo en el proceso de entrenamiento. Por su parte Pamungkas et al. [33] emplean un mecanismo de traducción automatizada, y propone el aprendizaje en conjunto para dos arquitecturas, una de las cuales utiliza red recurrentes LSTM [34] y embeddings basados en MUSE [35], mientras que la segunda arquitectura corresponde a un BERT Multilingue para identificar contenido odioso entre 11 conjunto de datos de uso público, a través de 7 diferentes idiomas. Para configurar un enfoque de aprendizaje *zero-shot*, los investigadores consideraron el idioma inglés como el conjunto de entrenamiento, y el resto de los idiomas como el conjunto de prueba. Si bien este modelo tiene

una respuesta robusta en la tarea translingüe, tiene una limitación atribuida al ruido excesivo en los datos, debido al módulo de traducción el cual genera errores que son propagados a través de los distintos segmentos de la arquitectura propuesta.

Capítulo 3

Marco Teórico

3.1 Transformer

La arquitectura Transformer, propuesta por Vaswani et al. [9], surge como una propuesta innovadora y disruptiva en el campo del procesamiento de lenguaje natural (PLN), ofreciendo una alternativa sólida a las redes neuronales recurrentes y redes convolucionales para el modelado de secuencias. Esta arquitectura ha generado un impacto significativo en diversas áreas del PLN, incluyendo la generación de texto, la traducción automática y el Entendimiento del Lenguaje Natural (ELN), redefiniendo la manera en que se abordan estas tareas. El Transformer se fundamenta en la combinación de capas densas y el mecanismo de auto-atención, los cuales posibilitan la captura eficaz de las relaciones entre los elementos presentes en una secuencia[1]. Integrando estos componentes, la arquitectura del Transformer se organiza en una estructura de codificador-decodificador (Encoder-Decoder networks) [36], que facilita la modelación tanto de secuencias de entrada como de secuencias de salida.

El mecanismo central de la arquitectura Transformer es la auto-atención, la cual habilita al modelo para evaluar la relevancia relativa de los distintos elementos que componen una secuencia durante los procesos de codificación y decodificación. Los pesos de atención asignados a cada elemento de la secuencia se determinan considerando las interacciones con todos los demás elementos de la misma. Para una secuencia de entrada de longitud n, el mecanismo de auto-atención calcula un conjunto de pesos de atención, denominados puntajes de atención, para cada elemento. Estos pesos de atención se determinan utilizando la siguiente ecuación:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}})\right)V \tag{3.1}$$

[1]Para los propósitos de esta tesis, se define una secuencia como la disposición ordenada de palabras, sub palabras o grafemas de izquierda a derecha

en la formulación del mecanismo de auto-atención, se definen $Q \in \mathbb{R}^{n_q \times d_q}$, $K \in \mathbb{R}^{n_k \times d_k}$ y $V \in \mathbb{R}^{n_v \times d_v}$ como las matrices que representan la secuencia de entrada proyectada en el espacio de embeddings del modelo, donde d_k corresponde al tamaño del vector asociado a los embeddings. Es importante resaltar que, en el contexto de la auto-atención, se establece que $n_q = n_k = n_v$ y $d_q = d_k = d_v$, dado que Q, K y V reflejan la secuencia de entrada y, en este escenario, se aplica un escalamiento de los puntajes de atención obtenidos.

En el contexto de la arquitectura del Transformer, el módulo de auto-atención se conoce como *Multi-Head Attention* (Atención con Múltiples Cabezas), en el cual se emplea una capa densa para procesar los distintos valores de entrada que alimentan la función de atención. Este proceso se replica h veces con el propósito de que el modelo considere múltiples aspectos de las entradas al evaluar la alineación. Una vez calculada la atención para cada cabeza, estas se concatenan y posteriormente atraviesan nuevamente por una capa densa, logrando obtener una representación que constituye una combinación lineal de todas las salidas provenientes de las múltiples cabezas atencionales. Este procedimiento se visualiza en la figura 3.1.

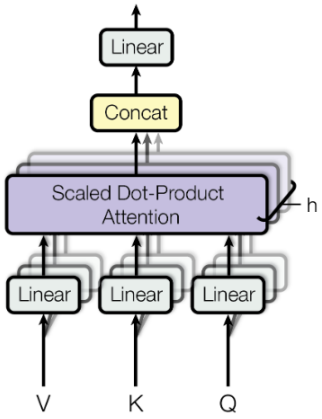


Figura 3.1: Cabeza multi-atencional

Como previamente expuesto, la arquitectura del Transformer integra múltiples instancias de capas densas y cabezas de atención con múltiples cabezas, con el propósito de generar capas tanto de codificadores como de decodificadores. Estas últimas se superponen en múltiples iteraciones para configurar la estructura arquitectónica que se ilustra en la figura 3.2.

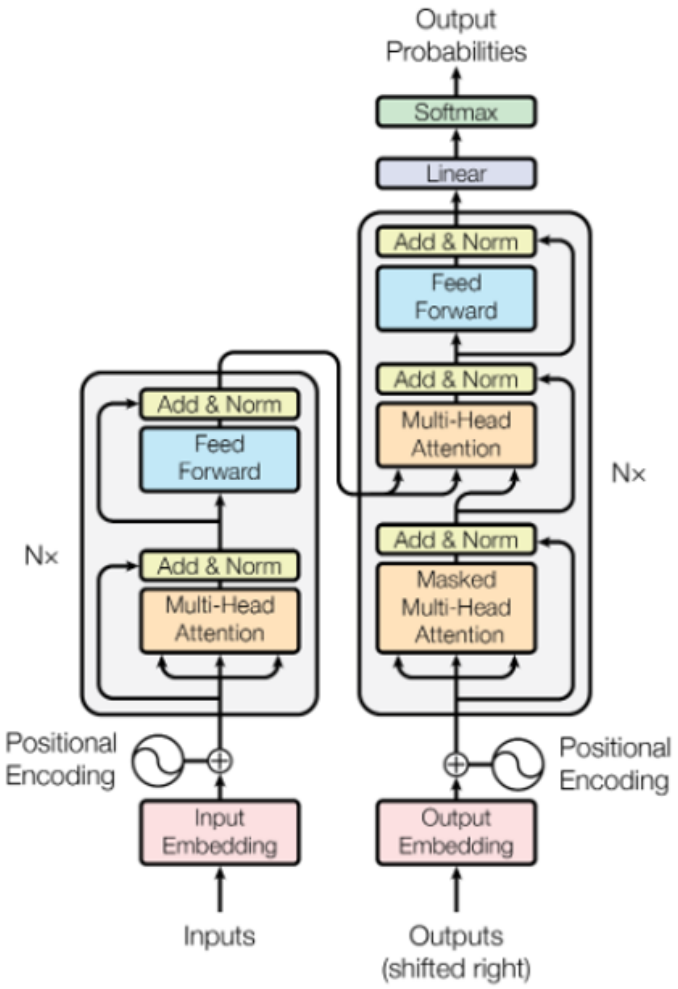


Figura 3.2: Arquitectura Transformer

Es relevante subrayar que, dentro de la formulación del Transformer, se considera el modelado de los embeddings de entrada en conjunto con la incorporación de información posicional en dichos embeddings. En lo que respecta al embedding de entrada, éste se aprende en conjunto con toda la arquitectura, posibilitando así la generación de una representación para cada token contenido en el vocabulario del modelo. Por otro lado, la inclusión del encoding posicional permite abordar la estructura secuencial de los tokens en la entrada, ya que es imperativo incorporar esta información sobre la posición de cada token. Esta necesidad surge debido a la ausencia de recurrencia o convolución que, por sí solas, modelen dicho orden en la secuencia.

3.2 BERT

BERT (Bidirectional Encoder Representations from Transformers) [2] constituye un modelo de lenguaje que ha alcanzado un rendimiento sobresaliente en diversas tareas de Procesamiento del Lenguaje Natural (PLN). Esta arquitectura se fundamenta en la utilización del codificador perteneciente a la estructura Transformer. La idea detrás de BERT radica en entrenar este modelo profundo bidireccional en un extenso corpus de texto no etiquetado a través de tareas de pre-entrenamiento. Posteriormente, se realiza un ajuste fino de los pesos del modelo para una tarea específica, como análisis de sentimiento o etiquetado de partes del discurso, entre otras. Las tareas de pre-entrenamiento de BERT comprenden la reconstrucción de oraciones mediante tokens enmascarados y la predicción de si una oración continúa a otra, aprovechando la atención contextual de las palabras tanto en la dirección izquierda a derecha como en la dirección inversa. Este proceso de entrenamiento bidireccional potencia el entendimiento y contexto que BERT tiene acerca del uso de las palabras. Este concepto se ilustra en la figura siguiente:

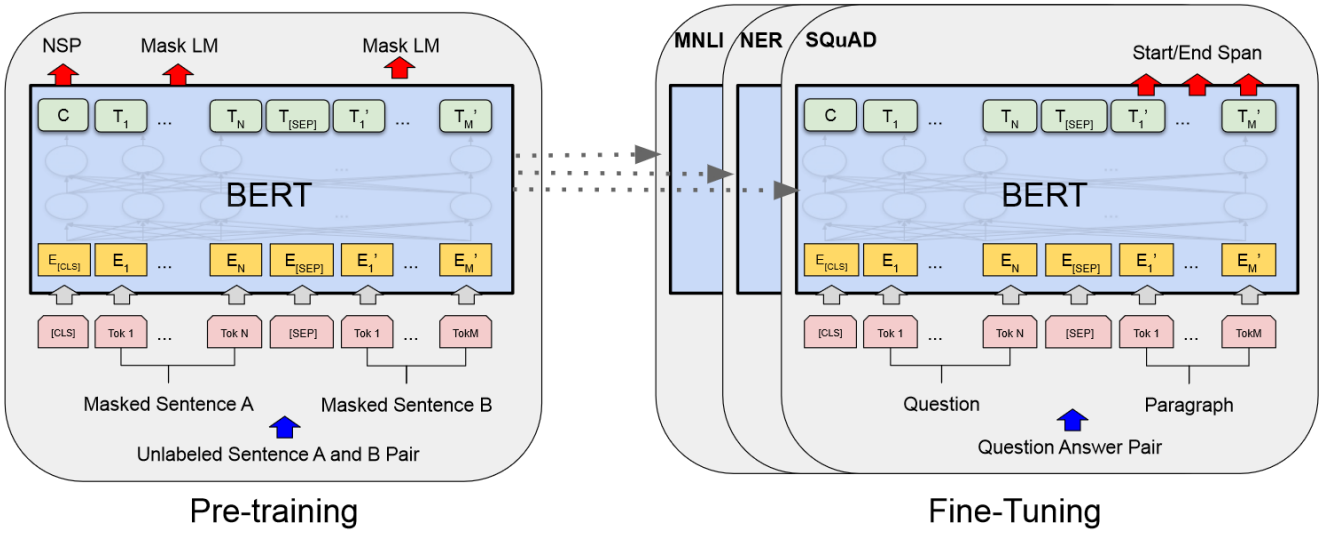


Figura 3.3: Pre-entrenamiento y Ajuste fino de BERT [2]

3.2.1 Representaciones de entrada

Para procesar lenguaje natural de manera efectiva, es esencial llevar a cabo un procedimiento de tokenización sobre las oraciones de entrada, adaptándolas a un vocabulario predeterminado para el modelo BERT. En esta instancia, la tokenización implica la aplicación del algoritmo WordPiece [37], un modelo no supervisado que aborda la segmentación de palabras en subpalabras. Estas subpalabras se generan considerando el uso y agrupación de grafemas en un corpus. La metodología de construcción de este vocabulario busca minimizar la presencia de palabras fuera de éste al descomponer palabras poco comunes o ausentes en el corpus de entrenamiento en distintas

subpalabras presentes en el vocabulario. Un ejemplo de este proceso, utilizando el tokenizador multilingüe de BERT, es la segmentación de la palabra "jugaban" en "juga" y "##ban", donde el uso del símbolo "#" indica que dicho segmento es continuación del token previo.

En adición al uso del vocabulario de subpalabras, se introducen dos tokens especiales: un token de clasificación al inicio de la oración tokenizada (`[CLS]`) y un token de separación de oraciones (`[SEP]`). Respecto a la tokenización definida, una parte fundamental de la arquitectura de BERT radica en los embeddings que emplea para representar cada subpalabra del vocabulario. Estos incluyen los embeddings de segmentos, los cuales indican si la subpalabra pertenece a la primera o segunda oración, así como los embeddings de posición que codifican el orden secuencial de cada subpalabra. Finalmente, estos embeddings se agregan para generar la representación numérica que actúa como entrada para el proceso de pre-entrenamiento y ajuste fino de BERT. Una ilustración de este proceso se presenta en la figura 3.4.

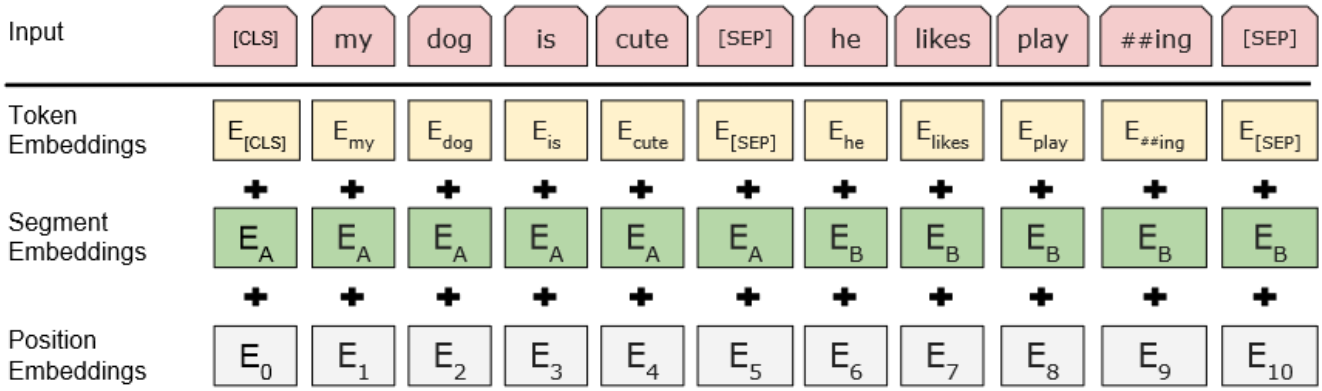


Figura 3.4: Representación de entrada de BERT [2]

3.2.2 Pre-entrenamiento y Ajuste fino

Conforme se expuso anteriormente, el pre-entrenamiento de BERT comprende la realización de dos tareas no supervisadas, a saber, el *Masked Language Modeling* (MLM) y la *Next Sentence Prediction* (NSP). La primera de estas tareas consiste en una tarea de reconstrucción de las oraciones de entrada, en la cual el 15% de las palabras en las oraciones se sustituyen con un token especial (`[MASK]`). En esta tarea, al emplear las subpalabras reemplazadas en el proceso de generación de datos de entrenamiento, el modelo se ajusta realizando la predicción de la palabra enmascarada a través de una capa de salida con V neuronas y una función de activación softmax, siendo V el tamaño del vocabulario.

Por otra parte, los creadores de BERT resaltan la importancia de la tarea de NSP para que el modelo adquiera la capacidad de comprender la relación existente entre un par de oraciones,

permitiendo así la transferencia de este conocimiento a tareas que demandan tal comprensión, como *Question Answering* o *Natural Language Inference*. En términos de implementación, el generador de datos de entrenamiento realiza un muestreo con una probabilidad del 50% para elegir pares de oraciones `A` y `B` que sean continuación de la una de la otra, y un 50% de probabilidad de seleccionar lo opuesto. Posteriormente, para ajustar los pesos, se obtiene la representación del token `[CLS]` y se utiliza una capa de salida con una sola neurona y función de activación sigmoide para predecir si la oración `B` es efectivamente la continuación de la oración `A`.

3.2.3 BERT Multilingüe (mBERT)

Como tal, las tareas mencionadas previamente pueden ser empleadas en cualquier idioma, siempre que se disponga del corpus recolectado necesario para llevarlas a cabo. Por consiguiente, existen diversas variantes de los modelos BERT especializadas en un idioma específico, como el caso del castellano (BETO) [38], francés (FlauBERT) [39], portugués (BERTimbau) [40] y alemán (GottBERT) [41], entre otros. Además, en el artículo original de BERT, los autores presentan las versiones de BERT en inglés, chino y una versión multilingüe (mBERT). Esta última está entrenada con un corpus recopilado de Wikipedia, que incluye artículos de 102 idiomas distintos y un vocabulario compartido de 110 mil tokens entre todos los idiomas.

Es importante resaltar que, aunque se cuente con un corpus multilingüe masivo, durante la fase de pre-entrenamiento de mBERT se emplean pares de oraciones correspondientes al mismo idioma. Esto implica que se desaprovecha el potencial para entrenar un modelo capaz de capturar las relaciones en el uso de los tokens entre diferentes idiomas, generando conocimiento para cada idioma a través de un vocabulario común. Uno de los primeros problemas evidenciados al utilizar mBERT es que su rendimiento tiende a ser inferior al realizar tareas de ajuste fino en comparación con los modelos de BERT entrenados para idiomas específicos [42], [43]. Otro de los problemas identificados radica en que el proceso de generación de corpus presenta desafíos de desequilibrio de datos, donde idiomas como el inglés, castellano o francés, entre otros, están sobrerrepresentados, lo que puede dar como resultado un rendimiento deficiente del modelo para idiomas subrepresentados [44]. Sin embargo, mBERT representa uno de los primeros modelos que permiten el ajuste fino para corpus multilingües.

3.3 XLM-RoBERTa

XLM-RoBERTa (Cross Language Model RoBERTa) es un modelo basado en la arquitectura de BERT, considerando las optimizaciones propuestas en "A Robustly Optimized BERT Pretraining Approach (RoBERTa)" [10], donde se plantea que BERT está sub-entrenado. Las diferencias

principales entre BERT y RoBERTa son las siguientes:

- RoBERTa emplea un corpus de entrenamiento más extenso y de mayor calidad en comparación con el corpus utilizado por BERT.
- La tarea de NSP se elimina, dado que se han observado empíricamente mejores resultados durante el proceso de *fine-tuning* al prescindir de ella.
- La tarea de MLM se modifica para realizar un enmascaramiento dinámico, generando múltiples combinaciones de enmascaramiento de tokens para un mismo ejemplo de entrenamiento.

Con estos detalles presentados, la principal diferencia que existe entre mBERT y XLM-RoBERTa radica en el tamaño del corpus utilizado por ambos modelos. El primero hace uso de un corpus obtenido de Wikipedia a partir de los 102 idiomas con mayor cantidad de artículos, mientras que el segundo emplea un corpus obtenido mediante web scraping, conocido como CommonCrawl[2], notablemente más extenso en términos de magnitud en comparación con el corpus de Wikipedia. La figura 3.5 muestra la diferencia en la cantidad de datos utilizados para llevar a cabo el entrenamiento de XLM-Roberta.

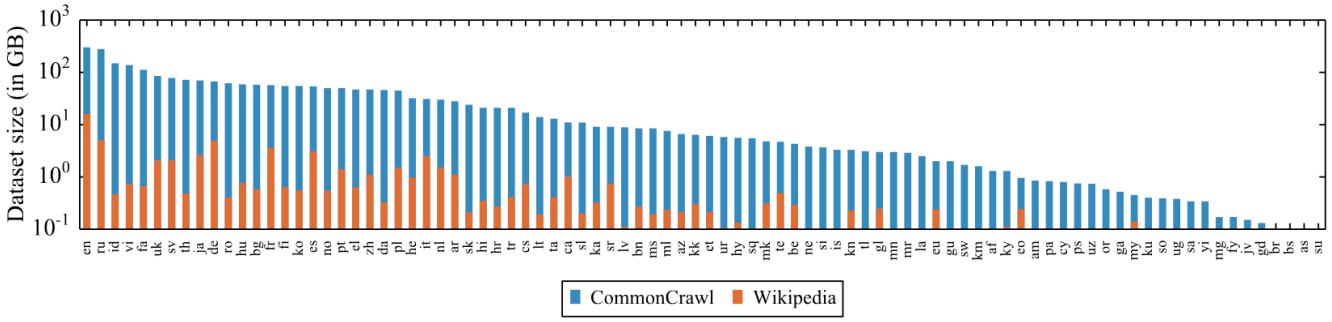


Figura 3.5: Comparación entre el tamaño en GBs entre 88 idiomas presentes en corpus de Wikipedia y CommonCrawl [45]

Adicionalmente, otra diferencia que existe entre mBERT y XLM-RoBERTa es que en el caso de XLM-RoBERTa, se implementa un muestreo de oraciones que considera los diversos lenguajes mediante el uso de una distribución multinomial. En contraposición, en el caso de mBERT no se contempla un proceso análogo, lo que puede resultar en un uso menos frecuente de oraciones correspondientes a idiomas sub-representados en el corpus durante la etapa de procesamiento.

[2]http://index.commoncrawl.org

3.4 LaBSE

El modelo de *Language Agnostic BERT Sentence Embedding* (LaBSE) [3] corresponde a un modelo pre-entrenado que emplea la misma arquitectura que BERT para la generación de embeddings de oraciones translingües. En este sentido, LaBSE se utiliza para producir representaciones de oraciones que posibilitan la comparación entre oraciones en diversos idiomas. La distinción primordial entre LaBSE y los modelos previamente mencionados radica en que LaBSE considera tareas de pre-entrenamiento que utilizan pares de oraciones paralelas en diferentes idiomas. Además, el régimen de entrenamiento de LaBSE incorpora una segunda fase de pre-entrenamiento que implica alinear los vectores de oraciones generados por el modelo, fortaleciendo así el aprendizaje adquirido en la primera etapa. Este proceso de alineación contribuye a obtener resultados de vanguardia en tareas de *bi-text retrieval/mining*. Otra disparidad entre los modelos previamente mencionados consiste en que LaBSE está diseñado para generar embeddings a nivel de oración, mientras que mBERT y XLM-RoBERTa generan representaciones a nivel de tokens.

Los corpus utilizados para llevar a cabo el entrenamiento de LaBSE consisten en los corpus de Wikipedia y CommonCrawl mencionados anteriormente, a fin de obtener datos monolingües. Adicionalmente, los autores emplean pares de traducciones en diversos idiomas obtenidos mediante *Web Scraping*. Estos pares de oraciones se evalúan a través de la traducción al idioma inglés, seguida de una medición de calidad de la traducción en dos etapas:

1. Evaluación según el criterio de evaluadores humanos en una muestra del corpus, quienes valoran la calidad de la traducción como buena o mala.

2. Utilización de un método automatizado de puntuación de pares de oraciones [46], el cual es ajustado utilizando la muestra de la etapa previa para garantizar una coincidencia mínima del 80% con las evaluaciones humanas para oraciones consideradas buenas.

3.4.1 Primera etapa de pre-entrenamiento: MLM y TLM

Para LaBSE, se ejecuta una fase de pre-entrenamiento análoga a la empleada en los modelos mencionados previamente; sin embargo, en este caso se incorpora la tarea de *Translation Language Modeling* (TLM). En esta tarea específica, es necesario considerar un par de oraciones en distintos idiomas que constituyan traducciones directas. Posteriormente, en un procedimiento análogo al utilizado en la tarea de MLM, se enmascaran subpalabras en ambos pares de oraciones. La finalidad del TLM es que el modelo debe aprender a reconstruir las oraciones, teniendo en cuenta tanto el contexto de la oración en su idioma original como su traducción. En la figura 3.6 se presentan dos ejemplos de las tareas de MLM y TLM, demostrando las diferencias entre ambas.

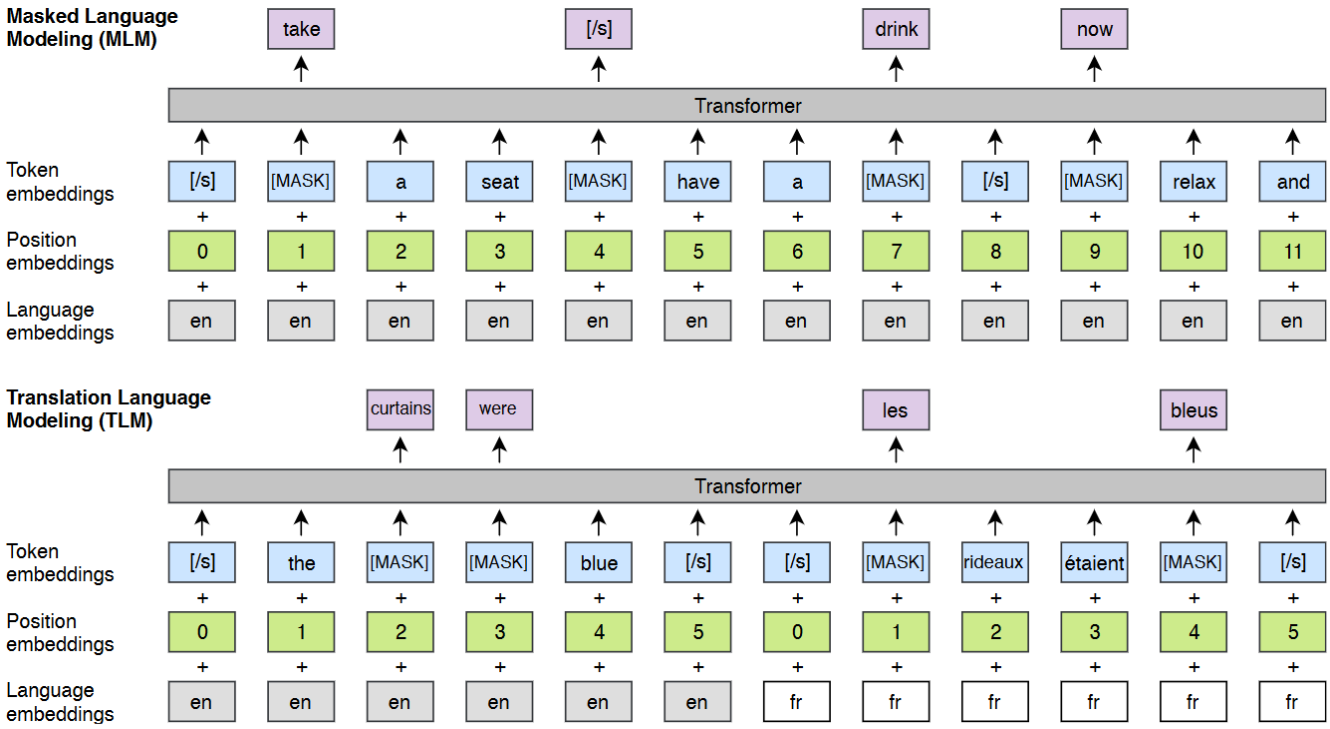


Figura 3.6: Diferencias entre Masked Language Modeling y Translation Language Modeling

Cabe destacar que, tal como se ilustra en la figura anterior, la tarea de TLM demanda una adaptación en la generación del embedding de entrada. En este caso, además de los vectores de tokens y vectores posicionales de oración, resulta necesario incorporar vectores de identificación de idioma que entreguen al modelo una guía respecto al idioma que se está modelando en dicha oración. Esta consideración se efectúa debido a que el modelo en cuestión emplea un vocabulario común para todos los idiomas en los que se lleva a cabo el entrenamiento. En consecuencia, una subpalabra correspondiente a un idioma A debería manifestar un comportamiento diferente respecto a esa misma subpalabra en un idioma B.

3.4.2 Segunda etapa de pre-entrenamiento: Alineamiento de vectores

Una vez finalizada la primera etapa de pre-entrenamiento, el modelo resultante se emplea para inicializar una arquitectura de Transformer Dual, en la cual ambos Transformers comparten sus pesos. Esta acción constituye el primer paso en la segunda etapa de pre-entrenamiento, mientras que la segunda etapa implica la alineación de los vectores generados para pares de oraciones que son mutuas traducciones. Para llevar a cabo esta alineación, se requiere definir una función de pérdida que permita evaluar si los vectores generados por el modelo para estos pares de oraciones son lo más similar posible. En esta línea, los autores proponen la utilización de *Additive Margin Softmax* [47] para entrenar el modelo con el objetivo de generar vectores alineados. La función de

pérdida se calcula de la siguiente forma:

Una vez que se completa la primera etapa de pre-entrenamiento, el modelo resultante se utiliza para inicializar una arquitectura de Transformer Dual, donde ambos Transformers comparten los pesos. Esto se realiza como primer paso de la segunda etapa de pre-entrenamiento, la cual consiste en alinear vectores de oraciones generados para pares de oraciones que son traducciones directas entre sí. Para realizar este alineamiento es necesario definir una función de pérdida, la que permita evaluar si los vectores generados por el modelo para estos pares de oraciones sean los más similares posibles. Por lo mismo, los autores proponen el uso de *Additive Margin Softmax* [47] para entrenar el modelo para generar vectores alineados y en este caso se computa:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\frac{e^{\phi(x_i,y_i)-m}}{e^{\phi(x_i,y_i)-m}+\sum_{n=1,n\neq i}^{N}e^{\phi(x_i,y_n)}}, \tag{3.2}$$

donde x_i e y_i corresponden a un par de oraciones que son traducciones directas, y y_n corresponde a ejemplos de oraciones que no son traducciones directas de las oraciones anteriores. La función ϕ representa una función de similitud, corresponde al producto punto entre los vectores. Es importante señalar que estos vectores se obtienen a partir del token `[CLS]` tras pasar por la totalidad de la arquitectura. En la figura 3.7 se exhibe el diagrama de la arquitectura dual y el flujo aplicado.

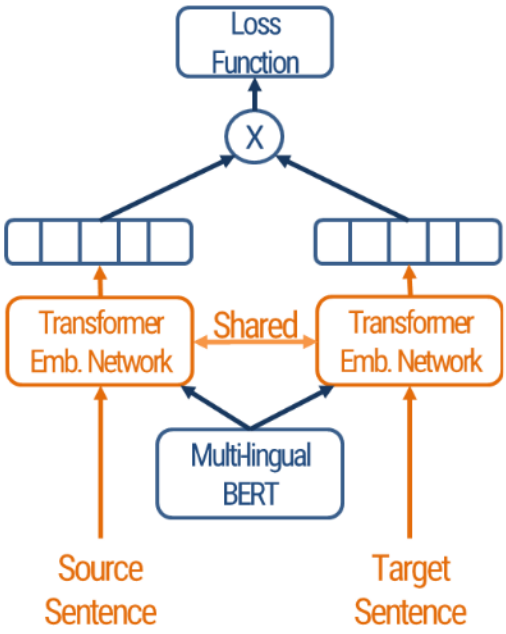


Figura 3.7: Arquitectura dual de LaBSE

3.5 InfoXLM

El modelo InfoXLM (Information Cross Language Model) propuesto por Chi et al. [11] se fundamenta en la arquitectura y corpus propuestos por XLM-RoBERTa, incorporando tareas de pre-entrenamiento adicionales. Concretamente, el pre-entrenamiento de InfoXLM abarca las tareas de Masked Language Modeling (MLM) y Translation Language Modeling (TLM), además de introducir la adición de la tarea de aprendizaje contrastante translingüe (*Cross-Lingual Contrastative Learning*) la cual se computa por:

$$\mathcal{L}_{X_L C_O} = -log \frac{e^{\phi(x_i, y_i)}}{\sum_{y_n \in \mathcal{N}} e^{\phi(x_i, y_n)}}, \tag{3.3}$$

donde ϕ, x_i, y y_i denotan la función de alineamiento y los pares de oraciones que son traducciones directas, como se detalla en la sección 3.4.2. La principal distinción en esta función de pérdida se encuentra en el término del denominador, donde se lleva a cabo una sumatoria sobre una cola de elementos que no constituyen traducciones directas, representada por el conjunto $\mathcal{N}$. En consecuencia, un hiperparámetro crucial durante el entrenamiento de InfoXLM es el tamaño de dicha cola, $|\mathcal{N}|$, un valor especificado por los autores como un total de $131,072$ oraciones. Durante la fase de entrenamiento, a medida que se procesan pares de oraciones, éstos se incorporan a la cola $\mathcal{N}$, y ejemplos antiguos son retirados para mantener constante el tamaño de la cola.

3.6 LASER

LASER (Language Agnostic Sentence Embedding Representations), propuesto por Artetxe y Schwenk [1], corresponde a un modelo basado en una arquitectura Codificador-Decodificador, el cual utiliza redes neuronales recurrentes, más específicamente *Long Short-Term Memory* (LSTMs) [34] como unidades recurrentes. En adición a la arquitectura, LASER implementa un tokenizador que utiliza el algoritmo de codificación Byte Pair Encoding (BPE) [48] para generar un vocabulario compartido abarcando 93 idiomas del corpus. Para cada token del vocabulario, se generan embeddings que se ajustan en conjunto durante el entrenamiento de la arquitectura. En el proceso del codificador, los vectores de tokens se utilizan como entrada y se propagan a través de 5 capas de LSTMs bidireccionales, seguido de una operación de reducción mediante Max Pooling para generar un vector representativo de la oración. Por otro lado, el decodificador toma como entrada una concatenación del vector de oración, un vector de token del diccionario y un vector de codificación del idioma. Esta concatenación se somete a una capa LSTM unidireccional que genera un token BPE de salida. La arquitectura inicial de LASER se presenta de manera gráfica en la figura 3.8.

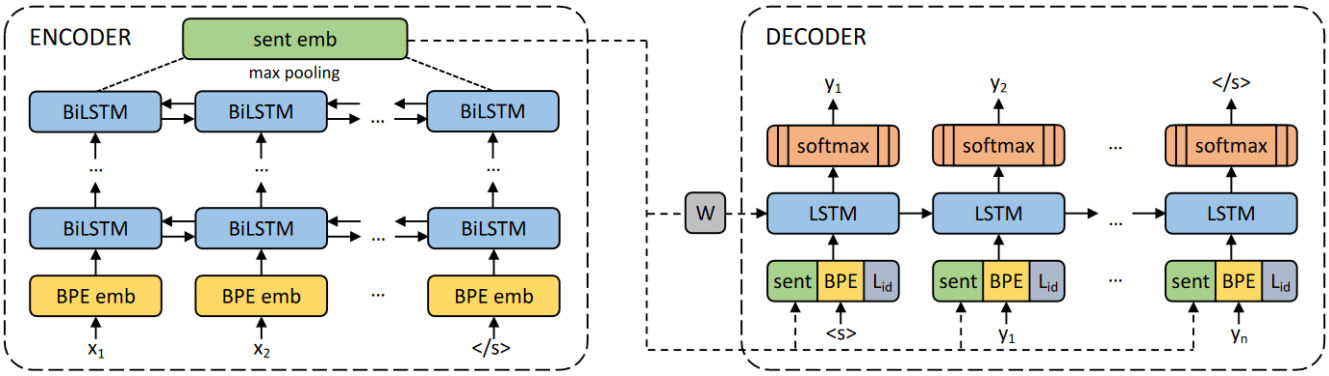


Figura 3.8: Arquitectura de LASER propuesta por [1]

3.6.1 Pre-entrenamiento

Para llevar a cabo el entrenamiento de esta arquitectura, se requiere de texto paralelo en diversos idiomas. El modelo se entrena en su totalidad, incluyendo el codificador y el decodificador, con el propósito de traducir oraciones de un idioma A un idioma B. Como se mencionó previamente, el codificador genera un vector de oración para el idioma A, que actúa como entrada para el decodificador. Dado que se conoce el idioma de la oración B, se obtiene el vector asociado a dicho idioma, que se emplea en cada etapa de predicción.

Con estos dos vectores, se procede a propagarlos hacia el decodificador en un entrenamiento autorregresivo, utilizando el embedding `<s>` como punto de partida. Al concatenar los tres vectores, se propagan por la red recurrente del decodificador para predecir el siguiente token en la oración. Posteriormente, este token se emplea como el siguiente paso dentro de la oración, y así sucesivamente, hasta que la red predice el token `</s>`, marcando el término de la oración, o hasta que se alcanza la longitud máxima esperada para la oración. Luego, se compara la oración generada por el decodificador con la oración real en el idioma B, utilizando la función de pérdida de entropía cruzada categórica:

$$\mathcal{L}_{ce} = -\sum_{i=1}^{V} y_i ln(\hat{y}_i), \tag{3.4}$$

donde y_i e $\hat{y}_i$ corresponden a la etiqueta verdadera y a la predicción que se obtiene del decodificador respectivamente, y V corresponde a la cantidad de palabras que existen en el vocabulario. En este caso, como se está haciendo la comparación con una sola palabra en cada paso de la red recurrente, la función de pérdida se simplifica de la siguiente forma:

$$\mathcal{L}'_{ce} = -ln(\hat{y}_i) = -ln(\frac{e^{z_i}}{\sum_{j=1}^{V} e^{z_j}}). \tag{3.5}$$

Esto ocurre principalmente debido a que el valor asociado a y_i corresponde a una variable binaria, reflejando el intento de predicción del token y_i. Debido a esto, los demás tokens $y_{j \neq i}$ adquieren un valor de cero, dada la naturaleza de la predicción. Por otro lado, $\hat{y}_i$ representa la salida de la función `softmax` aplicada a la palabra i-ésima durante el proceso de predicción.

Con estos fundamentos, se procede al cálculo de la función de pérdida $\mathcal{L}'_{ce}$ para cada paso de la red recurrente del decodificador. Esta función de pérdida es retropropagada a través del decodificador y el codificador, siguiendo un proceso que corresponde a la retropropagación a través del tiempo (BPTT, por sus siglas en inglés). En esta secuencia, los gradientes asociados al error en cada paso de la red recurrente son calculados considerando cada paso como una red independiente, posteriormente sumados y aplicados para ajustar los pesos de toda la red.

El proceso de pre-entrenamiento de LASER tiene como objetivo generar una representación de la oración de entrada que sea efectiva para que el decodificador pueda reconstruirla en otro idioma. Es esencial destacar que, durante la etapa de entrenamiento, una oración puede tener múltiples traducciones paralelas en el corpus. Por ende, se busca que esta representación de la oración sea robusta y aplicable a distintos idiomas que presenten múltiples traducciones para una misma oración. En este contexto, LASER tiene la capacidad de aprovechar la combinación de todos los pares de traducción presentes en el corpus para mejorar su rendimiento y eficacia.

Capítulo 4

Materiales y Métodos

En este capítulo se detallan los materiales, métodos o procedimientos utilizados en la realización del trabajo.

4.1 Conjuntos de datos

Para este trabajo se han utilizado dos conjuntos de datos para poder realizar la validación de la hipótesis. El primero corresponde al conjunto SemEval2019, y el segundo corresponde a un conjunto de datos relacionado al discurso de odio presente en redes sociales con respecto a la Convención Constituyente en Chile el año 2021.

4.1.1 SemEval2019

El primer conjunto de datos es el propuesto por Basile et al. [4] en la conferencia SemEval2019, Tarea 5: "Detección de Discurso de Odio Multilingüe en Contra de Inmigrantes y Mujeres en Twitter". Este conjunto de datos contiene 13.000 mensajes en el idioma inglés, y 6.600 mensajes en castellano. Estos mensajes fueron obtenidos durante el 2018 mediante 3 estrategias:

- Monitoreando cuentas de potenciales víctimas de los tipos de discursos de odio.
- Descargando data de usuarios con historial de mensajes relacionados o catalogados como discurso de odio.
- Recuperando data desde Twitter utilizando la API buscando palabras claves, tanto neutrales como palabras derogatorias o hashtags tóxicos que podrían ser utilizadas en un contexto de odio.

Luego de que la data fue recolectada, se realizó un proceso de etiquetado en dos etapas. La primera utilizó una plataforma de *crowdsourcing* para hacer el etiquetado de los mensajes en tres

niveles binarios: si el mensaje corresponde a discurso de odio o no, identificar si su objetivo es un individuo o un grupo, y si el mensaje presenta una intención agresiva hacia la persona o grupo que se ven afectados. Para los etiquetadores en la plataforma de *crowdsourcing*, se les entregaron guías y definiciones con respecto a los dos tipos de discurso de odio que se quería etiquetar. Una vez terminada la primera etapa, los autores procedieron a realizar un segundo etiquetado utilizando a dos expertos con experiencia previa en el etiquetado de misoginia y xenofobia en inglés y castellano. La etiqueta final para cada mensaje se generó a través de un voto de mayoría entre los 3 etiquetadores (*crowdsourcing*, experto 1 y experto 2). La tabla 4.1 presenta un resumen del número de mensajes para cada idioma, en conjunto con la distribución de los conjuntos de entrenamiento y de pruebas que proveen los autores. Cabe destacar que existe un pequeño desbalance hacia la categoría de "No Discurso de Odio" (no D.O) en ambos idiomas.

Idioma	Entrenamiento	Pruebas	no D.O. / D.O.
Inglés	10000	3000	$\approx 58\%/42\%$
Castellano	5000	1600	$\approx 59\%/41\%$

Tabla 4.1: Número de tweets por cada idioma en el dataset

Para este trabajo de tesis, el conjunto de datos se utilizó realizando múltiples combinaciones de este, con el fin de modelar las tareas de detección de discurso de odio monolingüe, multilingüe y translingüe. Esto se realiza con el fin de poder evaluar la calidad de las representaciones independientes a los idiomas generadas por los modelos mencionados en el capítulo 3, así como para también llevar a cabo la evaluación de estos mismos modelos realizando un ajuste fino utilizando los nuevos conjuntos de datos. Por lo mismo, de este conjunto de datos se generan 5 tareas específicas respetando los conjuntos de entrenamientos y de pruebas propuestos por los autores originalmente:

1. **Monolingüe:** Se utilizan los conjunto de datos originales en dos tareas, monoES (castellano) y monoEN (inglés)

2. **Multilingüe:** Se utilizan ambos conjuntos para generar un corpus de entrenamiento y de pruebas, el cual contiene ambos idiomas.

3. **Translingüe:** se utilizan ambos conjuntos, pero de forma cruzada; ES→EN correspondería al conjunto de entrenamiento en castellano y el conjunto de pruebas en inglés, y EN→ES correspondería a utilizar el conjunto de entrenamiento en inglés y el conjunto de pruebas en castellano.

4.1.2 Ataque y discurso de odio en redes sociales hacia la Convención Constituyente

El conjunto de datos de ataque y discurso de odio en redes sociales hacia la Convención Constituyente, fue un esfuerzo realizado por el grupo Demoscopia Electrónica del Espacio Público, perteneciente a la Pontificia Universidad Católica de Valparaíso (DEEP-PUCV). Este conjunto de datos están enmarcados en la iniciativa del odiometro, los cuales fueron recolectados en el año 2021 mediante el seguimiento de palabras clave y hashtags que hacían mención a la Convención Constituyente. Este corpus contiene 4000 mensajes, en su totalidad en castellano, donde el objetivo principal de la recolección de este conjunto de datos era principalmente obtener ejemplos de tweets que permitieran entrenar un clasificador para poder detectar 3 categorías:

- Ataque a la Convención: Mensajes que critiquen con animosidad a la convencion Constituyente, pero en que no se incite a la violencia
- Discurso de odio: Mensajes que utilicen lenguajes discriminatorios en relación a características de identidad de una persona o un grupo, y que inciten a la discriminación, la hostilidad o la violencia
- Otros mensajes: Mensajes que no estén en las categorías anteriores, siendo principalmente mensajes neutrales o de apoyo a la Convención Constituyente.

Para realizar el etiquetado de este conjunto de datos, se realizó un entrenamiento a 5 etiquetadores para poder identificar mensajes que estuviesen dentro de las categorías anteriormente mencionadas. Como tal, el etiquetado se realizó en una sola etapa, y las etiquetas finales para cada mensaje se obtuvieron mediante un voto de mayoría según la etiqueta que le asigno cada etiquetador. La siguiente tabla muestra un resumen de la cantidad de mensajes por categoría final:

Categoría	Cantidad de mensajes
Ataque a la Convención	1871
Discurso de odio	413
Otros mensajes	1716

Tabla 4.2: Conjunto de datos, ataque y discurso de odio en redes sociales hacia la convencion Constituyente

Una peculiariedad de la distribución de la data es que existe un claro desbalance, donde la cantidad de mensajes que pertenecen a la categoría de discurso de odio es aproximadamente un 10% del conjunto de datos. Adicionalmente, existe la dificultad dentro de la categoría "Ataque a

la Convención", en la cual se pueden presentar instancias de vocabulario que exhiban animosidad hacia la Convención, siendo un punto que se puede prestar para confusiones entre esta categoría y la categoría de discurso de Odio. Finalmente, cabe destacar que el tipo de discurso de odio que se logró recopilar utilizando la metodología antes descrita corresponde a misoginia y xenofobia, lo cual nos permitiría utilizar este conjunto de datos para validar la hipótesis presentada en este trabajo.

Este conjunto de datos se utilizó tanto como benchmark con el fin de poder evaluar el rendimiento de las representaciones y modelos generados, como también como conjunto con el fin de evaluar la transferencia de conocimiento entre conjunto de datos. Con esto en mente, se utiliza el conjunto de datos SemEval 2019 como conjunto de entrenamiento, y el conjunto de la Convención Constituyente como un conjunto de pruebas.

4.2 Modelos

4.2.1 Pre-procesamiento

En este caso se ha realizado un mínimo pre-procesamiento de los textos, debido a diversos factores. Como tal, los tokenizadores pueden verse afectados al momento de enraizar palabras (*stemming*), tal como se muestra en el punto 3.2.1 donde, a modo de ejemplo, una palabra era separada en dos tokens: raíz, seguida por su conjugación. Además de lo anterior, la data utilizada en las etapas de pre-entrenamiento fue utilizada sin pre-procesar. Se trata de mantener la data en la etapa de ajuste fino, tal como se utilizó en el pre-entrenamiento. Adicionalmente, no se han removido palabras vacías, principalmente para evitar romper la secuencialidad de una oración.

Por su parte, las operaciones aplicadas a ambos corpus corresponden a la eliminación de emojis, direcciones web (URLs), y la eliminación del token RT. Esta última operación se lleva a cabo, ya que no aporta información al contenido del mensaje, sino que una indicación de que dicho mensaje es compartido a través de la red social.

Cabe destacar que un pre-procesamiento necesario para llevar a cabo la evaluación de los modelos, en su capacidad como generadores de representaciones, consiste en transformar los tweets desde texto natural, a las representaciones vectoriales. Para este fin, se utilizan las arquitecturas sin ajustar, y utilizando la última capa de la arquitectura de la red neuronal previa a una capa de clasificación, generando un vector numérico, el cual debería ser una representación vectorial independiente del idioma. Un ejemplo de esto para las arquitecturas definidas en el capítulo 3, se presenta de forma gráfica en la figura 4.1 para la oración *The quick brown fox jumps over the lazy dog*. Para esto se realizó la traducción de esta oración a 3 idiomas, para luego obtener la representación vectorial para cada traducción, utilizando los modelos descritos. Luego, utilizando

el algoritmo t-SNE [49], se realizó una proyección en dos dimensiones de dichos vectores para poder graficarlos. Cabe destacar en esta imagen, que existe una proximidad en el espacio vectorial obtenidos de los modelos para las 3 traducciones y la oración original, a excepción de BERT Multilingüe.

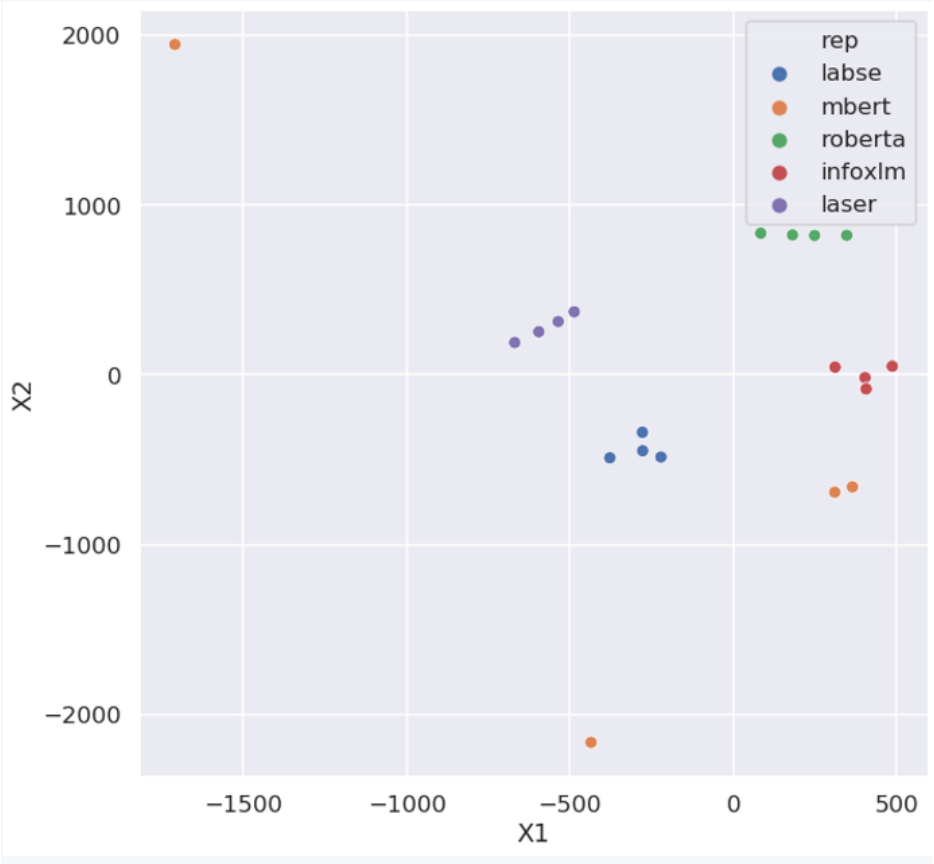


Figura 4.1: Proyección en dos dimensiones utilizando t-SNE [49] para la oración *The quick Brown fox jumps over the lazy dog* traducida en 4 idiomas.

Adicionalmente a la vectorización utilizando los modelos antes descritos, se realizó una vectorización utilizando bolsa de palabras y bolsa de grafemas. Esto se realizó principalmente para poder llevar a cabo una comparación con los modelos base, utilizando una representación tradicional en el área del PLN, en contraste con los métodos propuestos en esta tesis.

4.2.2 Modelos utilizados

Modelos base

Para realizar la validación de la hipótesis, se han implementado 7 modelos bases, de los cuales el primero corresponde a la regresión logística (LR), seguido por 3 modelos de máquinas de vectores

de soporte (SVM) utilizando distintos kernels: SVM linear (LSVM), SVM con un kernel RBF (RSVM), y SVM con kernel polinomial (PSVM). Los otros 3 modelos son basados en árboles de clasificación: el primero siendo el árbol de decisión (DT), seguido por dos modelos ensamblados, Random Forest (RF) y Extremely Randomized Trees (ET). La tabla 4.3, muestra la grilla utilizada para ajustar los hiperparámetros de los modelos base utilizados:

Modelo	Hiperparámetros
LR	C: (0.0001,0.001, 0.01, 0.1, 1, 10, 100), regularización: [”l2”, ”l1”, None]
LSVM	C: (0.0001,0.001, 0.01, 0.1, 1, 10, 100), regularización: [”l2”, ”l1”, None]
RSVM	C: (0.0001,0.001, 0.01, 0.1, 1, 10, 100)
PSVM	C: (0.0001,0.001, 0.01, 0.1, 1, 10, 100), grado polinomio [2, 3, 4 ,5 ,6 ,7]
DT	Profundidad máxima [5, 100] (con paso de 5 en 5)
RF	Profundidad máxima [1, 25], número de estimadores: (50,100,150,200,25)
ET	Profundidad máxima [1, 25], número de estimadores: (50,100,150,200,25)

Tabla 4.3: Hiperparámetros utilizados para los modelos base

Para los modelos basados en transformers, se utilizaron los modelos descritos en el capítulo 3 (LaBSE, mBERT, XLM-RoBERTa, infoXLM) en un régimen de ajuste fino. Los hiperparámetros seleccionados para realizar este ajuste fino corresponden a: utilizar una arquitectura fija, la cual consiste en una capa de dropout con un valor de $p = 0.1$, seguido por una neurona de salida; mantener la tasa de aprendizaje fija con un valor de 0.0001; mantener un tamaño de Batch de 8, y utilizar la entropía cruzada binaria como función de pérdida (4.1).

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}(y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i)) \tag{4.1}$$

4.3 Métricas

Se utilizaron tres métricas para evaluar la calidad de la clasificación en todos los experimentos. Dos de estas métricas se derivan de los resultados de la matriz de confusión: Verdaderos Positivos (TP), Verdaderos Negativos (TN), Falsos Positivos (FP) y Falsos Negativos (FN). La primera métrica fue el Puntaje de Exactitud (A), que proporciona la proporción de observaciones clasificadas correctamente respecto al total de observaciones. El puntaje de exactitud se calculó de la siguiente manera 4.2:

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

La segunda métrica corresponde al puntaje F_1, el cual corresponde a la media armónica entre la Precisión (P) y la exhaustividad (R):

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \tag{4.3}$$

$$F_1 - score = 2 \cdot \frac{P \cdot R}{P + R} \tag{4.4}$$

La tercera métrica corresponde al área bajo la curva ROC (ROC-AUC), la cual es una métrica utilizada para evaluar la capacidad discriminativa de un modelo de clasificación. La curva ROC representa la tasa de verdaderos positivos (exhaustividad) frente a la tasa de falsos positivos (1 - especificidad) para diferentes umbrales de decisión. Un ROC-AUC cercano a 1 indica que el modelo tiene una buena capacidad de discriminar entre las clases positiva y negativa, mientras que un AUC cercano a 0.5 indica un desempeño similar al azar. La métrica AUC se calcula integrando la curva ROC, es decir, calculando el área bajo esta curva. Cuanto mayor sea el AUC, mejor será el rendimiento del modelo en términos de discriminación entre las clases.

Detalles de implementación de los experimentos

Para la realización de los experimentos se definieron 25 semillas aleatorias, las cuales nos permiten separar del conjunto de entrenamiento, un subconjunto de validación. Esto se realiza con el fin de poder ver distintos escenarios para poder realizar el ajuste de hiperparámetros, para los modelos base en ambos conjunto de datos. Por su parte, para los modelos basados en Transformers, estas 25 semillas van a generar distintas inicializaciones de peso para la capa de salida, y afectar a los procesos estocásticos como Dropout y la optimización de los pesos basados en el optimizador Adam. Adicionalmente, dan la capacidad de reproducibilidad de nuestros resultados para ambos modelos base y basados en Transformers en el caso de volver a hacer corridas experimentales.

La implementación de los modelos base se llevó a cabo utilizando la librería scikit-learn [50], mientras que los modelos basados en transformers fueron implementados utilizando PyTorch en conjunto con la biblioteca HuggingFace [51]. La máquina utilizada para correr estos experimentos utiliza un procesador i9, 128 Gb de RAM, y una tarjeta gráfica Nivida RTX 3090.

Capítulo 5

Resultados

En este capítulo se presentan los resultados de las tareas y pruebas realizadas con el fin de contrastar la hipótesis. En primer lugar, se exhiben los resultados del análisis del conjunto de datos de SemEval 2019, el cual ha sido subdividido en cinco tareas llevadas a cabo con dicho conjunto de datos: monolingües (inglés y castellano), multilingüe y translingüe (ES→EN y EN→ES). A continuación, se presentan los resultados relativos al conjunto de datos de la convención constituyente, abordando las tareas monolingües (castellano), multilingüe y translingüe (EN→ES). Por último, se elabora una tabla resumen que resalta el modelo óptimo para cada conjunto y tarea, incluyendo la identificación de posibles diferencias estadísticamente significativas en comparación con el segundo modelo mejor posicionado.

Cabe mencionar que las tablas expuestas en esta sección reflejan valores agregados, como promedios y desviaciones estándar, obtenidos de los 25 experimentos realizados para cada modelo y representación empleada. Para presentar los modelos base, se ha implementado una tabla concisa que muestra los diez mejores resultados ordenados de manera descendente según el puntaje F_1. En el anexo se detallan todos los resultados pertinentes a los modelos base y sus respectivas representaciones, en función de las métricas definidas en la sección 4.3. Adicionalmente, se presentan diagramas de cajas y bigotes presentando las mejores representaciones obtenidas para cada tarea y conjunto de datos según su puntaje F_1.

5.1 Conjunto de datos 1: SemEval

5.1.1 Monolingüe

MonoEN

Para el conjunto de datos SemEval 2019, y la tarea monolingüe en inglés, se presentan los resultados en las tablas 5.1 y 5.2, para los modelos basados en Transformers y modelos base respectivamente.

Para los resultados de los modelos basados en Transformers, tenemos que el mejor rendimiento para las tres métricas a seguir, fue XLM-RoBERTa. Cabe destacar que este modelo es marginalmente superior a LaBSE, y para estas tres métricas no existe evidencia de una diferencia estadísticamente significativa entre ambos modelos.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.5257 ± 0.0181	0.4885 ± 0.0276	0.6423 ± 0.0275
XLM-RoBERTa	0.5289 ± 0.0257	0.4935 ± 0.0386	0.6460 ± 0.0336
infoXLM	0.5079 ± 0.0386	0.4408 ± 0.0498	0.6060 ± 0.0463
mBERT	0.4959 ± 0.0236	0.4405 ± 0.0406	0.6123 ± 0.0242

Tabla 5.1: Resultados para el conjunto de datos de SemEval, para la tarea monolingüe en inglés y utilizando modelos profundos.

Por su parte, el mejor rendimiento en términos de puntaje F_1, corresponde a Random Forest utilizando una bolsa de palabras como representación. Es notable la utilización de LaBSE como representación, que si bien no obtiene el mejor puntaje F_1, cabe destacar que obtiene mejores valores para exactitud y área bajo la curva ROC. Adicionalmente, de los 10 mejores valores, esta representación se encuentra en 7 de los 10 registros presentes en la tabla 5.2.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
RF	BOW	0.5662 ± 0.0156	0.5641 ± 0.0156	0.5911 ± 0.0157
LR	LaBSE	0.5680 ± 0.0188	0.5615 ± 0.0228	0.6727 ± 0.0113
DT	LaBSE	0.5579 ± 0.0088	0.5577 ± 0.0089	0.6049 ± 0.0113
PSVM	LaBSE	0.5656 ± 0.0088	0.5532 ± 0.0088	0.6775 ± 0.0096
LSVM	LaBSE	0.5618 ± 0.0169	0.5526 ± 0.0207	0.6716 ± 0.0104
DT	infoXLM	0.5519 ± 0.0094	0.5518 ± 0.0094	0.5584 ± 0.0112
RSVM	LaBSE	0.5551 ± 0.0089	0.5391 ± 0.0088	0.6851 ± 0.0105
ET	LaBSE	0.5389 ± 0.0093	0.5325 ± 0.0089	0.6121 ± 0.0101
RF	LaBSE	0.5403 ± 0.0091	0.5318 ± 0.0090	0.6219 ± 0.0101
ET	BOW	0.5722 ± 0.0175	0.5264 ± 0.0201	0.5978 ± 0.0211

Tabla 5.2: Resultados para el conjunto de datos de SemEval, para la tarea monolingüe en inglés y utilizando modelos base.

MonoES

Para la tarea monolingüe en castellano, utilizando el conjunto de datos de SemEval, podemos observar que LaBSE obtiene los mejores resultados en las 3 métricas. Adicionalmente, en comparación con el segundo mejor resultado (mBERT), se tiene una diferencia estadísticamente significativa con $p < 0.0001$.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.7610 ± 0.0127	0.7583 ± 0.0113	0.8504 ± 0.0095
XLM-RoBERTa	0.7356 ± 0.0414	0.7210 ± 0.0815	0.8045 ± 0.0775
infoXLM	0.6827 ± 0.0814	0.6095 ± 0.1725	0.7303 ± 0.1333
mBERT	0.7444 ± 0.0146	0.7420 ± 0.0135	0.8299 ± 0.0114

Tabla 5.3: Resultados para el conjunto de datos de SemEval, para la tarea monolingüe en castellano y utilizando modelos profundos.

Por otra parte, cuando se analizan los resultados para los modelos bases y las distintas representaciones utilizadas, se puede observar que LaBSE obtiene el mejor resultado para las tres métricas. Asimismo, al comparar el mejor resultado, el cual corresponde a una SVM con kernel rbf y LaBSE como representación de textos, versus SVM con kernel RBF y bolsa de palabras, se tiene una diferencia estadísticamente significativa con $p < 0.0001$.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
RSVM	LaBSE	0.7339 ± 0.0084	0.7250 ± 0.0090	0.7994 ± 0.0091
PSVM	LaBSE	0.7329 ± 0.0099	0.7223 ± 0.0107	0.7960 ± 0.0094
RSVM	BOW	0.7107 ± 0.0094	0.7045 ± 0.0097	0.7853 ± 0.0098
PSVM	LASER	0.7109 ± 0.0130	0.7037 ± 0.0133	0.7649 ± 0.0135
LSVM	BOW	0.7053 ± 0.0111	0.7017 ± 0.0110	0.7618 ± 0.0100
PSVM	BOW	0.7054 ± 0.0087	0.6974 ± 0.0094	0.7478 ± 0.0104
RSVM	LASER	0.7038 ± 0.0111	0.6966 ± 0.0116	0.7796 ± 0.0108
LR	BOW	0.7019 ± 0.0143	0.6965 ± 0.0136	0.7653 ± 0.0103
RSVM	infoXLM	0.6925 ± 0.0105	0.6840 ± 0.0108	0.7442 ± 0.0118
LR	LaBSE	0.6894 ± 0.0147	0.6833 ± 0.0142	0.7528 ± 0.0143

Tabla 5.4: Resultados para el conjunto de datos de SemEval, para la tarea monolingüe en castellano y utilizando modelos base

5.1.2 Multilingüe

Cuando se realiza la comparativa de los resultados para la tarea multilingüe, se presenta que para los modelos profundos el mejor resultado se obtiene por el modelo LaBSE, seguido por mBERT. Adicionalmente, los resultados obtenidos por LaBSE tienen una diferencia estadísticamente significativa con valor $p < 0.0001$.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.6097 ± 0.0145	0.6002 ± 0.0183	0.7172 ± 0.0183
XLM-RoBERTa	0.5981 ± 0.0329	0.5634 ± 0.0832	0.6764 ± 0.0762
infoXLM	0.5861 ± 0.0234	0.5167 ± 0.0986	0.6287 ± 0.0917
mBERT	0.5807 ± 0.0155	0.5671 ± 0.0196	0.6739 ± 0.0204

Tabla 5.5: Resultados para el conjunto de datos de SemEval, para la tarea multilingüe y utilizando modelos profundos.

Para los modelos base, LaBSE como representación en conjunto con una SVM polinomial obtienen los mejores resultados en puntaje F_1 y exactitud. Se puede destacar que este resultado sería mejor, inclusive en comparación con los modelos profundos, encontrando diferencias estadísticamente significativas con $p < 0.0001$. Adicionalmente, cabe destacar que LaBSE contiene la mayor cantidad de modelos, faltando solo la regresión logística dentro del top 10 de mejores resultados.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
PSVM	LaBSE	0.6322 ± 0.0063	0.6316 ± 0.0063	0.7116 ± 0.0074
RSVM	LaBSE	0.6322 ± 0.0073	0.6315 ± 0.0074	0.7149 ± 0.0064
LSVM	LaBSE	0.6188 ± 0.0121	0.6180 ± 0.0127	0.6942 ± 0.0065
RF	LaBSE	0.6191 ± 0.0085	0.6166 ± 0.0086	0.6630 ± 0.0074
LR	LaBSE	0.6171 ± 0.0124	0.6159 ± 0.0127	0.6940 ± 0.0061
ET	LaBSE	0.6164 ± 0.0072	0.6120 ± 0.0074	0.6575 ± 0.0065
RSVM	XLM-RoBERTa	0.5947 ± 0.0053	0.5946 ± 0.0053	0.6506 ± 0.0070
PSVM	XLM-RoBERTa	0.5941 ± 0.0063	0.5939 ± 0.0063	0.6609 ± 0.0078
LSVM	infoXLM	0.5945 ± 0.0205	0.5921 ± 0.0236	0.6713 ± 0.0095
LR	infoXLM	0.5938 ± 0.0180	0.5920 ± 0.0197	0.6651 ± 0.0075

Tabla 5.6: Resultados para el conjunto de datos de SemEval, para la tarea multilingüe y utilizando modelos base.

5.1.3 Translingüe

EN→ES (Conjunto de entrenamiento en inglés, y conjunto de pruebas en castellano)

En el caso de la primera tarea translingüe para el conjunto de datos de SemEval, tenemos diferencias entre los mejores resultados obtenidos para los modelos basados en Transformers, y los modelos base utilizando las representaciones independientes para los idiomas. En la tabla 5.7 podemos observar cómo LaBSE como modelo de aprendizaje obtiene un puntaje F_1 de 0.6732 en promedio. Es importante señalar que este puntaje es mayor al obtenido en la misma competencia de SemEval [4], donde se obtuvo un puntaje F_1 máximo de 0.6510. Esto es digno de mención, debido a que se obtuvo un mejor resultado utilizando el conjunto de datos en castellano, mientras que en la competencia solo se utilizó el conjunto en una tarea monolingüe.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.6908 ± 0.0108	0.6732 ± 0.0152	0.7584 ± 0.0108
XLM-RoBERTa	0.6740 ± 0.0166	0.6396 ± 0.0346	0.7399 ± 0.0168
infoXLM	0.6518 ± 0.0416	0.6045 ± 0.0912	0.6932 ± 0.0767
mBERT	0.6526 ± 0.0150	0.6066 ± 0.0383	0.7026 ± 0.0186

Tabla 5.7: Resultados para el conjunto de datos de SemEval, para la tarea translingüe EN→ES y utilizando modelos profundos.

Por su parte, para los modelos base, hay que destacar dos puntos: por una parte, los primeros 5 puntajes F_1 obtuvieron un mejor rendimiento que la competencia de SemEval, al igual que cuando se realizó la comparación con los modelos basados en Transformers. El segundo punto corresponde a que el mejor modelo obtenido para los modelos base, corresponde a una SVM con kernel RBF, y utilizando la representación LASER. Cabe destacar que, si se comparan los resultados de LaBSE como modelo y RSVM + LASER, se deriva que no hay evidencia de que la diferencia sea estadísticamente significativa entre ambos.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
RSVM	LASER	0.6763 ± 0.0103	0.6685 ± 0.0100	0.7401 ± 0.0107
PSVM	LASER	0.6676 ± 0.0111	0.6584 ± 0.0109	0.7286 ± 0.0109
LSVM	LASER	0.6641 ± 0.0146	0.6542 ± 0.0117	0.7322 ± 0.0108
PSVM	LaBSE	0.6630 ± 0.0104	0.6505 ± 0.0110	0.6973 ± 0.0126
LR	LASER	0.6591 ± 0.0120	0.6503 ± 0.0117	0.7235 ± 0.0102
RSVM	LaBSE	0.6477 ± 0.0107	0.6389 ± 0.0111	0.7029 ± 0.0132
RF	LASER	0.6405 ± 0.0114	0.6243 ± 0.0112	0.6699 ± 0.0109
ET	LASER	0.6361 ± 0.0113	0.6204 ± 0.0119	0.6659 ± 0.0106
LR	LaBSE	0.6231 ± 0.0170	0.6185 ± 0.0153	0.6724 ± 0.0136
RF	LaBSE	0.6338 ± 0.0115	0.6176 ± 0.0122	0.6686 ± 0.0127

Tabla 5.8: Resultados para el conjunto de datos de SemEval, para la tarea translingüe EN→ES y utilizando modelos base.

ES→EN (Conjunto de entrenamiento en castellano, y conjunto de pruebas en inglés)

Para esta tarea en el conjunto de datos de SemEval, se obtiene que el mejor modelo basado en Transformers corresponde a LaBSE para todas las métricas por un amplio margen. Por otra parte, cabe destacar que el rendimiento para la tarea translingüe es menor en comparación con la tarea de clasificación monolingüe en castellano.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.6637 ± 0.0134	0.6441 ± 0.0201	0.7214 ± 0.0141
XLM-RoBERTa	0.6346 ± 0.0268	0.5508 ± 0.0798	0.6765 ± 0.0625
infoXLM	0.6059 ± 0.0406	0.5258 ± 0.1051	0.6274 ± 0.0853
mBERT	0.6304 ± 0.0178	0.5385 ± 0.0528	0.6650 ± 0.0225

Tabla 5.9: Resultados para el conjunto de datos de SemEval, para la tarea translingüe ES→EN y utilizando modelos profundo.

En el caso de los modelos base, si bien el mejor modelo utiliza una representación de XLM-RoBERTa, tenemos en segundo lugar a LaBSE como representación. En este caso, si se realiza una comparación entre los resultados obtenidos para el mejor modelo basado en Transformers y los modelos base, no existe evidencia de una diferencia estadísticamente significativa entre los resultados obtenidos. Adicionalmente, si bien existe capacidad de ajuste de los modelos base utilizando las representaciones independientes al idioma, el efecto del conjunto de datos de entrenamiento, el cual no es óptimo para la tarea en castellano, se ve reflejado de igual manera para estos modelos.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
RSVM	XLM-RoBERTa	0.6474 ± 0.0099	0.6390 ± 0.0098	0.6902 ± 0.0110
RSVM	LaBSE	0.6420 ± 0.0083	0.6236 ± 0.0088	0.6858 ± 0.0087
PSVM	LaBSE	0.6449 ± 0.0082	0.6202 ± 0.0094	0.6798 ± 0.0095
PSVM	XLM-RoBERTa	0.6332 ± 0.0089	0.6199 ± 0.0089	0.6829 ± 0.0110
LR	XLM-RoBERTa	0.6208 ± 0.0236	0.6125 ± 0.0241	0.6886 ± 0.0132
LSVM	LaBSE	0.6232 ± 0.0096	0.5951 ± 0.0188	0.6620 ± 0.0079
LR	LaBSE	0.6219 ± 0.0066	0.5916 ± 0.0107	0.6610 ± 0.0067
LSVM	XLM-RoBERTa	0.6155 ± 0.0375	0.5891 ± 0.0429	0.6878 ± 0.0106
PSVM	LASER	0.6089 ± 0.0094	0.5881 ± 0.0104	0.6269 ± 0.0110
RF	XLM-RoBERTa	0.6057 ± 0.0127	0.5855 ± 0.0127	0.6285 ± 0.0155

Tabla 5.10: Resultados para el conjunto de datos de SemEval, para la tarea translingüe ES→EN y utilizando modelos base.

5.2 Conjunto de datos 2: Convención Constituyente

5.2.1 Monolingüe

Para el conjunto de datos de la Convención Constituyente en la tarea monolingüe en castellano, se tiene que el mejor resultado se da con el modelo LaBSE para todas las métricas. En este caso, vale la pena destacar que realizar un ajuste fino utilizando el conjunto de datos SemEval, y luego llevar a cabo pruebas en el conjunto de datos de la Convención Constituyente, otorga peores resultados que utilizando modelos, bases y representaciones independientes del idioma.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.8316 ± 0.0261	0.5816 ± 0.0188	0.6830 ± 0.0254
XLM-RoBERTa	0.8134 ± 0.0489	0.5572 ± 0.0367	0.6450 ± 0.0699
infoXLM	0.8423 ± 0.0539	0.5412 ± 0.0515	0.6374 ± 0.0790
mBERT	0.8180 ± 0.0360	0.5518 ± 0.0216	0.6248 ± 0.0282

Tabla 5.11: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea monolingüe en castellano y utilizando modelos basados en Transformers.

Por su parte, los resultados de los modelos base y representaciones para el conjunto de datos de la Convención Constituyente, son de mejor calidad que los de los modelos descritos en la tabla 5.11. Esto se puede evidenciar comparando ambos puntajes F_1 para el modelo de LaBSE con respecto al resto de las métricas. Si bien este modelo tiene un buen puntaje de exactitud, su bajo puntaje F_1 indica que el modelo tiene un sesgo hacia la categoría mayoritaria. Por otra parte, el modelo de SVM con kernel RBF y utilizando LaBSE como representación muestra mejores resultados, teniendo un balance a través de las distintas métricas. Esto se puede evidenciar en la tabla 5.12, donde se exhibe un comportamiento similar para la mayoría de los modelos dentro del top 10. Finalmente, es importante mencionar que la diferencia de los mejores modelos base es estadísticamente significativa entre estos y los modelos basados en Transformers. No obstante, para los dos primeros puntajes presentes en 5.12 no existe evidencia de una diferencia estadísticamente significativa.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
RSVM	LaBSE	0.7339 ± 0.0084	0.7250 ± 0.0090	0.7994 ± 0.0091
PSVM	LaBSE	0.7329 ± 0.0099	0.7223 ± 0.0107	0.7960 ± 0.0094
RSVM	BOW	0.7107 ± 0.0094	0.7045 ± 0.0097	0.7853 ± 0.0098
PSVM	LASER	0.7109 ± 0.0130	0.7037 ± 0.0133	0.7649 ± 0.0135
LSVM	BOW	0.7053 ± 0.0111	0.7017 ± 0.0110	0.7618 ± 0.0100
PSVM	BOW	0.7054 ± 0.0087	0.6974 ± 0.0094	0.7478 ± 0.0104
RSVM	LASER	0.7038 ± 0.0111	0.6966 ± 0.0116	0.7796 ± 0.0108
LR	BOW	0.7019 ± 0.0143	0.6965 ± 0.0136	0.7653 ± 0.0103
RSVM	infoXLM	0.6925 ± 0.0105	0.6840 ± 0.0108	0.7442 ± 0.0118
LR	LaBSE	0.6894 ± 0.0147	0.6833 ± 0.0142	0.7528 ± 0.0143

Tabla 5.12: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea monolingüe en castellano y utilizando modelos base.

5.2.2 Multilingüe

Para el caso multilingüe con el conjunto de datos de la convención constituyente y los modelos basados en Transformers, LaBSE es el que tiene mejor rendimiento. De igual manera, uno puede observar que estos modelos sufren de un sesgo hacia la clase mayoritaria debido a su exactitud

alta, y bajo puntaje F_1, presentando un comportamiento similar a la tarea monolingüe en la tabla 5.11.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.7999 ± 0.0372	0.5767 ± 0.0183	0.6878 ± 0.0216
XLM-RoBERTa	0.7634 ± 0.1250	0.5336 ± 0.0701	0.6369 ± 0.0689
infoXLM	0.7552 ± 0.1431	0.5261 ± 0.0826	0.6369 ± 0.0782
mBERT	0.7897 ± 0.0420	0.5611 ± 0.0218	0.6648 ± 0.0254

Tabla 5.13: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea multilingüe y utilizando modelos basados en Transformers.

Así mismo, en el caso de los modelos base para la tarea multilingüe. La tabla 5.14 presenta los resultados de las corridas experimentales. Podemos observar que existe una degradación del rendimiento de los modelos, en comparación con la tabla 5.12 de la tarea monolingüe para los modelos base, presentando el mismo comportamiento que los modelos basados en transformer para esta misma tarea. Como tal, esto podría indicar que la incorporación de texto en inglés en el conjunto de entrenamiento puede estar perjudicando en el proceso de aprendizaje para esta tarea en castellano.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
PSVM	XLM-RoBERTa	0.8143 ± 0.0106	0.5797 ± 0.0190	0.6576 ± 0.0278
RSVM	XLM-RoBERTa	0.8174 ± 0.0094	0.5755 ± 0.0181	0.6360 ± 0.0295
RSVM	infoXLM	0.8004 ± 0.0118	0.5609 ± 0.0146	0.6486 ± 0.0275
PSVM	infoXLM	0.8057 ± 0.0093	0.5571 ± 0.0148	0.6483 ± 0.0216
LSVM	infoXLM	0.8161 ± 0.0518	0.5509 ± 0.0209	0.6590 ± 0.0211
PSVM	LaBSE	0.8142 ± 0.0112	0.5505 ± 0.0162	0.6210 ± 0.0239
LSVM	XLM-RoBERTa	0.7935 ± 0.0717	0.5466 ± 0.0256	0.6223 ± 0.0314
RSVM	LaBSE	0.8098 ± 0.0113	0.5417 ± 0.0161	0.6103 ± 0.0303
PSVM	LASER	0.7670 ± 0.0098	0.5386 ± 0.0164	0.6337 ± 0.0194
LR	infoXLM	0.7663 ± 0.0502	0.5382 ± 0.0234	0.6413 ± 0.0229

Tabla 5.14: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea multilingüe y utilizando modelos base.

5.2.3 Translingüe

Para la tarea translingüe en el conjunto de la convención constituyente, podemos ver una degradación bastante evidente. Esto se debe a que el conjunto de datos de SemEval 2019 en inglés es muy disímil en relación al conjunto de datos de la Convención Consituyente. En este caso, los resultados obtenidos para los modelos basados en Transformers son los peores en comparación con todas las tareas anteriores, alcanzando puntajes F_1 por debajo de los 0.50. Si bien, se pueden utilizar

modelos con representaciones independientes del idioma, también es necesario tener conjuntos de datos que sean de calidad con el fin de poder realizar inferencias independientes del idioma.

Modelo	Exactitud	Puntaje F_1	ROC-AUC
LaBSE	0.7435 ± 0.0518	0.5293 ± 0.0222	0.6270 ± 0.0252
XLM-RoBERTa	0.7412 ± 0.0640	0.5260 ± 0.0271	0.6176 ± 0.0210
infoXLM	0.7037 ± 0.1983	0.4836 ± 0.1010	0.5990 ± 0.0527
mBERT	0.7926 ± 0.0362	0.5357 ± 0.0155	0.6367 ± 0.0232

Tabla 5.15: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea translingüe EN→ES y utilizando modelos basados en Transformers.

Por su parte, para los modelos base podemos presenciar la misma degradación del rendimiento de los modelos. En este caso, viendo las métricas para la mayoría de los modelos dentro de los mejores 10 resultados, se puede ver que se presenta el sesgo hacia la clase mayoritaria (no odio), debido a la alta exactitud y bajo puntaje F_1.

Modelo	Representación	Exactitud	Puntaje F_1	ROC-AUC
PSVM	infoXLM	0.8370 ± 0.0095	0.5503 ± 0.0159	0.5799 ± 0.0240
RSVM	infoXLM	0.8273 ± 0.0085	0.5426 ± 0.0182	0.5752 ± 0.0239
PSVM	XLM-RoBERTa	0.7800 ± 0.0110	0.5406 ± 0.0227	0.6099 ± 0.0291
RSVM	XLM-RoBERTa	0.7975 ± 0.0091	0.5354 ± 0.0213	0.6001 ± 0.0312
LSVM	infoXLM	0.8208 ± 0.0559	0.5284 ± 0.0176	0.5976 ± 0.0236
PSVM	mBERT	0.8229 ± 0.0096	0.5267 ± 0.0125	0.5785 ± 0.0241
RSVM	mBERT	0.8023 ± 0.0096	0.5196 ± 0.0143	0.5735 ± 0.0229
LR	infoXLM	0.8476 ± 0.0383	0.5195 ± 0.0221	0.5900 ± 0.0273
ET	BOC	0.7899 ± 0.0129	0.5171 ± 0.0098	0.5349 ± 0.0156
ET	XLM-RoBERTa	0.7974 ± 0.0159	0.5162 ± 0.0179	0.5484 ± 0.0320

Tabla 5.16: Resultados para el conjunto de datos de la Convención Constituyente, para la tarea translingüe EN→ES y utilizando modelos base.

5.3 Resumen de mejores resultados

En la tabla 5.17 se presenta un resumen de los resultados para cada tarea, donde se muestran cuál fue el mejor modelo y representación obtenido según su puntaje F_1. Adicionalmente, se muestra si el resultado obtenido tiene una significancia estadística, con respecto al segundo mejor modelo/representación distinta al mejor resultado. Se puede observar que LaBSE obtiene el mejor rendimiento para 5 de las 8 tareas realizadas en este trabajo. Sin embargo, solo en 3 de los 5 resultados para LaBSE se obtienen significancia estadística en relación a el segundo mejor modelo/representación distinto de LaBSE.

Con respecto a las tareas definidas en la hipótesis de este trabajo, tenemos que LaBSE obtiene los mejores resultados, ya sea como modelo de clasificación, como también representación para el conjunto de datos SemEval. No obstante, para las tareas translingües se puede observar que no hay diferencia estadísticamente significativa con LASER y XLM-RoBERTa para las tareas ES→EN y EN→ES respectivamente. Adicionalmente, se puede observar que para el segundo conjunto de datos en las tareas multilingüe y translingüe, podemos observar que LaBSE no es capaz de obtener los mejores resultados en términos de puntaje F_1.

Para el resto de resultados obtenidos podemos observar que predominantemente los modelos basados en SVMs en conjunto con las representaciones independientes al idioma, obtienen consistentemente resultados superiores al resto de los modelos bases para tareas multilingües y translingües. Por otra parte, la representación de bolsa de palabras aparece dentro de los mejores 10 puntajes F_1 para las corridas experimentales en las tareas monolingües. Finalmente, la bolsa de grafemas (BOC) solo aparece dentro de los mejores 10 resultados para la tarea translingüe del conjunto de datos de la Convención Constituyente.

Conjunto de datos	Tarea	Modelo	Representación	Significancia Estadistica	valores t y p
SemEval 2019	monoEN	Random Forest	Bag of Words	No	$t(48) = 0.4706, p = 0.6405$
SemEval 2019	monoES	LaBSE	-	Si	$t(48) = 4.6293, p = 2.8 \times 10^{-5}$
SemEval 2019	multilingüe	PSVM	LaBSE	Si	$t(48) = 22.4709, p < 0.00001$
SemEval 2019	ES→EN	LaBSE	-	No	$t(48) = 1.2916, p = 0.2027$
SemEval 2019	EN→ES	LaBSE	-	No	$t(48) = 1.1403, p = 0.2598$
Convención Constituyente	monoES	RSVM	LaBSE	Si	$t(48) = 7.7463, p < 0.00001$
Convención Constituyente	multilingüe	PSVM	XLM-RoBERTa	No	$t(48) = 0.5686, p = 0.5723$
Convención Constituyente	EN→ES	PSVM	infoXLM	No	$t(48) = 1.7500p = 0.0865$

Tabla 5.17: Mejores resultados para cada conjunto de datos y tarea presentada en este trabajo, donde se detalla que modelo, representación y si existe una significancia estadística en la comparación entre el siguiente mejor resultado para otra representación.

Capítulo 6

Conclusiones y Trabajo futuro

6.1 Conclusiones

En este trabajo se han estudiado las potencialidades de modelos basados en la arquitectura Transformers, los cuales han sido entrenados para generar representaciones independientes al idioma para tareas de clasificación monolingüe, multilingües y translingües. En concreto, se han realizado experimentos con los modelos descritos para la detección de dos tipos de discurso de odio: xenofobia y misoginia. Para esto, se han utilizado los modelos basados en Transformers como extractor de características generando vectores de oración, los cuales son posteriormente utilizados como datos de entrada para modelos base y contrastados con dos representaciones tradicionales en el mundo del procesamiento de lenguaje natural. Adicionalmente, se utilizaron los modelos basados en Transformers como modelos de aprendizaje, añadiendo una capa de salida y realizando el proceso de ajuste fino para los conjuntos de datos propuestos. Con el fin de poder medir las capacidades de estos modelos en las tareas anteriormente descritas, los conjuntos de datos fueron utilizados para generar tareas monolingües, multilingües y translingües, generando un total de 8 tareas en las cuales se evaluaron los modelos para realizar clasificación de misoginia y xenofobia sobre mensajes de la red social Twitter en castellano e inglés.

Los resultados obtenidos muestran la potencialidad no solo de LaBSE como modelo de clasificación y generador de representaciones independientes del idioma, sino que también se puede observar el potencial de otros modelos tales como XLM-RoBERTa, LASER e infoXLM para las distintas tareas evaluadas en este trabajo. Si bien LaBSE obtiene un buen rendimiento en las diversas tareas, obteniendo el mejor puntaje F_1 en cinco de ocho tareas en total, solo en tres de estas existe una diferencia estadísticamente significativa con respecto a otro modelo/representación, y de las cuales solo una corresponde a la tarea multilingüe. Con respecto a las otras dos tareas, las cuales LaBSE sí tiene una diferencia estadísticamente significativa, estas corresponden a las tareas monolingües en castellano para ambos conjuntos de datos. Si bien LaBSE puede generar mejores

representaciones que el resto de los otros modelos basados en Transformers, probablemente una implementación basada en BETO [38] podría obtener mejores resultados para esos conjuntos de datos. Finalmente, para las dos tareas translingües en la cual si bien LaBSE obtuvo el mejor desempeño, pero sin una diferencia estadísticamente significativa, se obtiene que los siguientes mejores modelos corresponden a LASER y XLM-RoBERTa para las tareas translingües ES→EN y EN→ES respectivamente.

Con respecto al resto de los experimentos en las otras tareas, se obtiene que para el primer conjunto de datos, el modelo con mayor puntaje F_1 corresponde a Random Forest utilizando una bolsa de palabras. Esto en sí tiene sentido, considerando que un modelo especializado para un idioma específico puede tener mejores resultados en una tarea monolingüe. No obstante, si vemos el segundo mejor resultado, éste corresponde a una regresión logística utilizando LaBSE como representación. Más aún, viendo que no existe una diferencia estadísticamente significativa entre estos dos modelos, y analizando la métrica de área bajo la curva ROC, podemos inferir que la regresión logística y LaBSE está obteniendo un mejor ajuste que Random Forest. No obstante, cabe destacar que en términos de la complejidad de ambos modelos, Random Forest es bastante más simple que la arquitectura de LaBSE. Sin embargo, para el segundo mejor resultado, LaBSE se utilizó como generador de representaciones, siendo éste un proceso que se puede realizar previamente al entrenamiento y clasificación, por lo que la comparación de la complejidad debería ser con respecto a la regresión logística, la cual es un modelo menos complejo que Random Forest.

Finalmente, con respecto al segundo conjunto de datos de la Convención Constituyente, las pruebas realizadas obtuvieron resultados positivos para LaBSE, y los otros modelos basados en Transformers. No obstante, analizando los resultados surgen preocupaciones con respecto a cómo se plantearon los experimentos realizados sobre este conjunto de datos. Específicamente, si uno observa los resultados obtenidos para las tareas multilingües y translingües, existe una degradación de los resultados con respecto a la tarea monolingüe, al incluir data en inglés del conjunto de datos de SemEval. Como tal, es esperable esta degradación principalmente debido a que si bien ambos consisten en conjuntos de datos enfocados a la misoginia y a la xenofobia, las diferencias de palabras coloquiales con respecto al inglés en el año 2019 es distinto al castellano empleado en Chile el año 2021. Más aún, el desbalance presente en el conjunto de datos puede estar sesgando los modelos a dicha clase mayoritaria.

6.2 Trabajo Futuro

Una propuesta para trabajo futuro sería una evaluación más exhaustiva, incluyendo no solo más idiomas a evaluar, sino que también otros tipos de discurso de odio. Ante esto mismo, una evaluación incluyendo un detalle del rendimiento con respecto al tipo de discurso de odio podría

ser útil para entender las dificultades que existen para generar un clasificador general de discurso de odio, en contraste con generar clasificadores que sean especializados para los distintos tipos de discurso de odio. Además de esto, poder explorar técnicas de aumentación de datos en texto con discurso de odio es de interés, debido a la experiencia con el desbalance en el conjunto de datos de la Convención Constituyente.

Con respecto a las arquitecturas utilizadas en este trabajo, un enfoque a explorar a futuro puede ser un refinamiento de los hiperparámetros de las redes basadas en Transformers. Ya sea desde investigar con respecto a variaciones de los hiperparámetros detallados en la sección 4.2.2, como también explorar el rendimiento de los vectores generados por un modelo, el cual haya recibido el tratamiento de ajuste fino en tareas de detección del discurso de odio. Por otra parte, poder realizar un ensamblado entre modelos y representaciones podría ser beneficioso, explorando cuáles pueden ser las mejores combinaciones de representaciones obtenidas de los modelos basados en Transformers. Por lo demás, para los modelos base se puede explorar la forma de obtener vocabularios que combinen grafemas, aplicando bigramas o trigramas de grafemas para tener una representación más robusta para codificar un vocabulario multilingüe. Del mismo modo, se podría explorar la utilización de Byte Pair Encoding para generar el vocabulario necesario. Adicionalmente a esto, el estado del arte de los modelos de lenguaje basados en transformers es un mundo altamente competitivo, por lo que nuevos modelos de lenguaje para tareas multilingüe y translingües pueden surgir mejorando el rendimiento dentro de la tarea de la clasificación del discurso de odio.

Finalmente, con el fin de asegurar la reproducibilidad de los experimentos presentados en este trabajo, los códigos necesarios se comparten en el siguiente enlace [1]. Asimismo, con respecto a la disponibilidad de datos, para el conjunto de SemEval se puede descargar de múltiples sitios, no obstante los autores disponen de un repositorio [2] para poder realizar la descarga del conjunto de datos. Adicionalmente, dentro del repositorio del código asociado a esta tesis se adjunta el conjunto de datos SemEval pre-procesado para poder realizar la reproducción de estos experimentos. Con respecto al conjunto de datos de la Convención Constituyente, para obtener este conjunto se necesita hacer una petición formal al director del proyecto DEEP-PUCV Pedro Santander (pedro.santander@pucv.cl). Con la aprobación del director, se pueden compartir los datos utilizados en este proyecto para realizar la reproducción de los experimentos asociados al segundo conjunto de datos. No obstante, los códigos utilizados para la experimentación del segundo conjunto de datos se harán disponibles en el repositorio de los códigos anteriormente mencionado.

[1] https://github.com/capkuro/DetectingHatespeechLabse
[2] https://github.com/cicl2018/HateEvalTeam/tree/master/Data%20Files

Bibliography

[1] M. Artetxe and H. Schwenk, "Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 597–610, 2019. DOI: 10.1162/tacl_a_00288.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[3] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT sentence embedding," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 878–891. DOI: 10.18653/v1/2022.acl-long.62.

[4] V. Basile, C. Bosco, E. Fersini, *et al.*, "SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, J. May, E. Shutova, A. Herbelot, X. Zhu, M. Apidianaki, and S. M. Mohammad, Eds., Minneapolis, Minnesota, USA: Association for Computational Linguistics, Jun. 2019, pp. 54–63. DOI: 10.18653/v1/S19-2007.

[5] M. S. Jahan and M. Oussalah, "A systematic review of hate speech automatic detection using natural language processing.," *Neurocomputing*, p. 126 232, 2023. DOI: 10.1016/j.neucom.2023.126232.

[6] G. Glavaš, M. Karan, and I. Vulić, "XHate-999: Analyzing and detecting abusive language across domains and languages," in *Proceedings of the 28th International Conference on Computational Linguistics*, D. Scott, N. Bel, and C. Zong, Eds., Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 6350–6365. DOI: 10.18653/v1/2020.coling-main.559.

[7] N. Ousidhoum, Z. Lin, H. Zhang, Y. Song, and D.-Y. Yeung, "Multilingual and multi-aspect hate speech analysis," K. Inui, J. Jiang, V. Ng, and X. Wan, Eds., pp. 4675–4684, Nov. 2019. DOI: 10.18653/v1/D19-1474.

[8] M. Zampieri, P. Nakov, S. Rosenthal, *et al.*, "SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020)," A. Herbelot, X. Zhu, A. Palmer, N. Schneider, J. May, and E. Shutova, Eds., pp. 1425–1447, Dec. 2020. DOI: 10.18653/v1/2020.semeval-1.188.

[9] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., vol. 30, Curran Associates, Inc., 2017. DOI: 10.5555/3295222.3295349.

[10] Y. Liu, M. Ott, N. Goyal, *et al.*, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019. DOI: 10.48550/arXiv.1907.11692. arXiv: 1907.11692.

[11] Z. Chi, L. Dong, F. Wei, *et al.*, "InfoXLM: An information-theoretic framework for cross-lingual language model pre-training," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, Jun. 2021, pp. 3576–3588. DOI: 10.18653/v1/2021.naacl-main.280. [Online]. Available: https://aclanthology.org/2021.naacl-main.280.

[12] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002. DOI: 10.1145/505282.505283.

[13] K. K, Z. Wang, S. Mayhew, and D. Roth, "Cross-lingual ability of multilingual bert: An empirical study," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=HJeT3yrtDr.

[14] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–30, 2018. DOI: 10.1145/3232676.

[15] Z. Zhang, D. Robinson, and J. Tepper, "Detecting hate speech on twitter using a convolution-gru based deep neural network," in *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*, Springer, 2018, pp. 745–760. DOI: 10.1007/978-3-319-93417-4_48.

[16] C. of Europe's Committee of Ministers, *Recommendation cm/rec(2022)16[1] of the committee of ministers to member states on combating hate speech*. [Online]. Available: https://search.coe.int/cm/Pages/result_details.aspx?ObjectId=0900001680a67955 (visited on 12/27/2023).

[17] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th international conference on world wide web*, 2016, pp. 145–153. DOI: 10.1145/2872427.2883062.

[18] X.com. "X's policy on hateful conduct." (2023), [Online]. Available: https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy (visited on 12/27/2023).

[19] U. Nations. "What is hate speech?" (2022), [Online]. Available: https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech (visited on 12/27/2023).

[20] S. Pinker, *Enlightenment Now: The Case for Reason, Science, Humanism, and Progress* (Business book summary). Penguin Publishing Group, 2018, ISBN: 9780525427575. [Online]. Available: https://books.google.cl/books?id=hf9MDwAAQBAJ.

[21] Y. Lupu, R. Sear, N. Velásquez, *et al.*, "Offline events and online hate," *PLOS ONE*, vol. 18, no. 1, pp. 1–14, Jan. 2023. DOI: 10.1371/journal.pone.0278511.

[22] E. Ortiz-Ospina, "The rise of social media," *Our World in Data*, 2019, https://ourworldindata.org/rise-of-social-media. (visited on 12/27/2023).

[23] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," *PLOS ONE*, vol. 14, no. 8, pp. 1–16, Aug. 2019. DOI: 10.1371/journal.pone.0221152.

[24] T. Davidson, D. Warmsley, M. W. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," *CoRR*, vol. abs/1703.04009, 2017. DOI: arXiv.1703.04009. arXiv: 1703.04009.

[25] S. Zimmerman, U. Kruschwitz, and C. Fox, "Improving hate speech detection with deep learning ensembles," in *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*, 2018.

[26] M. O. Ibrohim and I. Budi, "Translated vs non-translated method for multilingual hate speech identification in twitter," *Int. J. Adv. Sci. Eng. Inf. Technol*, vol. 9, no. 4, pp. 1116–1123, 2019. DOI: 10.18517/ijaseit.9.4.8123.

[27] M. Corazza, S. Menini, E. Cabrio, S. Tonelli, and S. Villata, "A multilingual evaluation for online hate speech detection," *ACM Transactions on Internet Technology (TOIT)*, vol. 20, no. 2, pp. 1–22, 2020. DOI: 10.1145/3377323.

[28] N. Vashistha and A. Zubiaga, "Online multilingual hate speech detection: Experimenting with hindi and english social media," *Information*, vol. 12, no. 1, p. 5, 2020. DOI: 10.3390/info12010005.

[29] S. Wang, J. Liu, X. Ouyang, and Y. Sun, "Galileo at SemEval-2020 task 12: Multi-lingual learning for offensive language identification using pre-trained language models," A. Herbelot, X. Zhu, A. Palmer, N. Schneider, J. May, and E. Shutova, Eds., pp. 1448–1455, Dec. 2020. DOI: 10.18653/v1/2020.semeval-1.189.

[30] G. Wiedemann, S. M. Yimam, and C. Biemann, "UHH-LT at SemEval-2020 task 12: Fine-tuning of pre-trained transformer networks for offensive language detection," A. Herbelot, X. Zhu, A. Palmer, N. Schneider, J. May, and E. Shutova, Eds., pp. 1638–1644, Dec. 2020. DOI: 10.18653/v1/2020.semeval-1.213.

[31] L. Stappen, F. Brunn, and B. Schuller, "Cross-lingual zero-and few-shot hate speech detection utilising frozen transformer language models and axel," *arXiv preprint arXiv:2004.13850*, 2020. DOI: 10.48550/arXiv.2004.13850.

[32] S. S. Aluru, B. Mathew, P. Saha, and A. Mukherjee, "Deep learning models for multilingual hate speech detection," *arXiv preprint arXiv:2004.06465*, 2020. DOI: 10.48550/arXiv.2004.06465.

[33] E. W. Pamungkas, V. Basile, and V. Patti, "A joint learning approach with knowledge injection for zero-shot cross-lingual hate speech detection," *Information Processing & Management*, vol. 58, no. 4, p. 102 544, 2021. DOI: 10.1016/j.ipm.2021.102544.

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

[35] Y. Yang, D. Cer, A. Ahmad, *et al.*, "Multilingual universal sentence encoder for semantic retrieval," *arXiv preprint arXiv:1907.04307*, 2019. DOI: 10.48550/arXiv.1907.04307.

[36] K. Cho, B. van Merriënboer, C. Gulcehre, *et al.*, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. DOI: 10.3115/v1/D14-1179.

[37] Y. Wu, M. Schuster, Z. Chen, *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016. DOI: 10.48550/arXiv.1609.08144.

[38] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, and J. Pérez, "Spanish pre-trained bert model and evaluation data," in *PML4DC at ICLR 2020*, 2020. DOI: 10.48550/arXiv.2308.02976.

[39] H. Le, L. Vial, J. Frej, *et al.*, "Flaubert: Unsupervised language model pre-training for french," *CoRR*, vol. abs/1912.05372, 2019. DOI: 10.48550/arXiv.1912.05372.

[40] F. Souza, R. Nogueira, and R. Lotufo, "Bertimbau: Pretrained bert models for brazilian portuguese," in *Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I*, Rio Grande, Brazil: Springer-Verlag, 2020, pp. 403–417, ISBN: 978-3-030-61376-1. DOI: 10.1007/978-3-030-61377-8_28.

[41] R. Scheible, F. Thomczyk, P. Tippmann, V. Jaravine, and M. Boeker, "Gottbert: A pure german language model," *CoRR*, vol. abs/2012.02110, 2020. DOI: 10.48550/arXiv.2012.02110.

[42] M. Pàmies, E. Öhman, K. Kajava, and J. Tiedemann, "LT@Helsinki at SemEval-2020 task 12: Multilingual or language-specific BERT?," A. Herbelot, X. Zhu, A. Palmer, N. Schneider, J. May, and E. Shutova, Eds., pp. 1569–1575, Dec. 2020. DOI: 10.18653/v1/2020.semeval-1.205.

[43] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual BERT?," A. Korhonen, D. Traum, and L. Màrquez, Eds., pp. 4996–5001, Jul. 2019. DOI: 10.18653/v1/P19-1493.

[44] S. Wu and M. Dredze, "Are all languages created equal in multilingual BERT?" In *Proceedings of the 5th Workshop on Representation Learning for NLP*, Online: Association for Computational Linguistics, Jul. 2020, pp. 120–130. DOI: 10.18653/v1/2020.repl4nlp-1.16.

[45] A. Conneau, K. Khandelwal, N. Goyal, *et al.*, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, Jul. 2020, pp. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747.

[46] W. Wang, T. Watanabe, M. Hughes, T. Nakagawa, and C. Chelba, "Denoising neural machine translation training with trusted data and online data selection," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 133–143. DOI: 10.18653/v1/W18-6314.

[47] Y. Yang, G. H. Ábrego, S. Yuan, *et al.*, "Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax," *CoRR*, vol. abs/1902.08564, 2019. DOI: 10.48550/arXiv.1902.08564. arXiv: 1902.08564.

[48] P. Gage, "A new algorithm for data compression," *C Users Journal*, vol. 12, no. 2, pp. 23–38, 1994. DOI: 10.5555/177910.177914.

[49] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html.

[50] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011. DOI: doi/10.5555/1953048.2078195.

[51] T. Wolf, L. Debut, V. Sanh, *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

[52] T. Ranasinghe and M. Zampieri, "Multilingual offensive language identification with cross-lingual embeddings," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 5838–5844. DOI: 10.18653/v1/2020.emnlp-main.470. [Online]. Available: https://aclanthology.org/2020.emnlp-main.470.

Capítulo 7

ANEXOS

7.1 Anexo A - Tablas de resultados del conjunto de datos SemEval

7.1.1 Tablas de resultados - monoEN

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5207 ± 0.0091	0.4689 ± 0.0085	0.5519 ± 0.0094	0.5579 ± 0.0088	0.4759 ± 0.0093	0.5120 ± 0.0087	0.5198 ± 0.0093
ET	0.5160 ± 0.0121	0.5722 ± 0.0175	0.5066 ± 0.0107	0.5389 ± 0.0093	0.4708 ± 0.0094	0.5121 ± 0.0081	0.5117 ± 0.0112
LR	0.4890 ± 0.0221	0.4830 ± 0.0117	0.5430 ± 0.0192	0.5680 ± 0.0188	0.4996 ± 0.0166	0.5311 ± 0.0323	0.5328 ± 0.0283
LSVM	0.4821 ± 0.0138	0.4798 ± 0.0123	0.5413 ± 0.0147	0.5618 ± 0.0169	0.5091 ± 0.0155	0.5173 ± 0.0311	0.5389 ± 0.0263
PSVM	0.5291 ± 0.0088	0.4859 ± 0.0086	0.5235 ± 0.0092	0.5656 ± 0.0088	0.5104 ± 0.0091	0.5175 ± 0.0093	0.5393 ± 0.0094
RSVM	0.4919 ± 0.0101	0.5063 ± 0.0104	0.5301 ± 0.0099	0.5551 ± 0.0089	0.4896 ± 0.0092	0.5231 ± 0.0098	0.5347 ± 0.0098
RF	0.5174 ± 0.0102	0.5662 ± 0.0156	0.5116 ± 0.0096	0.5403 ± 0.0091	0.4644 ± 0.0087	0.5122 ± 0.0084	0.5129 ± 0.0090

Tabla 7.1: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en inglés.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5204 ± 0.0091	0.4296 ± 0.0084	0.5518 ± 0.0094	0.5577 ± 0.0089	0.4589 ± 0.0102	0.5103 ± 0.0087	0.5162 ± 0.0096
ET	0.5132 ± 0.0122	0.5264 ± 0.0201	0.4792 ± 0.0103	0.5325 ± 0.0089	0.4231 ± 0.0099	0.4968 ± 0.0080	0.4940 ± 0.0110
LR	0.4792 ± 0.0240	0.4268 ± 0.0135	0.5239 ± 0.0281	0.5615 ± 0.0228	0.4615 ± 0.0242	0.5104 ± 0.0483	0.5174 ± 0.0403
LSVM	0.4733 ± 0.0171	0.4241 ± 0.0152	0.5208 ± 0.0203	0.5526 ± 0.0207	0.4758 ± 0.0204	0.4938 ± 0.0482	0.5233 ± 0.0382
PSVM	0.5179 ± 0.0084	0.4458 ± 0.0080	0.4955 ± 0.0089	0.5532 ± 0.0088	0.4727 ± 0.0080	0.4946 ± 0.0088	0.5230 ± 0.0093
RSVM	0.4886 ± 0.0101	0.4659 ± 0.0094	0.5044 ± 0.0093	0.5391 ± 0.0088	0.4412 ± 0.0076	0.5017 ± 0.0095	0.5193 ± 0.0097
RF	0.5142 ± 0.0100	0.5641 ± 0.0156	0.4843 ± 0.0095	0.5318 ± 0.0090	0.4076 ± 0.0081	0.4977 ± 0.0084	0.4953 ± 0.0093

Tabla 7.2: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en inglés.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5409 ± 0.0098	0.5338 ± 0.0080	0.5584 ± 0.0112	0.6049 ± 0.0113	0.5120 ± 0.0095	0.5425 ± 0.0098	0.5470 ± 0.0118
ET	0.5631 ± 0.0098	0.5978 ± 0.0211	0.5956 ± 0.0102	0.6121 ± 0.0101	0.5688 ± 0.0090	0.5729 ± 0.0088	0.5854 ± 0.0100
LR	0.5145 ± 0.0110	0.6100 ± 0.0080	0.6507 ± 0.0132	0.6727 ± 0.0113	0.6582 ± 0.0102	0.6483 ± 0.0093	0.6392 ± 0.0123
LSVM	0.5207 ± 0.0160	0.6291 ± 0.0060	0.6623 ± 0.0122	0.6716 ± 0.0104	0.6664 ± 0.0090	0.6321 ± 0.0122	0.6360 ± 0.0127
PSVM	0.5288 ± 0.0099	0.6202 ± 0.0063	0.6608 ± 0.0121	0.6775 ± 0.0096	0.6549 ± 0.0084	0.6360 ± 0.0105	0.6556 ± 0.0112
RSVM	0.5424 ± 0.0089	0.6265 ± 0.0084	0.6707 ± 0.0114	0.6851 ± 0.0105	0.6631 ± 0.0082	0.6400 ± 0.0103	0.6457 ± 0.0118
RF	0.5647 ± 0.0096	0.5911 ± 0.0157	0.6061 ± 0.0109	0.6219 ± 0.0101	0.5695 ± 0.0085	0.5748 ± 0.0100	0.5907 ± 0.0110

Tabla 7.3: Resultados de área bajo la curva para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en inglés.

7.1.2 Tablas de resultados - monoES

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.6089 ± 0.0084	0.6823 ± 0.0102	0.5876 ± 0.0105	0.5933 ± 0.0165	0.5949 ± 0.0122	0.5572 ± 0.0139	0.6200 ± 0.0127
ET	0.6267 ± 0.0136	0.6168 ± 0.0119	0.6630 ± 0.0103	0.6840 ± 0.0120	0.6708 ± 0.0143	0.6292 ± 0.0116	0.6613 ± 0.0117
LR	0.5692 ± 0.0165	0.7019 ± 0.0143	0.6801 ± 0.0142	0.6894 ± 0.0147	0.6924 ± 0.0123	0.6612 ± 0.0172	0.6646 ± 0.0316
LSVM	0.5826 ± 0.0220	0.7053 ± 0.0111	0.6745 ± 0.0139	0.6868 ± 0.0151	0.6951 ± 0.0144	0.6514 ± 0.0398	0.6545 ± 0.0322
PSVM	0.5870 ± 0.0119	0.7054 ± 0.0087	0.6881 ± 0.0111	0.7329 ± 0.0099	0.7109 ± 0.0130	0.6711 ± 0.0158	0.6896 ± 0.0091
RSVM	0.6301 ± 0.0117	0.7107 ± 0.0094	0.6925 ± 0.0105	0.7339 ± 0.0084	0.7038 ± 0.0111	0.6648 ± 0.0135	0.6831 ± 0.0112
RF	0.6372 ± 0.0107	0.6412 ± 0.0124	0.6698 ± 0.0130	0.6948 ± 0.0099	0.6750 ± 0.0145	0.6299 ± 0.0140	0.6681 ± 0.0112

Tabla 7.4: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en castellano.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5735 ± 0.0111	0.6778 ± 0.0101	0.5776 ± 0.0107	0.5846 ± 0.0167	0.5900 ± 0.0120	0.5446 ± 0.0141	0.5878 ± 0.0134
ET	0.5889 ± 0.0150	0.4678 ± 0.0216	0.6104 ± 0.0143	0.6348 ± 0.0175	0.6214 ± 0.0170	0.5662 ± 0.0130	0.6135 ± 0.0143
LR	0.5518 ± 0.0164	0.6965 ± 0.0136	0.6633 ± 0.0223	0.6833 ± 0.0142	0.6833 ± 0.0102	0.6424 ± 0.0187	0.6402 ± 0.0393
LSVM	0.5637 ± 0.0185	0.7017 ± 0.0110	0.6627 ± 0.0144	0.6819 ± 0.0145	0.6782 ± 0.0176	0.6236 ± 0.0449	0.6272 ± 0.0522
PSVM	0.5202 ± 0.0120	0.6974 ± 0.0094	0.6768 ± 0.0111	0.7223 ± 0.0107	0.7037 ± 0.0133	0.6481 ± 0.0169	0.6711 ± 0.0097
RSVM	0.6137 ± 0.0131	0.7045 ± 0.0097	0.6840 ± 0.0108	0.7250 ± 0.0090	0.6966 ± 0.0116	0.6338 ± 0.0149	0.6493 ± 0.0131
RF	0.6033 ± 0.0127	0.5206 ± 0.0167	0.6216 ± 0.0153	0.6544 ± 0.0119	0.6323 ± 0.0162	0.5688 ± 0.0157	0.6280 ± 0.0129

Tabla 7.5: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en castellano.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5917 ± 0.0148	0.6743 ± 0.0115	0.5858 ± 0.0149	0.5681 ± 0.0168	0.6155 ± 0.0150	0.5448 ± 0.0141	0.5894 ± 0.0187
ET	0.6336 ± 0.0143	0.7265 ± 0.0206	0.7169 ± 0.0149	0.7466 ± 0.0157	0.7079 ± 0.0152	0.6263 ± 0.0179	0.7116 ± 0.0114
LR	0.5740 ± 0.0175	0.7653 ± 0.0103	0.7398 ± 0.0128	0.7528 ± 0.0143	0.7579 ± 0.0118	0.7125 ± 0.0150	0.7319 ± 0.0115
LSVM	0.5912 ± 0.0204	0.7618 ± 0.0100	0.7274 ± 0.0119	0.7479 ± 0.0146	0.7621 ± 0.0112	0.7114 ± 0.0161	0.7333 ± 0.0127
PSVM	0.6047 ± 0.0131	0.7478 ± 0.0104	0.7343 ± 0.0118	0.7960 ± 0.0094	0.7649 ± 0.0135	0.7150 ± 0.0152	0.7409 ± 0.0108
RSVM	0.6486 ± 0.0152	0.7853 ± 0.0098	0.7442 ± 0.0118	0.7994 ± 0.0091	0.7796 ± 0.0108	0.7120 ± 0.0154	0.7391 ± 0.0121
RF	0.6507 ± 0.0147	0.7568 ± 0.0145	0.7210 ± 0.0157	0.7475 ± 0.0121	0.7123 ± 0.0161	0.6247 ± 0.0158	0.7176 ± 0.0119

Tabla 7.6: Resultados de área bajo la curva para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea monolingüe en castellano.

7.1.3 Tablas de resultados - Multilingue

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5474 ± 0.0095	0.5508 ± 0.0097	0.5375 ± 0.0063	0.5479 ± 0.0069	0.5489 ± 0.0065	0.5502 ± 0.0059	0.5451 ± 0.0073
ET	0.5765 ± 0.0092	0.5741 ± 0.0074	0.5636 ± 0.0077	0.6164 ± 0.0072	0.5648 ± 0.0083	0.5537 ± 0.0066	0.5624 ± 0.0072
LR	0.5564 ± 0.0091	0.5579 ± 0.0073	0.5938 ± 0.0180	0.6171 ± 0.0124	0.5812 ± 0.0116	0.5782 ± 0.0271	0.5843 ± 0.0169
LSVM	0.5555 ± 0.0072	0.5568 ± 0.0068	0.5945 ± 0.0205	0.6188 ± 0.0121	0.5827 ± 0.0131	0.5691 ± 0.0301	0.5899 ± 0.0185
PSVM	0.5657 ± 0.0083	0.5657 ± 0.0083	0.5781 ± 0.0064	0.6322 ± 0.0063	0.5842 ± 0.0075	0.5723 ± 0.0076	0.5941 ± 0.0063
RSVM	0.5803 ± 0.0080	0.5803 ± 0.0080	0.5854 ± 0.0065	0.6322 ± 0.0073	0.5827 ± 0.0067	0.5833 ± 0.0068	0.5947 ± 0.0053
RF	0.5777 ± 0.0154	0.5846 ± 0.0091	0.5674 ± 0.0056	0.6191 ± 0.0085	0.5548 ± 0.0072	0.5545 ± 0.0074	0.5702 ± 0.0075

Tabla 7.7: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea multilingüe.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5408 ± 0.0098	0.5457 ± 0.0103	0.5375 ± 0.0063	0.5442 ± 0.0069	0.5487 ± 0.0065	0.5411 ± 0.0067	0.5443 ± 0.0073
ET	0.4723 ± 0.0098	0.4644 ± 0.0069	0.5636 ± 0.0077	0.6120 ± 0.0074	0.5645 ± 0.0083	0.5531 ± 0.0068	0.5623 ± 0.0073
LR	0.5463 ± 0.0098	0.5490 ± 0.0075	0.5920 ± 0.0197	0.6159 ± 0.0127	0.5781 ± 0.0134	0.5719 ± 0.0320	0.5813 ± 0.0217
LSVM	0.5435 ± 0.0075	0.5452 ± 0.0068	0.5921 ± 0.0236	0.6180 ± 0.0127	0.5793 ± 0.0149	0.5624 ± 0.0399	0.5856 ± 0.0193
PSVM	0.5593 ± 0.0083	0.5593 ± 0.0083	0.5749 ± 0.0064	0.6316 ± 0.0063	0.5795 ± 0.0073	0.5712 ± 0.0076	0.5939 ± 0.0063
RSVM	0.5727 ± 0.0080	0.5727 ± 0.0080	0.5821 ± 0.0064	0.6315 ± 0.0074	0.5771 ± 0.0064	0.5831 ± 0.0068	0.5946 ± 0.0053
RF	0.5543 ± 0.0125	0.5591 ± 0.0144	0.5672 ± 0.0057	0.6166 ± 0.0086	0.5532 ± 0.0070	0.5541 ± 0.0074	0.5701 ± 0.0076

Tabla 7.8: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea multilingüe.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5833 ± 0.0088	0.5853 ± 0.0097	0.5454 ± 0.0079	0.5479 ± 0.0104	0.5659 ± 0.0085	0.5575 ± 0.0068	0.5568 ± 0.0094
ET	0.6056 ± 0.0102	0.6131 ± 0.0087	0.6075 ± 0.0066	0.6575 ± 0.0065	0.6120 ± 0.0083	0.5841 ± 0.0068	0.6059 ± 0.0085
LR	0.6261 ± 0.0070	0.6244 ± 0.0077	0.6651 ± 0.0075	0.6940 ± 0.0061	0.6786 ± 0.0074	0.6480 ± 0.0072	0.6510 ± 0.0069
LSVM	0.6401 ± 0.0073	0.6408 ± 0.0075	0.6713 ± 0.0095	0.6942 ± 0.0065	0.6849 ± 0.0070	0.6336 ± 0.0068	0.6541 ± 0.0076
PSVM	0.6464 ± 0.0069	0.6464 ± 0.0069	0.6679 ± 0.0069	0.7116 ± 0.0074	0.6844 ± 0.0058	0.6454 ± 0.0072	0.6609 ± 0.0078
RSVM	0.6674 ± 0.0072	0.6674 ± 0.0072	0.6717 ± 0.0067	0.7149 ± 0.0064	0.6879 ± 0.0062	0.6448 ± 0.0070	0.6506 ± 0.0070
RF	0.5886 ± 0.0104	0.5919 ± 0.0082	0.6137 ± 0.0062	0.6630 ± 0.0074	0.6071 ± 0.0058	0.5863 ± 0.0067	0.6138 ± 0.0068

Tabla 7.9: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea multilingüe.

7.1.4 Tablas de resultados - EN→ES

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5316 ± 0.0117	0.5845 ± 0.0111	0.5585 ± 0.0127	0.5779 ± 0.0102	0.5925 ± 0.0113	0.5732 ± 0.0135	0.5548 ± 0.0151
ET	0.5730 ± 0.0106	0.5852 ± 0.0116	0.5871 ± 0.0156	0.6240 ± 0.0140	0.6361 ± 0.0113	0.5917 ± 0.0156	0.6061 ± 0.0141
LR	0.5173 ± 0.0220	0.5849 ± 0.0112	0.6090 ± 0.0112	0.6231 ± 0.0170	0.6591 ± 0.0120	0.6080 ± 0.0126	0.6181 ± 0.0159
LSVM	0.5132 ± 0.0279	0.5845 ± 0.0115	0.6190 ± 0.0125	0.6163 ± 0.0154	0.6641 ± 0.0146	0.6056 ± 0.0109	0.6164 ± 0.0153
PSVM	0.5482 ± 0.0127	0.5878 ± 0.0116	0.6043 ± 0.0109	0.6630 ± 0.0104	0.6676 ± 0.0111	0.6111 ± 0.0119	0.6277 ± 0.0115
RSVM	0.5691 ± 0.0129	0.5857 ± 0.0124	0.5971 ± 0.0111	0.6477 ± 0.0107	0.6763 ± 0.0103	0.6116 ± 0.0117	0.6281 ± 0.0110
RF	0.5581 ± 0.0144	0.5851 ± 0.0115	0.5792 ± 0.0176	0.6338 ± 0.0115	0.6405 ± 0.0114	0.5888 ± 0.0107	0.6038 ± 0.0128

Tabla 7.10: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe EN→ES.

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.4435 ± 0.0117	0.3750 ± 0.0047	0.5021 ± 0.0146	0.5670 ± 0.0107	0.5349 ± 0.0135	0.5135 ± 0.0141	0.5420 ± 0.0155
ET	0.4505 ± 0.0125	0.3695 ± 0.0049	0.5126 ± 0.0170	0.6038 ± 0.0152	0.6204 ± 0.0119	0.5237 ± 0.0193	0.5300 ± 0.0162
LR	0.5046 ± 0.0155	0.3725 ± 0.0046	0.5307 ± 0.0386	0.6185 ± 0.0153	0.6503 ± 0.0117	0.5443 ± 0.0406	0.5533 ± 0.0440
LSVM	0.5001 ± 0.0220	0.3714 ± 0.0048	0.5793 ± 0.0317	0.6132 ± 0.0143	0.6542 ± 0.0117	0.4898 ± 0.0467	0.5650 ± 0.0382
PSVM	0.4600 ± 0.0108	0.3818 ± 0.0061	0.5003 ± 0.0131	0.6505 ± 0.0110	0.6584 ± 0.0109	0.4730 ± 0.0132	0.5839 ± 0.0116
RSVM	0.4903 ± 0.0124	0.3751 ± 0.0066	0.4857 ± 0.0133	0.6389 ± 0.0111	0.6685 ± 0.0100	0.4864 ± 0.0138	0.5697 ± 0.0114
RF	0.4673 ± 0.0128	0.3693 ± 0.0044	0.5239 ± 0.0196	0.6176 ± 0.0122	0.6243 ± 0.0112	0.5064 ± 0.0143	0.5276 ± 0.0136

Tabla 7.11: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe EN→ES

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.4901 ± 0.0146	0.5597 ± 0.0102	0.5316 ± 0.0150	0.5891 ± 0.0152	0.5985 ± 0.0147	0.5436 ± 0.0134	0.5230 ± 0.0162
ET	0.5134 ± 0.0140	0.5637 ± 0.0206	0.6025 ± 0.0169	0.6599 ± 0.0127	0.6659 ± 0.0106	0.5876 ± 0.0170	0.6277 ± 0.0137
LR	0.5176 ± 0.0164	0.5801 ± 0.0140	0.6391 ± 0.0167	0.6724 ± 0.0136	0.7235 ± 0.0102	0.6401 ± 0.0186	0.6338 ± 0.0127
LSVM	0.5134 ± 0.0171	0.5579 ± 0.0124	0.6501 ± 0.0157	0.6755 ± 0.0140	0.7322 ± 0.0108	0.6452 ± 0.0156	0.6383 ± 0.0125
PSVM	0.5262 ± 0.0141	0.5878 ± 0.0130	0.6256 ± 0.0159	0.6973 ± 0.0126	0.7286 ± 0.0109	0.6559 ± 0.0117	0.6566 ± 0.0134
RSVM	0.5349 ± 0.0143	0.5600 ± 0.0127	0.6166 ± 0.0155	0.7029 ± 0.0132	0.7401 ± 0.0107	0.6576 ± 0.0116	0.6591 ± 0.0133
RF	0.4987 ± 0.0123	0.5703 ± 0.0157	0.5865 ± 0.0187	0.6686 ± 0.0127	0.6699 ± 0.0109	0.5817 ± 0.0154	0.6319 ± 0.0135

Tabla 7.12: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe EN→ES

7.1.5 Tablas de resultados - ES→EN

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5506 ± 0.0089	0.5800 ± 0.0112	0.5309 ± 0.0143	0.5665 ± 0.0076	0.5587 ± 0.0107	0.5161 ± 0.0135	0.5568 ± 0.0090
ET	0.5813 ± 0.0102	0.5812 ± 0.0100	0.5816 ± 0.0097	0.5997 ± 0.0109	0.5850 ± 0.0096	0.5319 ± 0.0115	0.5976 ± 0.0103
LR	0.5743 ± 0.0123	0.5668 ± 0.0095	0.5535 ± 0.0429	0.6219 ± 0.0066	0.5993 ± 0.0087	0.5481 ± 0.0196	0.6208 ± 0.0236
LSVM	0.5642 ± 0.0213	0.5542 ± 0.0110	0.5333 ± 0.0294	0.6232 ± 0.0096	0.6013 ± 0.0090	0.5514 ± 0.0228	0.6155 ± 0.0375
PSVM	0.5776 ± 0.0101	0.5810 ± 0.0102	0.5645 ± 0.0097	0.6449 ± 0.0082	0.6089 ± 0.0094	0.5715 ± 0.0092	0.6332 ± 0.0089
RSVM	0.5460 ± 0.0079	0.5712 ± 0.0102	0.5555 ± 0.0098	0.6420 ± 0.0083	0.6025 ± 0.0085	0.5701 ± 0.0093	0.6474 ± 0.0099
RF	0.5730 ± 0.0115	0.5813 ± 0.0100	0.5829 ± 0.0119	0.6055 ± 0.0097	0.5830 ± 0.0081	0.5128 ± 0.0132	0.6057 ± 0.0127

Tabla 7.13: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe ES→EN

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.4523 ± 0.0082	0.3865 ± 0.0321	0.5097 ± 0.0112	0.5476 ± 0.0076	0.5349 ± 0.0097	0.5077 ± 0.0122	0.5469 ± 0.0091
ET	0.3996 ± 0.0086	0.3676 ± 0.0040	0.4454 ± 0.0139	0.4765 ± 0.0121	0.4682 ± 0.0097	0.5147 ± 0.0144	0.5592 ± 0.0132
LR	0.3878 ± 0.0142	0.4186 ± 0.0181	0.5302 ± 0.0625	0.5916 ± 0.0107	0.5597 ± 0.0238	0.5215 ± 0.0141	0.6125 ± 0.0241
LSVM	0.4133 ± 0.0180	0.4573 ± 0.0088	0.5065 ± 0.0437	0.5951 ± 0.0188	0.5780 ± 0.0192	0.5210 ± 0.0151	0.5891 ± 0.0429
PSVM	0.3877 ± 0.0056	0.3736 ± 0.0049	0.5599 ± 0.0095	0.6202 ± 0.0094	0.5881 ± 0.0104	0.5397 ± 0.0099	0.6199 ± 0.0089
RSVM	0.4342 ± 0.0071	0.3953 ± 0.0068	0.5386 ± 0.0095	0.6236 ± 0.0088	0.5722 ± 0.0107	0.5410 ± 0.0096	0.6390 ± 0.0098
RF	0.4191 ± 0.0108	0.3676 ± 0.0040	0.4639 ± 0.0175	0.4948 ± 0.0096	0.4990 ± 0.0106	0.5047 ± 0.0116	0.5855 ± 0.0127

Tabla 7.14: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe ES→EN

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.4928 ± 0.0095	0.4910 ± 0.0095	0.5142 ± 0.0143	0.5220 ± 0.0120	0.5403 ± 0.0106	0.5093 ± 0.0119	0.5204 ± 0.0143
ET	0.5131 ± 0.0128	0.5463 ± 0.0187	0.5927 ± 0.0168	0.6254 ± 0.0130	0.6000 ± 0.0099	0.5252 ± 0.0143	0.6150 ± 0.0138
LR	0.4856 ± 0.0105	0.5311 ± 0.0110	0.6823 ± 0.0126	0.6610 ± 0.0067	0.6266 ± 0.0079	0.5576 ± 0.0136	0.6886 ± 0.0132
LSVM	0.5001 ± 0.0249	0.5241 ± 0.0109	0.6718 ± 0.0153	0.6620 ± 0.0079	0.6385 ± 0.0089	0.5578 ± 0.0132	0.6878 ± 0.0106
PSVM	0.4992 ± 0.0144	0.5128 ± 0.0115	0.6262 ± 0.0103	0.6798 ± 0.0095	0.6269 ± 0.0110	0.5801 ± 0.0113	0.6829 ± 0.0110
RSVM	0.4777 ± 0.0112	0.5182 ± 0.0112	0.6739 ± 0.0095	0.6858 ± 0.0087	0.6303 ± 0.0114	0.5864 ± 0.0109	0.6902 ± 0.0110
RF	0.5092 ± 0.0144	0.5544 ± 0.0121	0.5827 ± 0.0134	0.6384 ± 0.0109	0.5851 ± 0.0091	0.5105 ± 0.0165	0.6285 ± 0.0155

Tabla 7.15: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de SemEval y para la tarea translingüe ES→EN

7.2 Anexo B - Tablas de resultados del conjunto de datos Convención Constituyente

7.2.1 Tarea Monolingüe en castellano

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.7729 ± 0.0074	0.7356 ± 0.0102	0.6276 ± 0.0173	0.6538 ± 0.0219	0.7038 ± 0.0175	0.6318 ± 0.0168	0.7619 ± 0.0635
et	0.8603 ± 0.0071	0.8939 ± 0.0056	0.8877 ± 0.0096	0.8883 ± 0.0172	0.8868 ± 0.0102	0.8426 ± 0.0237	0.8767 ± 0.0120
lr	0.6825 ± 0.1238	0.7864 ± 0.0185	0.7925 ± 0.0511	0.8025 ± 0.0322	0.8468 ± 0.0193	0.6960 ± 0.0939	0.8091 ± 0.0661
lsvm	0.6812 ± 0.1623	0.7832 ± 0.0145	0.7999 ± 0.0340	0.7886 ± 0.0301	0.8522 ± 0.0187	0.7087 ± 0.1230	0.8137 ± 0.0746
poly-svm	0.7696 ± 0.0083	0.7885 ± 0.0061	0.8443 ± 0.0083	0.8148 ± 0.0093	0.8234 ± 0.0082	0.7783 ± 0.0085	0.8483 ± 0.0091
rbf-svm	0.7495 ± 0.0093	0.7813 ± 0.0070	0.8292 ± 0.0098	0.8174 ± 0.0089	0.8413 ± 0.0077	0.8046 ± 0.0077	0.8638 ± 0.0102
rf	0.8401 ± 0.0098	0.8951 ± 0.0057	0.8827 ± 0.0097	0.8889 ± 0.0105	0.8759 ± 0.0166	0.8558 ± 0.0146	0.8660 ± 0.0098

Tabla 7.16: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea monolingüe en castellano

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.4926 ± 0.0093	0.5007 ± 0.0074	0.4671 ± 0.0130	0.4723 ± 0.0191	0.5007 ± 0.0141	0.4548 ± 0.0138	0.5174 ± 0.0247
et	0.5150 ± 0.0120	0.4739 ± 0.0033	0.5066 ± 0.0181	0.4941 ± 0.0179	0.4840 ± 0.0151	0.5007 ± 0.0145	0.5212 ± 0.0198
lr	0.4745 ± 0.0419	0.5098 ± 0.0097	0.5374 ± 0.0200	0.5412 ± 0.0191	0.5184 ± 0.0202	0.5100 ± 0.0383	0.5376 ± 0.0251
lsvm	0.4645 ± 0.0694	0.5212 ± 0.0098	0.5388 ± 0.0175	0.5364 ± 0.0198	0.5174 ± 0.0215	0.5083 ± 0.0509	0.5347 ± 0.0288
poly-svm	0.5048 ± 0.0072	0.5104 ± 0.0072	0.5500 ± 0.0222	0.5556 ± 0.0197	0.5255 ± 0.0175	0.5489 ± 0.0135	0.5582 ± 0.0152
rbf-svm	0.5006 ± 0.0076	0.5214 ± 0.0077	0.5528 ± 0.0155	0.5569 ± 0.0187	0.5255 ± 0.0185	0.5504 ± 0.0128	0.5645 ± 0.0160
rf	0.5188 ± 0.0119	0.4746 ± 0.0037	0.5118 ± 0.0141	0.4993 ± 0.0147	0.4974 ± 0.0189	0.5031 ± 0.0165	0.5252 ± 0.0160

Tabla 7.17: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea monolingüe en castellano

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.5171 ± 0.0142	0.5158 ± 0.0100	0.5184 ± 0.0333	0.5106 ± 0.0280	0.5398 ± 0.0296	0.4913 ± 0.0243	0.5515 ± 0.0276
et	0.5337 ± 0.0177	0.5610 ± 0.0188	0.6009 ± 0.0312	0.5905 ± 0.0336	0.5465 ± 0.0334	0.5304 ± 0.0295	0.5932 ± 0.0313
lr	0.5225 ± 0.0201	0.5740 ± 0.0143	0.6269 ± 0.0218	0.6203 ± 0.0295	0.5916 ± 0.0248	0.6075 ± 0.0275	0.6366 ± 0.0242
lsvm	0.5229 ± 0.0239	0.5713 ± 0.0157	0.6248 ± 0.0229	0.6209 ± 0.0277	0.5939 ± 0.0274	0.6073 ± 0.0279	0.6374 ± 0.0258
poly-svm	0.5094 ± 0.0141	0.5545 ± 0.0117	0.6430 ± 0.0211	0.6636 ± 0.0221	0.5769 ± 0.0244	0.6208 ± 0.0218	0.6535 ± 0.0268
rbf-svm	0.5140 ± 0.0122	0.5627 ± 0.0146	0.6354 ± 0.0285	0.6693 ± 0.0212	0.6053 ± 0.0217	0.6124 ± 0.0225	0.6535 ± 0.0241
rf	0.5451 ± 0.0205	0.5572 ± 0.0143	0.5937 ± 0.0334	0.6017 ± 0.0274	0.5625 ± 0.0268	0.5453 ± 0.0278	0.5899 ± 0.0300

Tabla 7.18: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea monolingüe en castellano

7.2.2 Tarea Multilingüe

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.5474 ± 0.0095	0.5508 ± 0.0097	0.5375 ± 0.0063	0.5479 ± 0.0069	0.5489 ± 0.0065	0.5502 ± 0.0059	0.5451 ± 0.0073
et	0.5765 ± 0.0092	0.5741 ± 0.0074	0.5636 ± 0.0077	0.6164 ± 0.0072	0.5648 ± 0.0083	0.5537 ± 0.0066	0.5624 ± 0.0072
lr	0.5564 ± 0.0091	0.5579 ± 0.0073	0.5938 ± 0.0180	0.6171 ± 0.0124	0.5812 ± 0.0116	0.5782 ± 0.0271	0.5843 ± 0.0169
lsvm	0.5555 ± 0.0072	0.5568 ± 0.0068	0.5945 ± 0.0205	0.6188 ± 0.0121	0.5827 ± 0.0131	0.5691 ± 0.0301	0.5899 ± 0.0185
poly-svm	0.5657 ± 0.0083	0.5657 ± 0.0083	0.5781 ± 0.0064	0.6322 ± 0.0063	0.5842 ± 0.0075	0.5723 ± 0.0076	0.5941 ± 0.0063
rbf-svm	0.5803 ± 0.0080	0.5803 ± 0.0080	0.5854 ± 0.0065	0.6322 ± 0.0073	0.5827 ± 0.0067	0.5833 ± 0.0068	0.5947 ± 0.0053
rf	0.5777 ± 0.0154	0.5846 ± 0.0091	0.5674 ± 0.0056	0.6191 ± 0.0085	0.5548 ± 0.0072	0.5545 ± 0.0074	0.5702 ± 0.0075

Tabla 7.19: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea multilingüe

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5408 ± 0.0098	0.5457 ± 0.0103	0.5375 ± 0.0063	0.5442 ± 0.0069	0.5487 ± 0.0065	0.5411 ± 0.0067	0.5443 ± 0.0073
et	0.4723 ± 0.0098	0.4644 ± 0.0069	0.5636 ± 0.0077	0.6120 ± 0.0074	0.5645 ± 0.0083	0.5531 ± 0.0068	0.5623 ± 0.0073
lr	0.5463 ± 0.0098	0.5490 ± 0.0075	0.5920 ± 0.0197	0.6159 ± 0.0127	0.5781 ± 0.0134	0.5719 ± 0.0320	0.5813 ± 0.0217
lsvm	0.5435 ± 0.0075	0.5452 ± 0.0068	0.5921 ± 0.0236	0.6180 ± 0.0127	0.5793 ± 0.0149	0.5624 ± 0.0399	0.5856 ± 0.0193
poly-svm	0.5593 ± 0.0083	0.5593 ± 0.0083	0.5749 ± 0.0064	0.6316 ± 0.0063	0.5795 ± 0.0073	0.5712 ± 0.0076	0.5939 ± 0.0063
rbf-svm	0.5727 ± 0.0080	0.5727 ± 0.0080	0.5821 ± 0.0064	0.6315 ± 0.0074	0.5771 ± 0.0064	0.5831 ± 0.0068	0.5946 ± 0.0053
rf	0.5543 ± 0.0125	0.5591 ± 0.0144	0.5672 ± 0.0057	0.6166 ± 0.0086	0.5532 ± 0.0070	0.5541 ± 0.0074	0.5701 ± 0.0076

Tabla 7.20: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea multilingüe

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.5833 ± 0.0088	0.5853 ± 0.0097	0.5454 ± 0.0079	0.5479 ± 0.0104	0.5659 ± 0.0085	0.5575 ± 0.0068	0.5568 ± 0.0094
et	0.6056 ± 0.0102	0.6131 ± 0.0087	0.6075 ± 0.0066	0.6575 ± 0.0065	0.6120 ± 0.0083	0.5841 ± 0.0068	0.6059 ± 0.0085
lr	0.6261 ± 0.0070	0.6244 ± 0.0077	0.6651 ± 0.0075	0.6940 ± 0.0061	0.6786 ± 0.0074	0.6480 ± 0.0072	0.6510 ± 0.0069
lsvm	0.6401 ± 0.0073	0.6408 ± 0.0075	0.6713 ± 0.0095	0.6942 ± 0.0065	0.6849 ± 0.0070	0.6336 ± 0.0068	0.6541 ± 0.0076
poly-svm	0.6464 ± 0.0069	0.6464 ± 0.0069	0.6679 ± 0.0069	0.7116 ± 0.0074	0.6844 ± 0.0058	0.6454 ± 0.0072	0.6609 ± 0.0078
rbf-svm	0.6674 ± 0.0072	0.6674 ± 0.0072	0.6717 ± 0.0067	0.7149 ± 0.0064	0.6879 ± 0.0062	0.6448 ± 0.0070	0.6506 ± 0.0070
rf	0.5886 ± 0.0104	0.5919 ± 0.0082	0.6137 ± 0.0062	0.6630 ± 0.0074	0.6071 ± 0.0058	0.5863 ± 0.0067	0.6138 ± 0.0068

Tabla 7.21: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea multilingüe

7.2.3 Tarea translingüe

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.7336 ± 0.0072	0.8943 ± 0.0058	0.6965 ± 0.0101	0.5360 ± 0.0143	0.7165 ± 0.0107	0.6868 ± 0.0123	0.5370 ± 0.0173
et	0.7899 ± 0.0129	0.8955 ± 0.0057	0.7048 ± 0.0378	0.7107 ± 0.0144	0.5964 ± 0.0184	0.7514 ± 0.0208	0.7974 ± 0.0159
lr	0.5925 ± 0.1486	0.8925 ± 0.0059	0.8476 ± 0.0383	0.6871 ± 0.0346	0.7052 ± 0.0501	0.7145 ± 0.0816	0.7423 ± 0.1303
lsvm	0.6232 ± 0.0853	0.8935 ± 0.0057	0.8208 ± 0.0559	0.6600 ± 0.0410	0.7021 ± 0.0604	0.7508 ± 0.0860	0.7186 ± 0.1692
poly-svm	0.6862 ± 0.0071	0.8833 ± 0.0064	0.8370 ± 0.0095	0.7383 ± 0.0123	0.6916 ± 0.0100	0.8229 ± 0.0096	0.7800 ± 0.0110
rbf-svm	0.7786 ± 0.0053	0.8914 ± 0.0056	0.8273 ± 0.0085	0.7261 ± 0.0104	0.7411 ± 0.0114	0.8023 ± 0.0096	0.7975 ± 0.0091
rf	0.7507 ± 0.0097	0.8954 ± 0.0057	0.7193 ± 0.0378	0.7029 ± 0.0166	0.5790 ± 0.0189	0.7454 ± 0.0207	0.7998 ± 0.0184

Tabla 7.22: Resultados de exactitud para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea translingüe EN→ES

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
DT	0.5125 ± 0.0077	0.4751 ± 0.0036	0.4927 ± 0.0095	0.4302 ± 0.0113	0.4944 ± 0.0156	0.4840 ± 0.0104	0.4188 ± 0.0110
et	0.5171 ± 0.0098	0.4724 ± 0.0016	0.4974 ± 0.0193	0.5008 ± 0.0148	0.4565 ± 0.0140	0.5071 ± 0.0152	0.5162 ± 0.0179
lr	0.4305 ± 0.0579	0.4743 ± 0.0027	0.5195 ± 0.0221	0.4962 ± 0.0164	0.5100 ± 0.0221	0.4979 ± 0.0314	0.5020 ± 0.0612
lsvm	0.4496 ± 0.0327	0.4723 ± 0.0018	0.5284 ± 0.0176	0.4886 ± 0.0182	0.5118 ± 0.0211	0.5043 ± 0.0326	0.4927 ± 0.0874
poly-svm	0.4704 ± 0.0066	0.4873 ± 0.0043	0.5503 ± 0.0159	0.5088 ± 0.0138	0.5055 ± 0.0120	0.5267 ± 0.0125	0.5406 ± 0.0227
rbf-svm	0.4916 ± 0.0065	0.4786 ± 0.0054	0.5426 ± 0.0182	0.5085 ± 0.0113	0.5117 ± 0.0200	0.5196 ± 0.0143	0.5354 ± 0.0213
rf	0.5061 ± 0.0088	0.4726 ± 0.0018	0.4961 ± 0.0190	0.5014 ± 0.0166	0.4518 ± 0.0129	0.5033 ± 0.0130	0.5157 ± 0.0174

Tabla 7.23: Resultados de puntaje F_1 para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea translingüe EN→ES

Modelo	BOC	BOW	infoXLM	LaBSE	LASER	mBERT	XLM-RoBERTa
dt	0.5268 ± 0.0145	0.5005 ± 0.0083	0.5375 ± 0.0343	0.5471 ± 0.0180	0.5254 ± 0.0300	0.5052 ± 0.0225	0.4870 ± 0.0302
et	0.5349 ± 0.0156	0.4985 ± 0.0145	0.5614 ± 0.0295	0.5758 ± 0.0254	0.5571 ± 0.0241	0.5334 ± 0.0266	0.5484 ± 0.0320
lr	0.4909 ± 0.0217	0.5311 ± 0.0129	0.5900 ± 0.0273	0.5917 ± 0.0276	0.6128 ± 0.0186	0.5559 ± 0.0275	0.5761 ± 0.0325
lsvm	0.4830 ± 0.0208	0.5388 ± 0.0130	0.5976 ± 0.0236	0.5939 ± 0.0270	0.6218 ± 0.0229	0.5521 ± 0.0285	0.5897 ± 0.0319
poly-svm	0.4997 ± 0.0158	0.5162 ± 0.0118	0.5799 ± 0.0240	0.5759 ± 0.0248	0.6055 ± 0.0198	0.5785 ± 0.0241	0.6099 ± 0.0291
rbf-svm	0.5109 ± 0.0133	0.5086 ± 0.0128	0.5752 ± 0.0239	0.5862 ± 0.0260	0.5987 ± 0.0192	0.5735 ± 0.0229	0.6001 ± 0.0312
rf	0.5275 ± 0.0147	0.4959 ± 0.0149	0.5493 ± 0.0312	0.5797 ± 0.0295	0.5729 ± 0.0241	0.5381 ± 0.0236	0.5532 ± 0.0283

Tabla 7.24: Resultados deárea bajo la curva ROC para todos los modelos base y representaciones, para el conjunto de datos de la Convención Constituyente y para la tarea translingüe EN→ES

7.3 Anexo C - Diagramas de cajas y bigotes presentando el resumen de los resultados obtenidos en las distintas tareas, para ambos conjuntos de datos

7.3.1 Conjunto de datos SemEval

Tarea monoEN

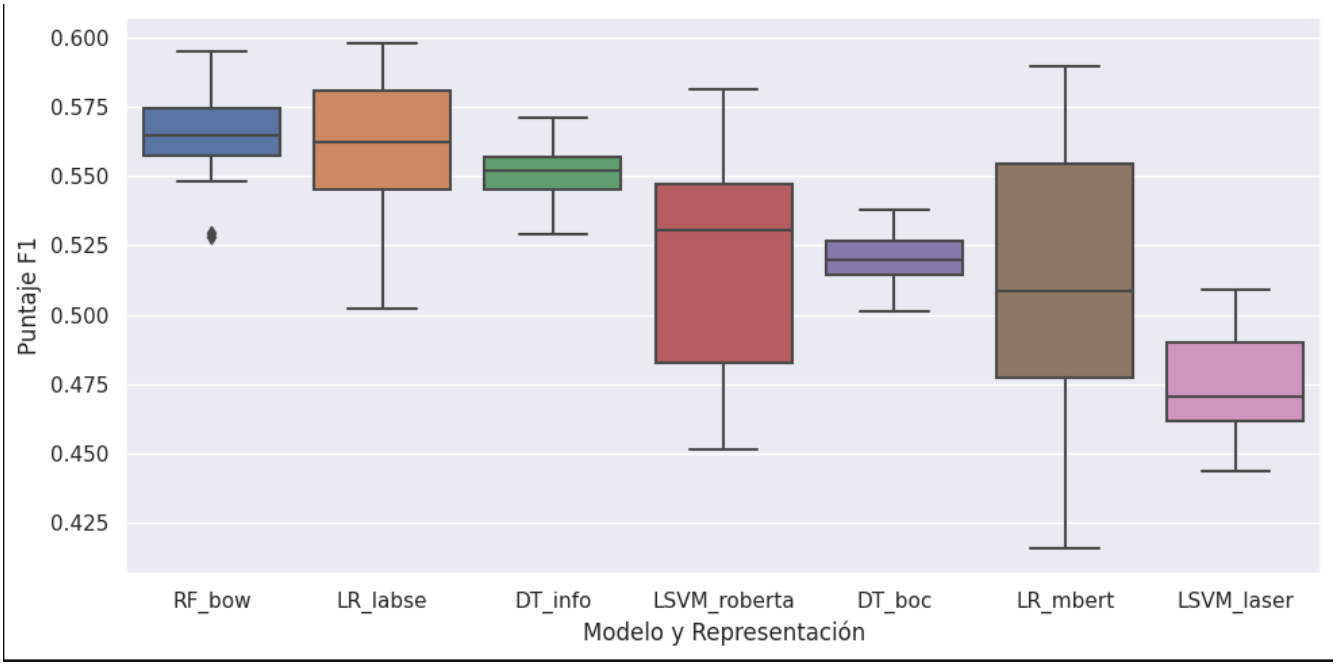


Figura 7.1: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea monolingüe en inglés, para el conjunto de datos SemEval.

Tarea monoES

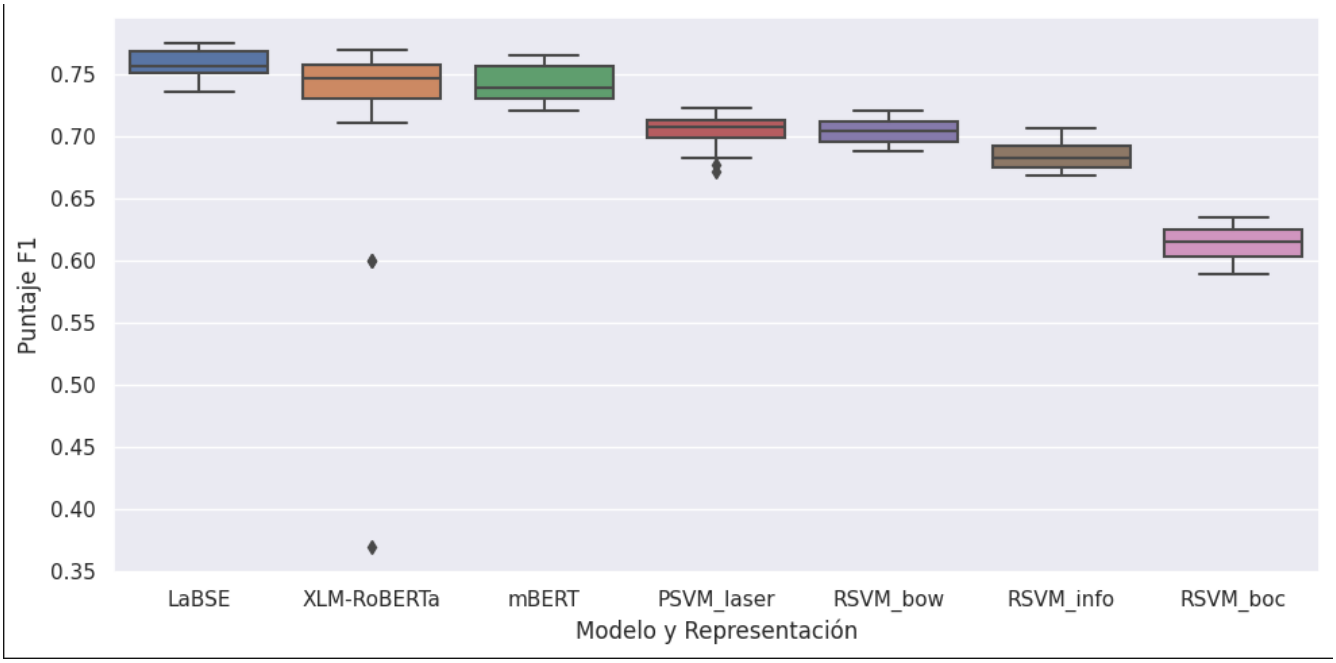


Figura 7.2: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea monolingüe en castellano, para el conjunto de datos SemEval.

Tarea Multilingüe

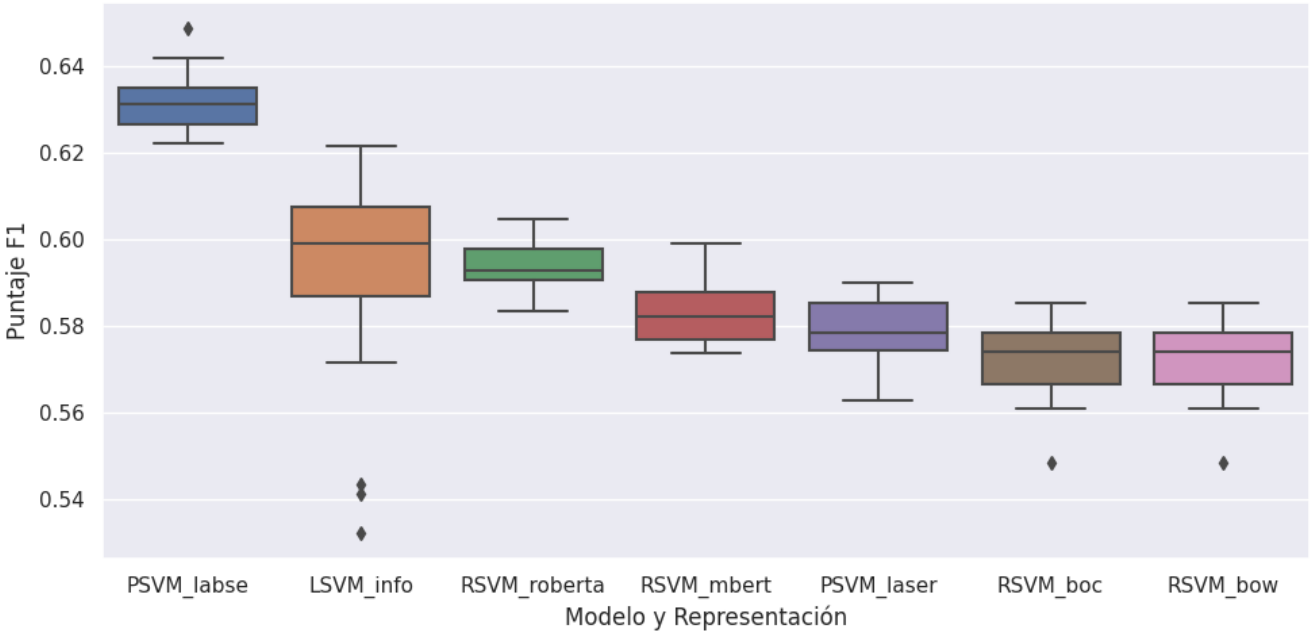


Figura 7.3: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea multilingüe, para el conjunto de datos SemEval.

Tarea Translingüe EN→ES

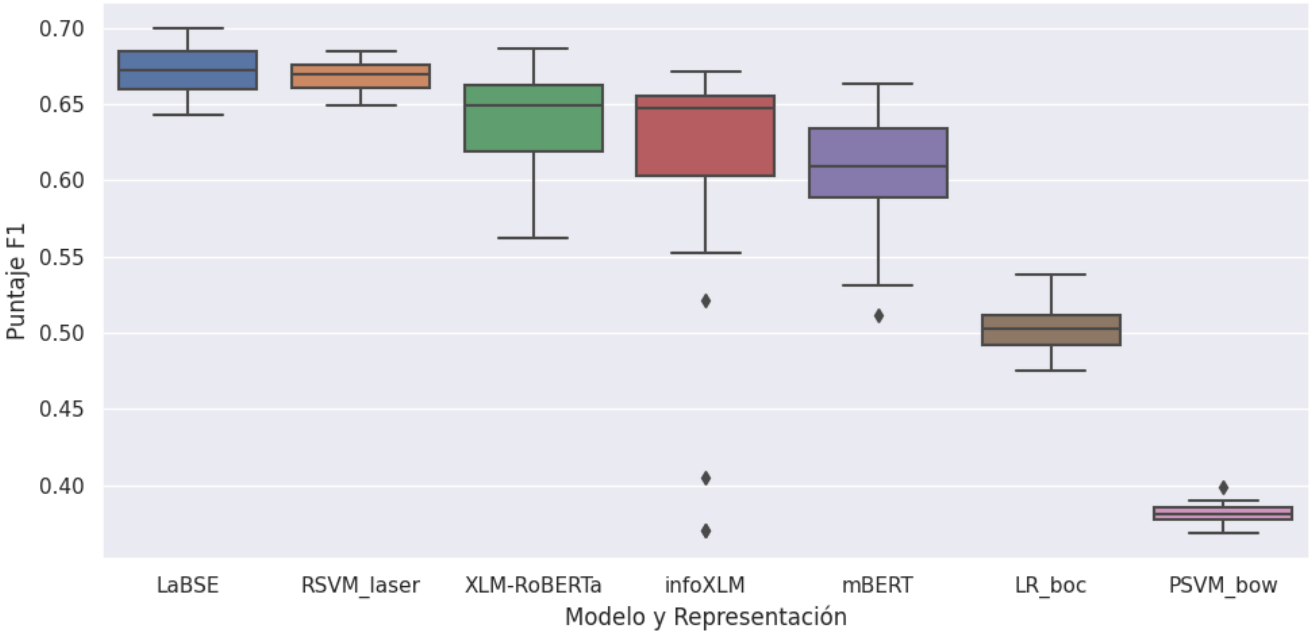


Figura 7.4: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea translingüe entrenando en inglés y evaluando en castellano, para el conjunto de datos SemEval.

Tarea Translingüe ES→EN

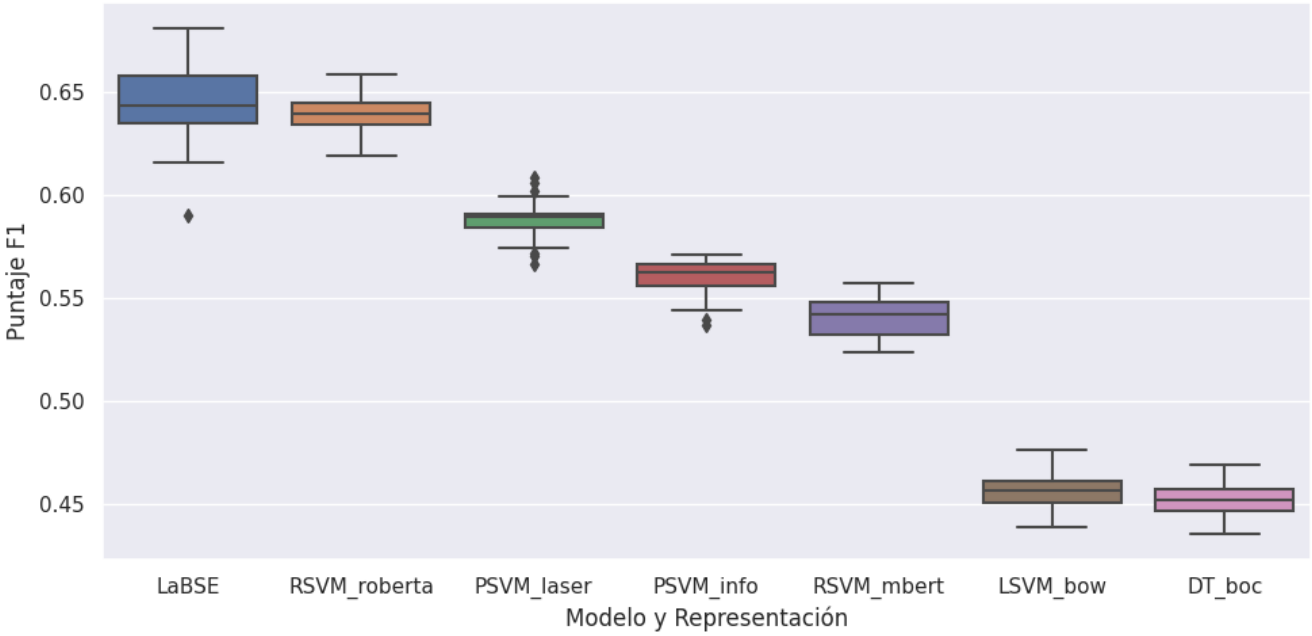


Figura 7.5: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea translingüe entrenando en castellano, y evaluando en inglés, para el conjunto de datos SemEval.

7.3.2 Conjunto de datos de la Convención Constituyente

Tarea monoES

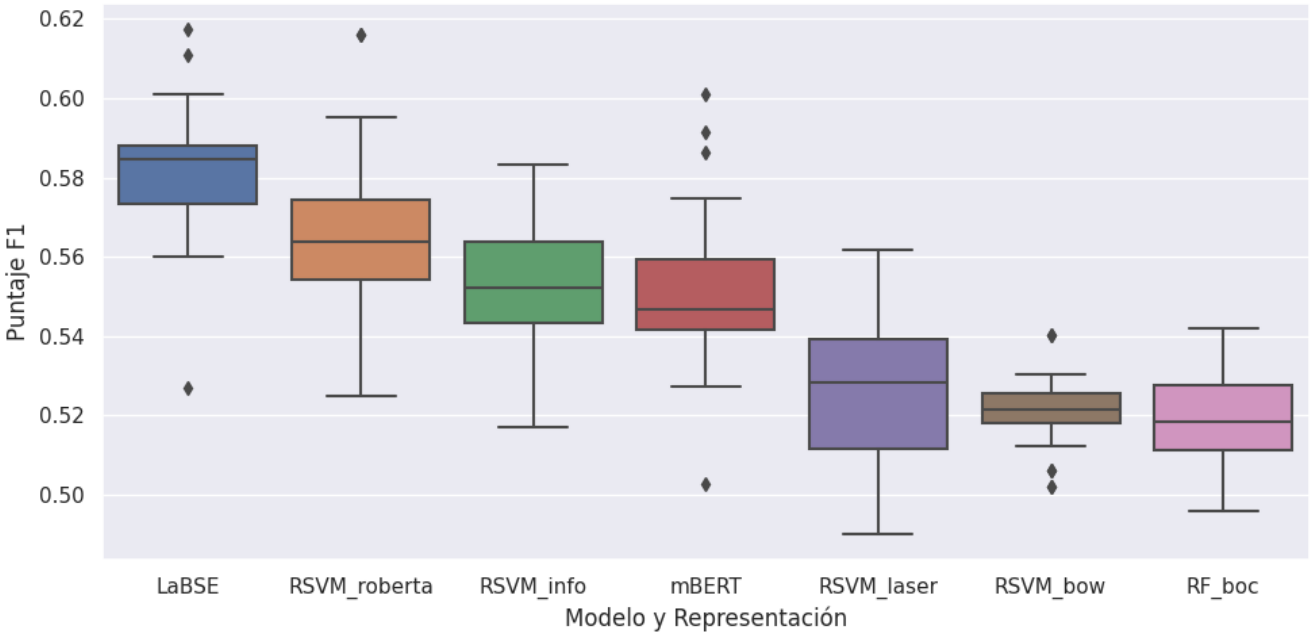


Figura 7.6: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea monolingüe en castellano, en el conjunto de datos de la Convención Constituyente.

Tarea Multilingüe

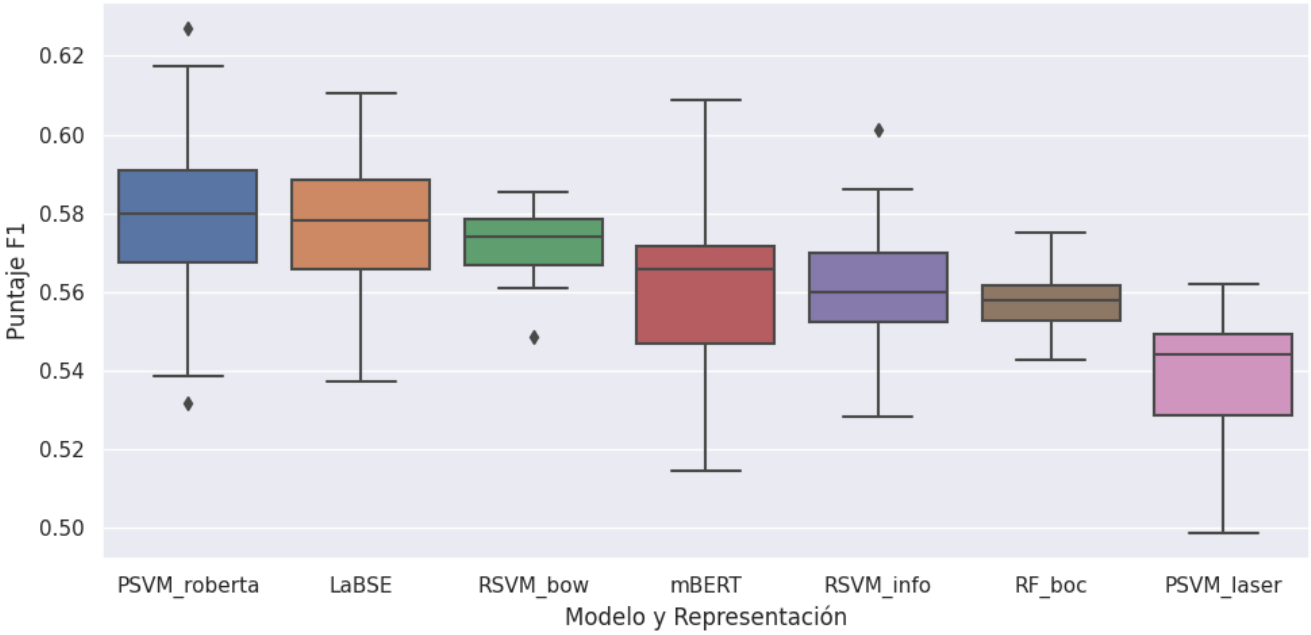


Figura 7.7: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea monolingüe en castellano, en el conjunto de datos de la Convención Constituyente.

Tarea Translingüe EN→ES

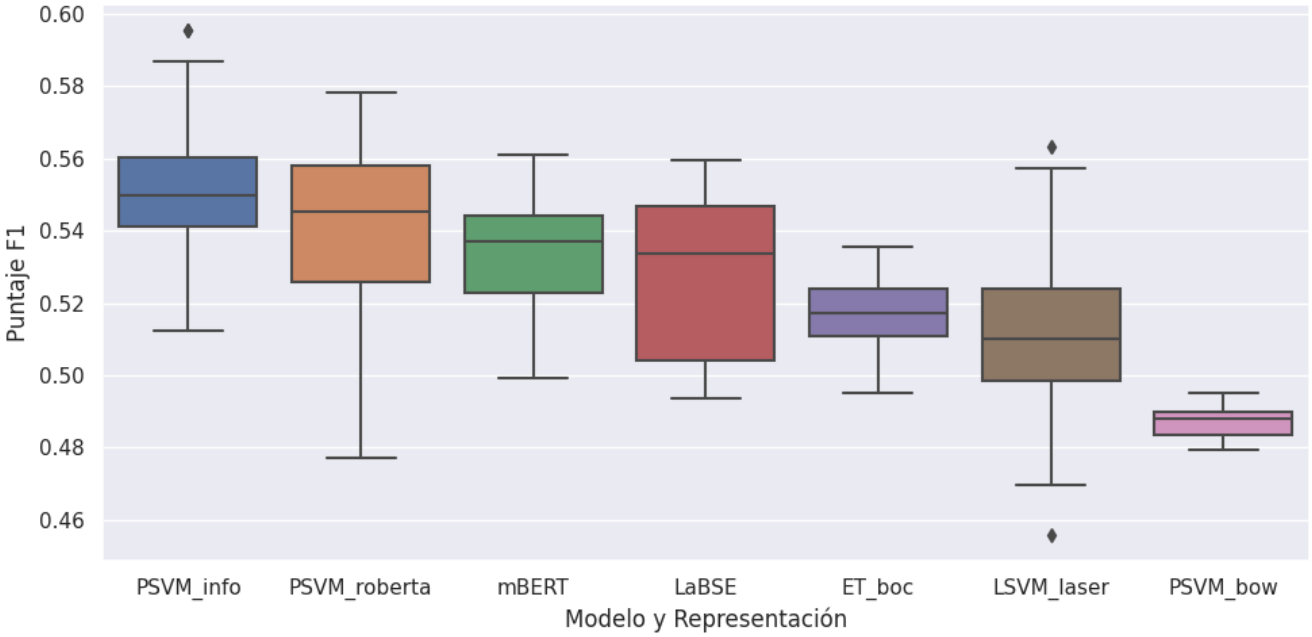


Figura 7.8: Diagrama de cajas y bigotes presentando los mejores resultados por método de representación utilizados y ordenados de mayor a menor puntaje F_1 para la tarea translingüe entrenando en inglés, y evaluando en castellano, en el conjunto de datos de la Convención Constituyente.