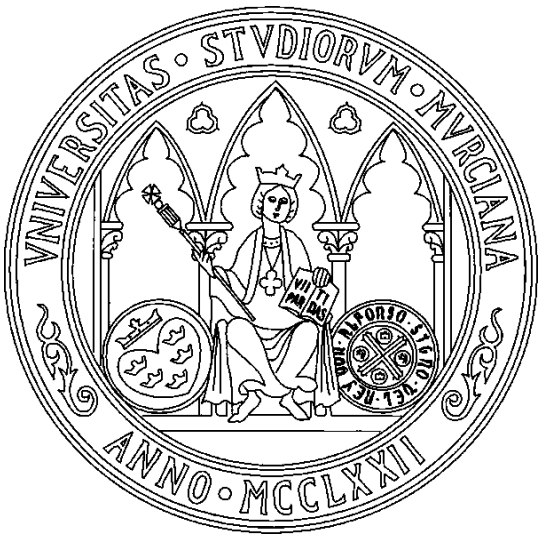


Universidad de Murcia

Facultad de Informática

Trabajo Final de Grado

Compilación y evaluación de un corpus sobre emociones relacionadas con la salud mental aplicando modelos del lenguaje

---

Autor:
Alejandro Salmerón Ríos

Tutor:
Rafael Valencia García

Convocatoria:
2 de Junio de 2022

Cotutor:
José Antonio García Díaz

A todas la personas que me han apoyado y ayudado a finalizar este camino. Gracias.

> *"Las emociones de un hombre son lo que lo definen, y el control es la marca de la verdadera fuerza. Carecer de sentimiento es estar muerto, pero actuar al dictado de cada sentimiento es ser un niño."*
>
> - Brandon Sanderson, *El camino de los reyes.*

Índice general

Índice de figuras

Índice de cuadros

Glosario

ALPAC - Automatic Language Processing Advisory Committee. 17

Automatic Language Processing Advisory Committee - Fue un comité de siete científicos dirigido por John R. Pierce, establecido en 1964 por el gobierno de los Estados Unidos para evaluar el progreso de la lingüística computacional en general y la traducción automática en particular. 17

Centro de Investigación Biomédica en Red de Salud Mental - Centro cuyo principal objetivo el realizar investigación de excelencia que redunde en la prevención de los trastornos mentales y por ende, en una mejor calidad de vida de la población en general así como en mejores tratamientos para aquellos que padecen algún tipo de trastorno mental. 19

CIBERSAM - Centro de Investigación Biomédica en Red de Salud Mental. 19

INE - Instituto Nacional de Estadística. 19

Instituto Nacional de Estadística - Organismo autónomo de carácter administrativo, con personalidad jurídica y patrimonio propio, adscrito al Ministerio de Asuntos Económicos y Transformación Digital a través de la Secretaría de Estado de Economía y Apoyo a la Empresa. 19

Machine Translation - Los sistemas de traducción automática son aplicaciones que utilizan tecnologías de aprendizaje automático para traducir grandes cantidades de texto desde y hacia cualquiera de sus idiomas admitidos. El servicio traduce un texto ”fuente” de un idioma a un idioma ”objetivo” diferente. 16

Massachusetts Institute of Technology - Universidad privada localizada en Cambridge, Massachusetts (Estados Unidos) considerada por numerosos rankings como una de las mejores y más prestigiosas universidades a nivel mundial, manteniendo

durante diez años consecutivos el título de la mejor universidad del mundo según la clasificación mundial de universidades. 17

MIT - Massachusetts Institute of Technology. 17

MT - Machine Translation. 16, 17

Natural Language Understanding - Subcampo dentro del procesamiento del lenguaje natural en inteligencia artificial que se ocupa de la comprensión de lectura automática. 19

NLU - Natural Language Understanding. 19

OMS - Organización Mundial de la Salud. 19

PLN - Procesamiento del Lenguaje Natural. 15–20, 29, 41, 42, 73

Resumen

El Análisis de Emociones es un área enmarcada en el dominio de la clasificación de texto cuyo principal objetivo dentro del Procesamiento del Lenguaje Natural es la identificación de emociones en base a un texto. El auge actual de las redes sociales y motivos como poder reconocer el estado de ánimo de un usuario en un determinado momento, han impulsado este campo a una continua investigación y desarrollo. Este interés, ligado a los avances y mejoras que también ha sufrido la investigación sobre el Procesamiento del Lenguaje Natural, han proporcionado nuevas técnicas que obtienen resultados muy llamativos y favorables sobre este tipo de tareas, como los Transformers.

El objetivo de este trabajo es hacer uso de dichas herramientas para clasificar un conjunto definido de 14 emociones. Tratando de dar un enfoque distinto y realizar una selección de emociones y estados que no estén enmarcados en las emociones básicas comúnmente utilizadas en este ámbito (ira, asco, miedo, alegría, tristeza y sorpresa). Como pueden ser la soledad, la depresión o la desesperanza. Para ello, se realizará la compilación de un corpus específico para este trabajo y la evaluación de distintos modelos del lenguaje sobre el mismo. Lo que supondrá que se aborden puntos como la selección de los diferentes corpus utilizados, el procesamiento realizado sobre ellos, así como la traducción necesaria de algunos. Siendo este último, un punto importante de este trabajo ya que fue de gran utilidad debido a la falta de datos en español sobre ciertas emociones. A su vez, se presentará una visión del estado del arte tanto en el área de la clasificación de texto como de emociones.

Ofreciendo finalmente un análisis de los modelos del lenguaje evaluados sobre esta tarea, con el objetivo de proporcionar una visión del desempeño obtenido por este tipo de modelos sobre conjuntos de emociones más complejas. Lo cual, dará pie a plantear posibles vías futuras de investigación que podrían resultar de interés en relación al análisis de emociones.

Extended abstract

This project has been possible thanks to Rafael Valencia García, my mentor, who from the beginning approved the possibility of adapting the original idea and focusing it on Emotional Analysis. Also a special thanks to Jose Antonio García Díaz, co-tutor of this project, who has helped me in every possible way.

Emotion Analysis is an area within the domain of Text Classification whose main objective within Natural Language Processing is the identification of emotions on a text basis. Reasons such as the current boom of social networks and its increased use, have led users to share their thoughts and feelings more and more through social networks. As a result, a large amount of data and an active interest in analyzing it have been generated, making this field more and more attractive every day. In addition, advances and improvements in Natural Language Processing research have provided innovative techniques that obtain very significant and favorable results on different types of Natural Language tasks.

The main objective of this work is the compilation and analysis of a corpus through language models. It has been conducted on a set of 14 emotions different from basic emotions according to its relation with mental health. In order to achieve the different objectives, including the set of emotions, the first step was to identify the different taxonomies of emotions that could be used. Among them, it is worth highlighting the taxonomy proposed by Ekman, consisting of 7 basic emotions (anger, surprise, disgust, enjoyment, fear, and sadness). The ones of greatest interest for the present project were anger, fear and sadness. Another taxonomy considered was Plutchik's wheel of emotions, designed to help classify emotions into primary emotions and the responses given to those emotions. Due to the fact that it is vaster than the previous one and considers other more complex emotions, it helped in the selection of some of them, such as grief, disgust and remorse. As for the rest of the emotions, different articles dealing with depression were used, particularly the one proposed by Angela Leis et.al, providing a list of words that could suggest signs of depression. Although certain words may suggest this state, some of them may reflect another emotion. Bearing this idea in mind, the words with the highest scores were selected, identifying them as words commonly used in texts that could

suggest signs of depression, and an attempt was made to identify possible alternative emotions and states they could reflect, such as loneliness, nervousness, disappointment, suicidal thoughts, hopelessness, and embarrassment. Finally, the neutral class was added to the set since most of the time texts do not contain an implicit emotion in them, and therefore it was considered important to add it to the final set in order to be able to identify it. This resulted in a total set of 14 emotions: anger, fear, sadness, grief, disgust, remorse, loneliness, nervousness, disappointment, suicidal, hopeless, embarrassment, depression and neutral.

Once the emotions to be used were decided, the next step was to search for and compile different datasets of data that could meet these criteria. For this purpose, we used both the Google search engine for data localization (Google Dataset Search [30]) and different articles related to specific emotions, such as loneliness or suicide. The first three sets come from sundry corpus originally in Spanish. The first one comes from the article *An Approach using BETO on Spanish Tweets* [3], presented in 2021 at the International Workshop on Software Engineering Automation: A Natural Language Perspective (NLP-SEA), providing all the resources used in it. The resource of greatest interest and selected among them contains a set of 5260 Spanish sentences classified as: sadness, not-relevant, fear, happiness, anger or surprise. They were collected from Twitter and are related to COVID-19. Sentences labeled sadness, fear and anger were chosen, to which sentences with the neutral class were added, drawn from a set based on the analysis of text polarity. Finally, the third set belongs to the data used in the article *Detecting Signs of Depression in Tweets in Spanish* [42]. It was created for the detection of signs of depression in Spanish tweets in order to study mental disorders in non-English texts.

The remaining emotions required the use of corpus in English, GoEmotions being one of the most relevant due to the number of labels it provides. It is a dataset created by Google, extracted from popular English subreddits and manually tagged with 27 emotion categories, including 12 positive, 11 negative, 4 ambiguous and 1 neutral emotion categories, providing the following emotions: Remorse, Disappointment, Nervousness, Embarrassment, Grief, Fear and Disgust. The next emotions extracted were Depressed, Hopeless, Lonely and Suicide. All of them coming from the paper *Detecting Depression in Social Media* [54]. As the name suggests, it was developed for the detection of signs of depression in social networks by extracting tweets through the Twitter API. While the last set used is "Life!' [38], a corpus created for research purposes on suicide and from which tweets labeled with the category Risk"were selected since they represented tweets with suicide risk.

The immediate step after obtaining all the corpora was to process them individually. This procedure involved different parts such as the selection of the classes of interest

in corpus that included emotions that had not been selected, the translation of those that were originally in English or a reduction in the size of certain sets that exceeded in number the instances of the classes of other sets, causing a very large imbalance between classes. Finally, all sets were merged all into a single set and cleaned up the texts in order to generate a new set with which to test the models and identify whether this process led to substantial improvements in the final performance of the models.

The option of translating texts was considered due to the lack of data in Spanish on categories such as hopeless, lonely or suicidal, so the option of using a professional translation tool such as SYSTRAN or a "normal"(a free automatic) translator such as Google Translator or DeepL was contemplated, allowing the translation of files. Although it can be considered that using this tool does not give great results and that, during the translation process, the original meaning of a sentence can be lost, causing the results to be not as good as with texts originally in the target language. Nowadays, automatic translators are able to identify the context and to comply with morphology and syntax, so when the original text is grammatically correct, the resulting translation can be as good as the one produced by a human translator. However, syntactic well-formedness does not equivalate semantic well-formedness, that is, a sentence or expression that makes sense. Thus, should both of these things not be the case and if there is a poor grammatical and semantic background, the results of a translation can be terrible. Therefore, for this project, different corpora were excluded after realizing that their texts contained a large number poorly formed and meaningless sentences, even for a human being. Only those corpora containing well-formed texts were used, so that the translation process provided good results and a minimum loss of the source context.

Once the translation process was completed, the number of categories from English texts was considerably higher compared to the rest of the categories. A selection process was therefore carried out with the aim of retrieving the texts that dealt with the author's own emotion in the tweet. In other words, if the tweet is labeled as loneliness, it should reflect the loneliness that the author himself may feel and not a general statement. In order to achieve this, a similar process was performed for the following categories: depressed, hopeless, loneliness and suicide. Articles such as Angela Leis et. al, Jianhong Luo et. al, and Kiritchenko et. al, provided lists of words and expressions that could suggest signs of depression, loneliness, or suicide respectively. These terms were classified into groups to reduce the emotion they represented, in order to obtain texts that at least contained one of these words, or one from each list, depending on the number of terms provided by the article. For example, the depressed class, which originally had 20,932 tweets, was reduced to 4,993. At the same time, the texts were focused on the actual user who wrote the text, since the original extraction process was based on the identification of users who could reflect these feelings and how they expressed them

in social networks. Once this process was finished, all the data was gathered into a single set, which resulted in a total of 40,806 tweets, which were then filtered in order to eliminate double tweets, special characters, line breaks, mentions, links, etc.

The completion of the emotion and corpus selection as well as processing tasks allowed further research in order to identify which were the most widespread techniques both in the development of articles and for Natural Language and emotion recognition tasks. BERT, RoBERTa or ELECTRA, all of them Transformer-based, were identified and studied in order to explain which improvements were implemented by these techniques, the differences between them and the concepts on which they are based. Three of them stand out: positional encodings, .ªttention.ªnd self-attention. The main role of positional encodings is to add to each word a number corresponding to the input order. Therefore, Transformers overcome the problem that RNNs (Recurrent Neural Networks) had when they required to sequentially process words in order to understand their order. Basically, a Transformer uses an encoder-decoder architecture. And within the encoder we find the Self-attention layer, which helps the encoder observe other words in the input sentence as it encodes a specific word. The generated outputs pass to a feed-forward neural network, which is automatically assigned to each position independently. As for the decoder, it contains the same two layers plus a new one in between (Encoder-Decoder Attention), which helps the decoder to focus on the relevant parts of the sentence, allowing the neural network to understand a word in the context of the surrounding words. Finally, attention is the concept of greatest importance. It was introduced in 2015 by researchers from the University of Montreal and Bremen. In fact, is a mechanism that allows a text model to "look.ªt each word in a sentence to decide the best possible way to translate it.

BERT is based on the Transfomers architecture but adds modifications such as the use of the encoder only, since its objective is to generate a language model. It features a pre-training and a finetuning process, and unlike directional models that read entry from right to left or left to right, it uses bidirectional processing. This allows a sentence to be parsed in two directions, so that the model can learn the context of a word based on all the words surrounding it in both directions. Regarding RoBERTa, the main changes were modifications in key hyperparameters and the byte-pair encoding, removing one of the main aspects of predicting the next sentence. RoBERTa uses larger batches and learning rates, replaces the static masking performed by BERT with dynamic masking and increases the training data. Finally, ELECTRA provides a new approach in training based on a generator and a discriminator. The generator's objective is to replace tokens in a sequence, so it is trained as a MLM (Masked Language Model). The discriminator, which is the discriminative model that ELECTRA provides, tries to identify which tokens were originally replaced by the sequence generator. The results provide a better understanding of the context learned by the model and exceed the results obtained by

other models such as BERT. The understanding of some of the concepts explained above, allowed us to obtain an overview of its architecture and to understand the reasons why such good results are obtained and its use is so widespread nowadays.

Based on these techniques, the next step was the selection of language models based on such techniques, namely Roberta-base, Roberta-base-bne, Bertin-roberta-base, BETO, RuPERTa, Robertuito, ELECTRA and Electricidad (Electra-esp). These models have been fine-tuned to meet the task on which they are to be evaluated and for which a similar set of hyperparameters has been selected for training. Batch size: 16, Learning rate: 5e-5, Number of epochs: 2, Weight decay: 0.01 and Maximum Sequence Length: 64. They were selected based on the characteristics of the machine to be tested and the recommendations suggested in the original BERT article, which described the fine-tuning procedure and a range of possible hyperparameter values that work well for most tasks, such as Batch size: 16, 32, Learning rate (Adam): 5e-5, 3e-5, 2e-5 and Number of epochs: 2, 3. The Batch size shall determine the total number of training samples in a single batch. The learning rate shall determine how much the model should change as a response to the estimated error every time its weights are updated. In this case, the training set will pass through the model twice, dictated by the number of epochs. Finally, to add a minor penalty to the loss function, the weigth decay was set to 0.1, and the maximum number of input tokens, represented by maximum sequence length, was set to 64, since the GPU did not have enough memory for a larger size.

Once the hyperparameters were established, the final step was the training of each model, a process for which the corpus created was divided into three subsets: Train, Validation (used for training), and Test (for the final tests). The results and the performance shown by each model revealed that, among them, Roberta-base-bne, Beto, Bertín and Robertuito stood out, since despite the penalty given by the classes that appear to a lesser extent in the Macro F1 metric, these models offer values between 61 and 69 percent. While in the Weighted F1 metric they offer values of up to 82 percent. The Macro F1 metric is the most relevant to analyze the performance obtained when there is no imbalance between classes and all classes are of equal importance. In this case, not all of them turn out to have the same importance since classes such as depressed, suicidal or loneliness have a higher classification interest and are expected to be accurately classified rather than classes such as grief, which appears to a lesser extent and is considered by some taxonomies as a subclass of sadness and could have been so disregarded. However, it was decided to include these types of classes that appear to a lesser extent and are mostly the ones that cause the penalty, in order to check how they performed and whether they were enough instances to provide a certain level of classification.

To conclude, it should be noted that the results that these models provide on the classes considered of greatest importance such as suicidal, loneliness, hopeless, depressed sadness, anger or neutral are between 80.5 and 98.5 percent for the Weighted F1 metric, with a very low level of classification error according to their confusion matrix. This means that the translation process carried out on most of these classes has not been an impediment for the correct performance of the models and the good classification of these classes. These results seem to be quite good and interesting as a first approach to the use of Transformer-based models for the classification of more complex emotions.

Capítulo 1

Introducción

El Procesamiento del Lenguaje Natural o PLN es un campo dentro de la Inteligencia Artificial, la Computación y la Lingüística, el cual ayuda a utilizar e interpretar el lenguaje humano a una máquina, proporcionándole de esta forma no solo la capacidad de leer un texto sino también la posibilidad de poder hablarlo o escucharlo. Esta gran evolución no se produjo de forma inmediata en el tiempo, razón por la cual esta primera parte del documento tiene como principal objetivo la puesta en contexto del tema elegido y los motivos que han impulsado su selección.

1.1. Contexto

El estudio del Procesamiento del Lenguaje Natural comienza a principios de 1900 cuando el profesor de lingüística suizo, *Ferdinand de Saussure*, desarrolló un enfoque que describía las lenguas como "sistemas" [67], es decir, sus elementos se relacionan entre sí, no están aislados. Argumentó que el significado se crea dentro del lenguaje, en las relaciones y diferencias entre sus partes. Y estableció diferencias claras entre lengua y habla, definiendo en primer lugar la lengua como social y el habla como individual. Para *Saussure* esto viene a decir que la sociedad es un sistema de normas sociales "compartidas" que proporciona condiciones para un pensamiento razonable, lo que resulta en decisiones que permiten el ejercicio del habla por parte de los individuos. Siendo esta, una visión todavía respetada y aplicada en los lenguajes informáticos actuales.

Sin embargo, su muerte en 1913 casi privó al mundo del concepto de "Lenguaje como ciencia", y fueron sus compañeros *Albert Sechehaye* y *Charles Bally* los que reconocieron la importancia de sus ideas y recopilaron toda la información necesaria para publicar en 1916 el libro *Cours de Linguistique Générale* [67], el cual sentó las bases y fue la inspiración para obras como la de *Levi-Strauss* y *Tristes trópicos*, comenzando de esta forma un movimiento intelectual conocido a día de hoy como Estructuralismo. Dicho movimiento se basa en la consideración de dualidades y afirma que se deben estudiar las lenguas en base a su realidad y no solo a su evolución, como había hecho hasta entonces la tradición historicista de la lingüística.

En 1952, *Alan Lloyd Hodgkin* y *Andrew Huxley* escribieron una serie de cinco artículos describiendo los experimentos que habían realizado para determinar las leyes que gobiernan el movimiento de los iones en una célula nerviosa durante un potencial de acción. El artículo final consagró el modelo *Hodgkin-Huxley*, el cual explicaba cómo el cerebro utiliza las neuronas en la formación de una red eléctrica. Estos sucesos ayudaron a impulsar la idea de la Inteligencia Artificial, la evolución de los ordenadores y del procesamiento del lenguaje natural. Siendo en 1954 cuando la Universidad de Georgetown junto con IBM realizaron el conocido *Experimiento Georgetown-IBM*, el cual implicó la traducción automática de oraciones del Ruso al Inglés. El sistema creado contaba con seis reglas gramaticales y 250 elementos léxicos en su vocabulario y a pesar de que su diccionario nunca se supo en su totalidad, estaba especializado en la química orgánica además de contar con temas generales.

Cabe destacar, que en el proceso seguido no hubo un análisis relacional o de la oración que pudiera reconocer su estructura, sino que fue un enfoque "lexicográfico" basado en dicho diccionario donde una palabra específica tenía una conexión con reglas y pasos clave, además de que las frases a traducir fueron cuidadosamente seleccionadas. Aun así, los resultados de las traducciones obtenidas se recibieron como un éxito y provocó la financiación de los gobiernos en el campo de la lingüística computacional, puesto que los autores afirmaban que en pocos años problemas como la traducción automática o MT (Machine Translation) serían un problema resuelto. Sin embargo, los avances que los autores habían afirmado que se producirían en tan breve periodo de tiempo realmente no se produjeron a ese ritmo, y más adelante se verá lo que esto supuso para el campo del MT y el PLN.

El siguiente hecho destacable ocurre en 1957 cuando *Noam Chomsky* publica su libro, *Estructuras sintácticas* (Syntactic Structures) [11]. El cual es considerado uno de los estudios más significativos del siglo XX debido a que revolucionó los conceptos lingüísticos anteriores. En esta breve monografía de unas cien páginas concluye que para que un computador entendiera un idioma, la estructura de la oración debería cambiarse. Afirmaba que existían conjuntos de reglas universales que facilitaban a nuestro cerebro

comprender el lenguaje además de permitirnos operar con el mismo desde que nacemos.

Es un año después de este acontecimiento, en 1958, cuando *John McCarthy* creó *LISP* (Locator/Identifier Separation Protocol). El segundo lenguaje de programación de alto nivel más antiguo, después de Fortran, todavía en uso. Fue desarrollado originalmente como una notación matemática para programas de ordenador, influenciado por la notación del lambda cálculo de *Alonzo Church*. Sin embargo, rápidamente se convirtió en uno de los lenguajes de programación favoritos para investigación en Inteligencia Artificial, siendo uno de los pioneros en incluir muchas ideas en ciencias de la computación como estructuras de datos de árboles, condicionales, recursividad o gestión del almacenamiento automático.

Años más tarde, en 1964, nace uno de los primeros programas informáticos de procesamiento del lenguaje natural que usó el enfoque comentado anteriormente basado en reglas. *ELIZA* fue un programa de comentario y respuesta mecanográfico creado por el Laboratorio de Inteligencia Artificial del MIT (Massachusetts Institute of Technology) por *Joseph Weizenbaum*. Quien lo diseñó con el objetivo de imitar a un humano o en su versión más famosa a un psiquiatra, tratando de demostrar la superficialidad de la comunicación entre humano y máquina. A pesar de no conversar con verdadero entendimiento, ELIZA es considerada como uno de los primeros chatbots y uno de los primeros programas capaces de intentar el test de Turing, con la importancia que ello conlleva.

Todos estos logros, a pesar de su gran aportación, no fueron considerados con la misma importancia para todas las personas. En 1966, los avances prometidos sobre el campo del MT comentados anteriormente, y que afirmaban que de dos a cinco años dejaría de ser un problema, resultaron en que tras diez años de investigación no se habían cumplido dichas expectativas y todavía quedaban muy lejos, ya que las traducciones automáticas seguían siendo mucho más caras que las traducciones humanas manuales. Este hecho sumado al informe de ALPAC (Automatic Language Processing Advisory Committee), donde dicho comité se encontraba muy escéptico de la investigación realizada hasta la fecha en traducción automática, y enfatizaba la necesidad de la investigación básica en lingüística computacional, provocó que las inversiones se redujeron severamente y que la investigación en Inteligencia Artificial y en Procesamiento del lenguaje natural, se parara casi por completo y fuera considerada por muchos un callejón sin salida.

No es hasta la década de 1980, catorce años después, cuando esta condición volvió a mejorar, ya que hasta entonces gran parte de los sistema de PLN estaban basados en conjuntos de reglas diseñadas a mano, siguiendo las teorías de *Chomsky*. La interrupción y casi abandono de la IA inició una nueva fase de ideas frescas y nuevas investigaciones, provocando que se dejaran a un lado conceptos anteriores de traducción automática o los sistemas expertos.

Siendo en 1990 cuando los modelos estadísticos y su uso para PLN aumentó sobremanera debido a la creación de Internet, el aumento de los textos online y la necesidad de procesamiento. El primer software de traducción automática que siguió esta vertiente basada en modelos probabilísticos fue *CANDIDE*, creada en 1991 por investigadores del Thomas J. Watson Center de IBM en Yorktown Heights (Nueva York). Traducía texto de francés a inglés y podía utilizarse como un programa totalmente automático o como un asistente de traductor.

La década de los 2000 y los años posteriores, han traído gran cantidad de trabajos nuevos e innovaciones. En 2003, *Yoshuo Bengio* y su equipo de investigación propusieron el primer modelo de lenguaje neuronal, utilizando una red neuronal prealimentada (Feed-forward) [5]. La cual difiere de las redes neuronales recurrentes en que las conexiones entre las unidades no forman un ciclo, los datos se mueven en una única dirección desde los nodos de entrada a través de los nodos ocultos (si los hay) y luego hacia los nodos de salida. En 2011 Apple lanza uno de los primeros asistentes de voz exitoso, *Siri*. En 2013 se produjo un gran avance en el campo del PLN con la técnica *Word2Vec*, que permitió la creación de vectores de palabras de forma mas eficiente, reduciendo el tiempo de cómputo y mejorando el rendimiento. En 2014 y 2015 las redes neuronales recurrentes (RNN) y las redes Long Short Term Memory (LSTM), una versión de las RNN, aumentaron en popularidad. Siendo a su vez en 2015 y 2016 cuando el concepto de "atención", técnica que intenta imitar la percepción cognitiva humana, comienza a coger fuerza y al ser utilizada en las redes neuronales permitía a la red centrarse en un subconjunto específico de datos. Estos modelos llegaron a batir records de rendimiento en muchas tareas de PLN, como la traducción automática o la respuesta de preguntas.

Por último, dentro del campo del PLN, el reconocimiento de emociones se ha desarrollado como un área de investigación importante con la capacidad de proporcionar gran información. No obstante, obtener la emoción tras un texto, imagen o el habla, es un proceso complejo. Y es por esto que las investigaciones actuales se centran en encontrar formas para abordar de manera más eficiente o con mayor precisión esta tarea. Como el artículo mostrado por Seunghyun Yoon et al. [84], en el que proponen un modelo de codificador recurrente dual basado en deep learning, que utiliza datos de texto y señales de audio simultáneamente para obtener un mejor entendimiento de los datos del habla.

Tanto los avances actuales como futuros, pueden suponer una mejora sustancial en el reconocimiento de emociones y con ello un perfeccionamiento para las computadoras a la hora de tomar decisiones y seleccionar la que mejor convenga al usuario. Consiguiendo a su vez una interacción más natural entre humano y máquina.

1.2. Motivación

A pesar de que el análisis de emociones sea un tema tratado y estudiado con frecuencia y del cual hay gran cantidad de trabajos. Pocos de ellos están centrados en nuestro idioma, y gran parte de estos se centran solamente en el uso de las siete emociones básicas de *Paul Ekman* [21]. Como son la tristeza, la ira, la sorpresa, el miedo, el desprecio, la alegría y el asco. Sin embargo, la importancia que tiene en estos momentos la salud mental y estados relacionados con ella, como son, la depresión, la soledad o la desesperanza, suponen algunos de los principales motivos que han impulsado este cambio y la selección de emociones ligadas a este tema.

Habiéndose tratado años atrás como un tema tabú debido a la estigmatización producida, en la actualidad la salud mental se está convirtiendo en un tema cada vez más importante para la sociedad española. Según el INE (*Instituto Nacional de Estadística*) [24], y la última Encuesta Europea de salud mental realizada en España [66], entre Julio de 2019 y Julio de 2020 se observó un incremento en estados de decaimiento y depresión en la población debido a la pandemia. Siendo en 2020, año en el cual el 5,4 % de la población (2,1 millones de personas) sufrió algún cuadro depresivo. A esto se le añade la relación de la pandemia con el aumento de suicidios. Donde estudios como el realizado por el CIBERSAM (*Centro de Investigación Biomédica en Red de Salud Mental*) [12], concluye que la tasa de suicidio ascendió del 8,3 por cada 100.000 habitantes de 2019 al 8,9 por cada 100.000 habitantes en 2020. Asimismo organizaciones como la OMS defienden que "*la sensación de aislamiento puede generar conductas suicidas*".

Por otro lado Twitter, con 4,2 millones de usuarios en la actualidad, se ha convertido en una de las redes sociales más utilizadas en nuestro país y por consiguiente, muchos de sus usuarios pueden llegar a reflejar sus estados de ánimo a través de ella. Esto deriva en una gran cantidad de información generada, de interés y que puede ser utilizada en el campo del PLN con fines éticos.

A lo comentado anteriormente se suma la atracción por la Inteligencia Artificial, dentro de esta área por el Procesamiento del Lenguaje Natural y por el NLU (Natural Language Understanding). Áreas como la búsqueda de respuestas, la de resumen automático o el análisis de sentimientos, han sufrido una evolución y mejora en los últimos años. Además, el hecho de que las aplicaciones dadas a algunas de estas áreas sean de gran importancia y tengan una utilidad real en la actualidad, pudiendo llegar a provocar una mejora en la vida de las personas, como sería una mejor y pronta identificación de estos estados preocupantes. Supondría una posible forma de abordarlos antes de que escalen, lo que resulta tener un valor añadido y hace que su estudio se sienta como una aportación útil y de interés.

Capítulo 2

Estado del arte

Distintas arquitecturas se han ido desarrollando en el campo del PLN a lo largo de estos últimos años. Entre todas ellas destacan los Transformers. Precursora de subcategorías dentro de la misma como los Autoencoding Transformers, Autoregressive Transformers o Rendezvous. No ha sido otra cosa, más que el éxito de las mismas, el que nos ha llevado al estado actual del campo y a la era de BERT [17], XLNet [83], ERNIE 2.0 [72], T5 [59], GPT-3 [7] o RoBERTa [43], como podemos ver en la Figura 2.1.

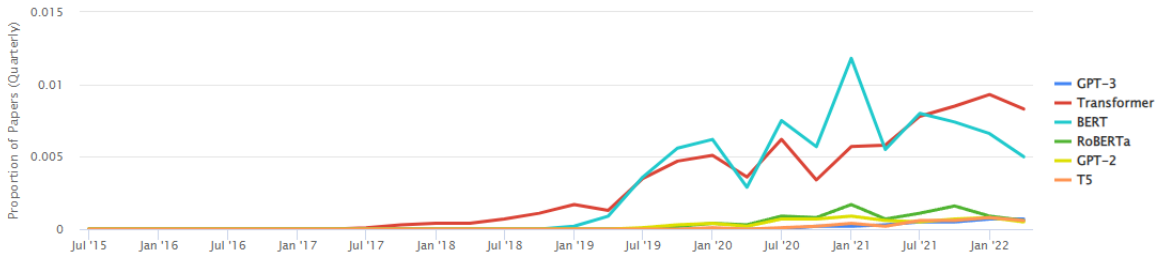


Figura 2.1: Popularidad en el tiempo. [14]

Un Transformer es un tipo de arquitectura de red nueronal. Las redes neuronales son un modelo muy efectivo para analizar datos complejos como video, audio, imágenes y texto. Existiendo diferentes tipos de redes optimizadas dependiendo del tipo de tarea. Por ejemplo, para analizar imágenes normalmente utilizaremos una Red Neuronal Convolucional o CNN (Convolutional Neural Network). Este tipo de redes ha tenido gran éxito en problemas de visión artificial, como reconocimiento de caras, identificación de objetos o leer dígitos escritos a mano.

Sin embargo, no existió nada con resultados igual de buenos para tareas lingüísticas como traducción, generación o resumen de texto. La forma de utilizar el deep learning para comprender el texto era con un tipo de modelo llamado Red Neuronal Recurrente o RNN (Recurrent Neural Network), mostrada a la izquierda en la Figura 2.2. Cuya principal diferencia con una Red Neuronal Pre-alimentada o Feedforward, mostrada a su derecha, es que incluye ciclos. Permitiéndole obtener información de entradas anteriores para influir en la entrada y salida actual.

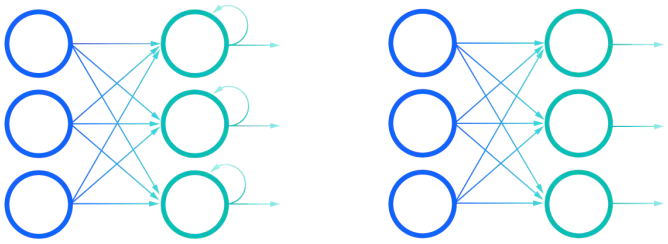


Figura 2.2: Red Neuronal Recurrente vs Red Neuronal Pre-alimentada. [20]

Si quisiéramos traducir una oración de inglés a francés. Una RNN tomaría como entrada una oración en ingles, procesaría las palabras una a una y luego, secuencialmente, devolvería su equivalente en francés. El hecho de ser secuencial tiene un inconveniente ya que en el lenguaje el orden de las palabras importa, por tanto cualquier modelo que vaya a entender el lenguaje debe capturar y tener en cuenta el orden de las palabras. A la hora de procesar grandes textos, las RNN tenían un problema y es que al llegar al final de un párrafo podían no recordar lo que había sucedido al principio. Con lo que, por ejemplo, podía olvidar el género del sujeto que se estaba procesando durante párrafo largo. Otro de sus mayores inconvenientes fue lo difíciles que eran de entrenar. Eran muy susceptibles al problema de "desaparición y explosión de gradiente". El gradiente de desaparición conduce a una convergencia lenta y el gradiente explosivo lleva a un cambio excesivo en los pesos, indeseable en la mayoría de casos. Debido a este problema, las redes neuronales no convergían, lo que derivaba en un reinicio del entrenamiento, suponiendo un gasto de tiempo y recursos. Además el procesamiento de las palabras secuencialmente hacía que las RNN fueran muy difíciles de paralelizar, no permitiendo acelerar el entrenamiento aunque se les proporcionara más GPU.

Esto cambió con la llegada de los Transformers en 2017 [73]. Desarrollados por investigadores de Google y la Universidad de Toronto, inicialmente para traducción, la principial diferencia con las RNN es la capacidad de ser paralelizados de forma muy eficiente. Llegando a ser capaces de entrenar modelos muy grandes. Lo cual quedó demostrado con el desarrollo de GPT-3 (Generative Pre-trained Transformer 3) [7], un tipo especial de Transformer (Autoregressive Transformers). Considerado uno de los chatbots más potentes y mejor desarrollados hasta la fecha, fue entrenado con cerca de 43 TB de datos de texto admitiendo 175 mil millones de parámetros de aprendizaje automático, como se puede apreciar en la Figura 2.3.

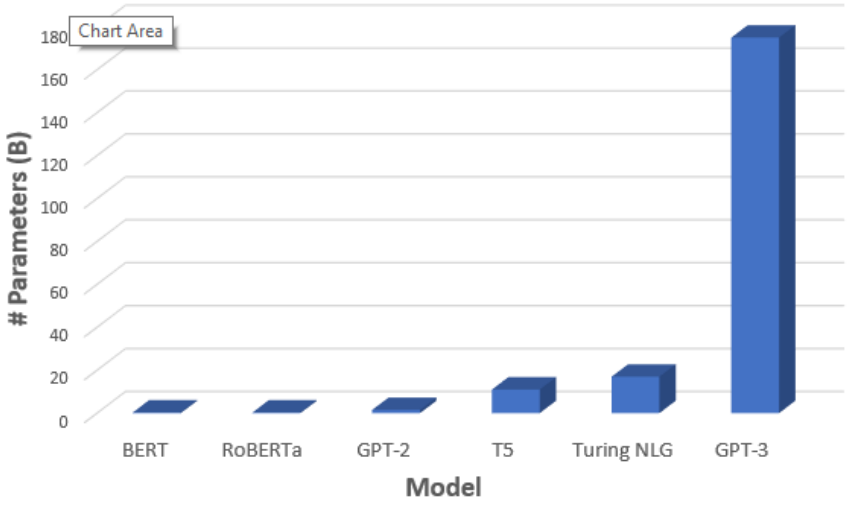


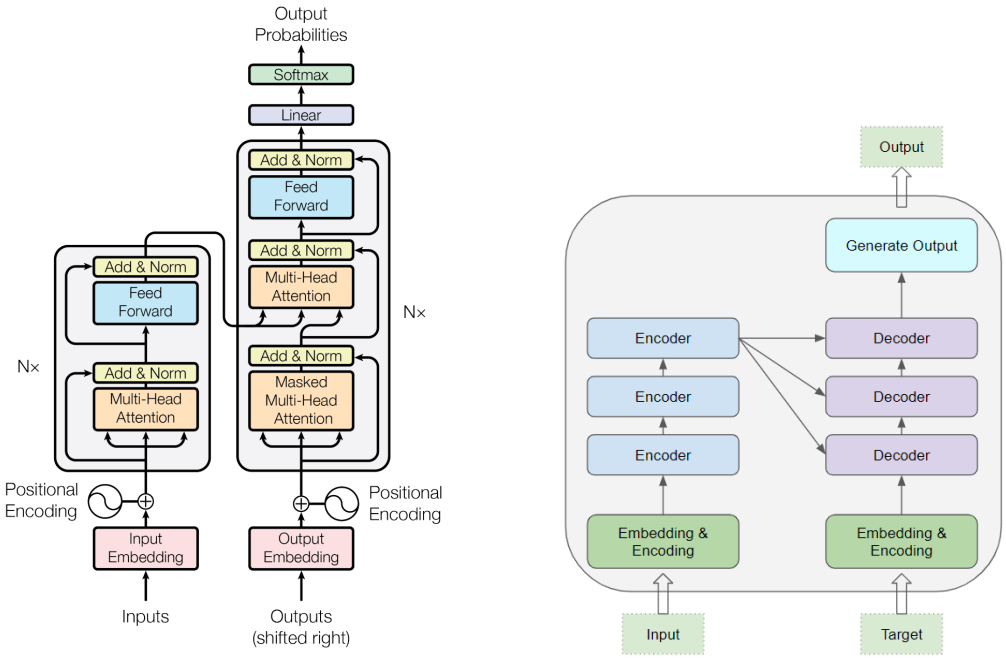
Figura 2.3: GPT-3 Parámetros entrenables [64]

2.1. Transformers

Por lo general, los Transformers presentan una combinación de mecanismos de atención de "múltiples cabezas", conexiones residuales, normalización de capas, conexiones de avance e incrustaciones posicionales. Aunque, simplificando, podemos decir que la principal innovación detrás de ellos se basa en tres conceptos principales en los que nos centremos a continuación para explicar su funcionamiento:

- **Codificaciones posicionales** (Positional Encodings).
- **Atención** (Attention).
- **Autoatención** (Self-Attention).

Puesto que la arquitectura mostrada por los investigadores en el artículo original (Figura 2.4 (a)) consta de diferentes partes, muy aglutinadas entre sí y pueden no resultar visualmente entendibles. Nos ayudaremos en la Figura 2.4 (b) para una descripción a más alto nivel, ya que agrupa conceptos que facilitan la explicación y el tratamiento de las diferentes partes que componen la arquitectura.



(a) Artículo Original. [73] (b) Simplificada [18]

Figura 2.4: Arquitectura Transformers.

El primero de los tres conceptos con mayor importancia comentados anteriormente es el de la codificación posicional. Cuya función es agregar a cada palabra un numero correspondiente al orden de entrada. Con lo que tendríamos para una oración de entrada como “Thinking Machines”, un resultado similar a: [(”Thinking”, 1), (”Machines”, 2)].

Con esto, los Transformers consiguen superar el problema con el que contaban las RNN, a la hora de necesitar procesar las palabras secuencialmente para entender el orden de las mismas. Aunque en un primer momento el Transformer no sabe como interpretar estas codificaciones posicionales, a medida que entrena y ve más ejemplos de oraciones y sus codificaciones, aprende a utilizarlos de manera correcta.

Esencialmente, un Transformer contiene una secuencia de codificadores y otra de decodificadores. Todos los codificadores son idénticos el uno al otro, y lo mismo ocurre con los decodificadores. Sin embargo, no comparten pesos. Como vemos en la Figura 2.5, cuentan con la siguiente arquitectura de subcapas, dividiéndose en dos para los codificadores y tres para los decodificadores.

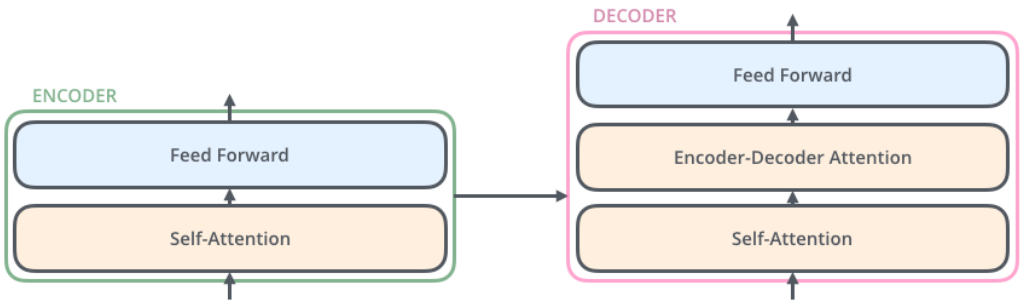


Figura 2.5: Arquitectura interna Encoder y Decoder. [2]

El codificador contiene uno de los principales conceptos de los Transformers comentados inicialemente. La capa de autoatención (Self-Attention). Dicha capa ayuda al codificador a mirar otras palabras en la oración de entrada a medida que codifica una palabra específica. Las salidas generadas, pasan a una red neuronal pre-alimentada (Feed-forward), la cual es aplicada a cada posición independientemente. En cuanto al decodificador, contiene las mismas dos capas, pero añade una nueva entre ambas (Encoder-Decoder Attention), que ayuda al decodificador a centrarse en las partes relevantes de la oración.

Cada palabra de entrada se convierte en un vector usando un algoritmo de embedding, como vemos en la Figura 2.6. Esto solo ocurre para la entrada recibida por el primer codificador, ya que lo general para todos es recibir una lista de vectores, de tamaño 512, procedente de la salida del codificador anterior.

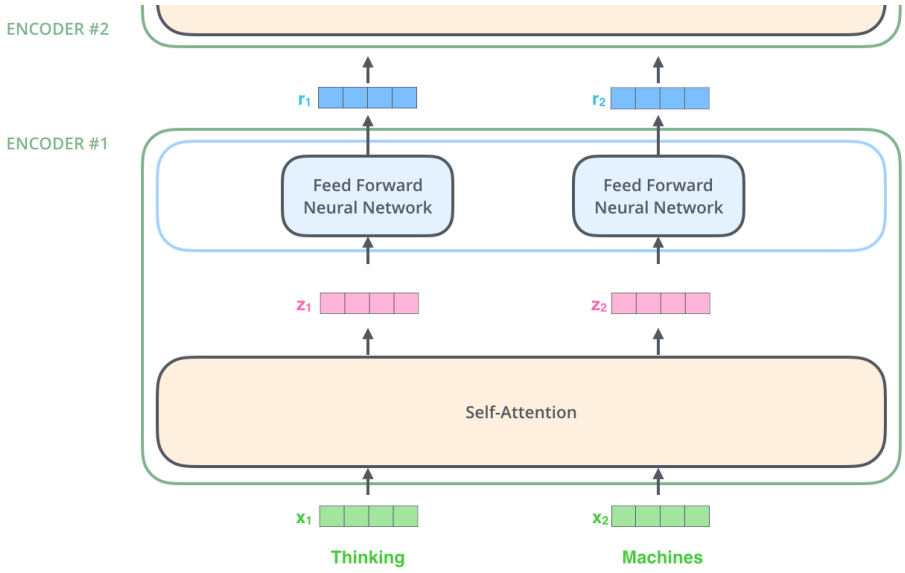


Figura 2.6: Proceso interno del Encoder. [2]

En este punto vemos un aspecto clave de los Transformers, y es que cada palabra fluye a través de su propio camino en el codificador. Existe dependencia entre estos caminos en la capa de Self-Attention, pero no la hay en la capa de Feed-forward, por tanto, los diferentes caminos se pueden ejecutar en paralelo mientras se sigue a través de la capa de Feed-forward.

El segundo concepto de importancia es la Atención, introducida en 2015 por investigadores de la Universidad de Montreal y Bremen [19]. Este mecanismo permite que un modelo de texto "mire" cada palabra en una oración para decidir la mejor forma posible de traducirla en la oración de salida. Podemos ver este concepto reflejado en la Figura 2.7, perteneciente al artículo original que presentó dicho concepto, además de aportar la siguiente frase como ejemplo: "The agreement on the European Economic Area was signed in August 1992." es traducida como "L'accord sur la zone économique européenne a été signé en août 1992.".

Vemos como en su traducción, el modelo se centra en la palabra más acorde y en algunos casos como para "la" se centra tanto en "the" como en "Area". Esto tiene sentido ya que en francés las palabras tienen género y el modelo debe saber cuál es el que mejor le corresponde a cada una. De ahí que centre su atención en otras que le ayuden a tomar la decisión, puesto que una traducción palabra a palabra desembocaría en errores. De igual forme ocurre a la hora de traducir "européenne", el modelo se centra en "European" y "Economic", pero en este caso debido a que en francés algunas palabras cambian de orden. Lo cual, el modelo, también debe tener en cuenta.

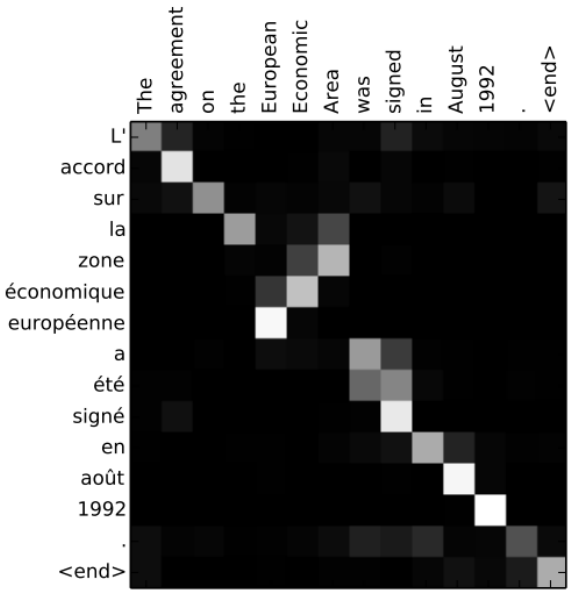


Figura 2.7: Attention heatmap. [19]

El hecho de saber a qué palabras "mirar" se desconoce inicialmente. Conforme el modelo entrena y ve miles y miles de ejemplos de traducción de un idioma a otro, aprende las reglas gramaticales, cómo respetar el género, el orden correcto o la pluralidad.

En cuanto a la capa de Self-Attention, último concepto de los tres principales. Es el que permite a la red neuronal entender una palabra en el contexto de las palabras que la rodean. Por ejemplo, en la frase: "The animal didn't cross the street because it was too tired", un humano sabe que "it" hace referencia a "animal", pero esto no es tan simple para un algoritmo.

Mediante esta capa, el modelo puede mirar otras posiciones en la secuencia de entrada en busca de información que le permita codificar una palabra de la mejor forma. Siendo este su mecanismo para asociar “it” con “animal”, como se refleja en la Figura 2.8.

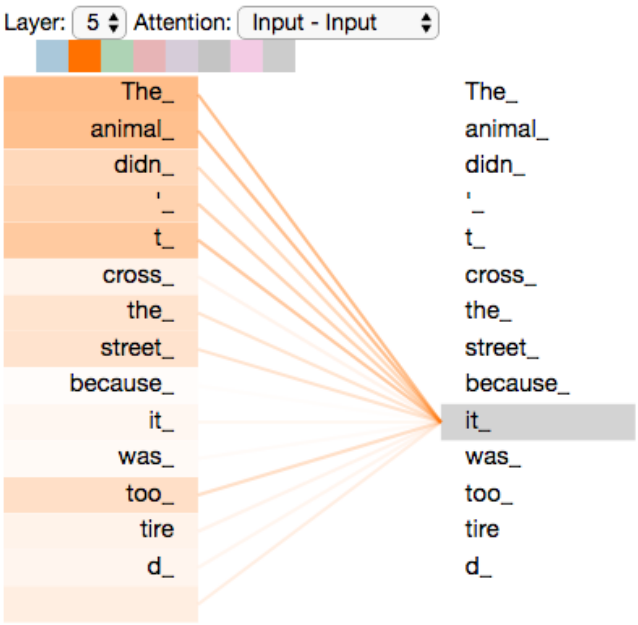
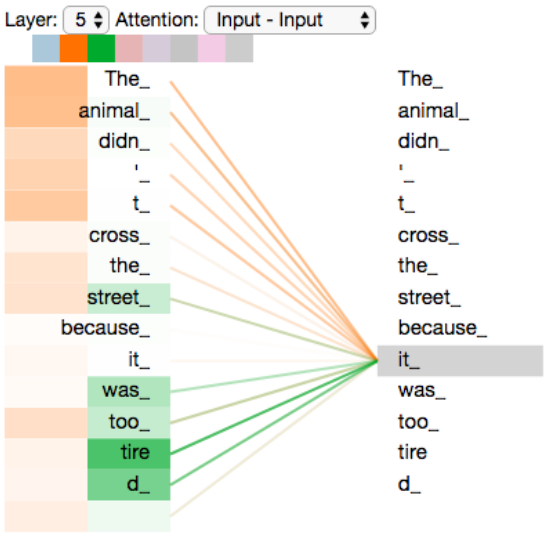
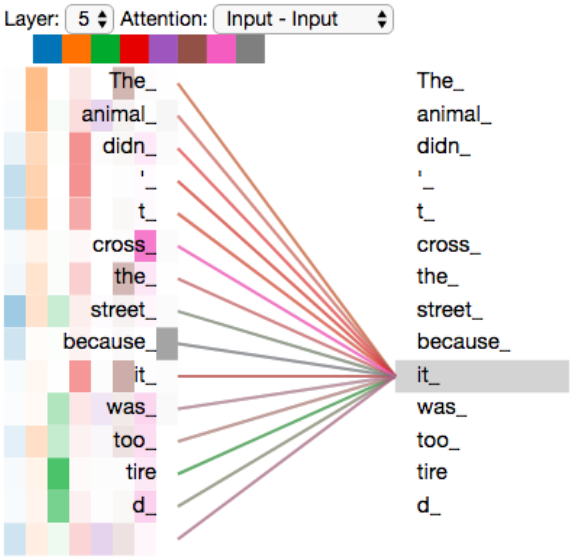


Figura 2.8: Self-Attention. [2]

El documento original refinó todavía más la capa de Self-Attention añadiendo un mecanismo de ”multiples cabezas”, como vemos en la Figura 2.9. El cual mejoró el rendimiento debido a que expandía la capacidad del modelo de centrarse en diferentes posiciones. Permitiendo de esta forma múltiples “subespacios de representación”, lo que deriva en un mejor entendimiento del contexto por parte del modelo.



(a) Dos cabezales. [2]



(b) Todos los cabezales. [2]

Figura 2.9: Self-Attention con múltiples cabezales.

2.1.1. BERT

Abreviatura de Bidirectional Encoder Representation from Transformers, BERT, es un modelo de machine learning para procesamiento de lenguaje natural. Su artículo original [17], fue publicado en 2018 por investigadores de Google sirviendo como solución a más de 11 tareas lingüísticas. Dentro de las que se encuentra el análisis de sentimientos, la generación de texto o la respuesta a preguntas. Los buenos resultados obtenidos en esta gran variedad de tareas junto con las innovaciones aportadas, provocó gran revuelo en el mundo del PLN.

BERT hace uso de la técnica de Transformers comentada anteriormente, la cual utilizaba entre otras cosas un codificador, y un decodificador. Sin embargo, BERT solo utiliza el codificador puesto que su objetivo es generar un modelo del lenguaje. Además, la innovación clave de BERT, al contrario de los modelos direccionales que leen la entrada de derecha a izquierda o de izquierda a derecha, fue utilizar un entrenamiento bidireccional. Esto permite analizar una oración en dos direcciones, consiguiendo con ello que el modelo aprenda el contexto de una palabra en función de todas las palabras que la rodean en ambas direcciones. BERT a su vez cuenta con dos procesos, como se puede ver en la Figura 2.10, un pre-entrenamiento donde el modelo se entrena en diferentes tareas con datos no etiquetados, y un proceso de finetuning, donde primero se inicializa el modelo con los parámetros previamente entrenados y todos los parámetros se ajustan utilizando datos etiquetados de las tareas posteriores.

Figura 2.10: Proceso de training y finetuning en BERT . [31]

El proceso de pre-entrenamiento puede llevar incluso días, debido a que pueden necesitarse miles y miles de muestras de entrenamiento. Sin embargo, una vez terminado este proceso, podemos usar los pesos previamente entrenados como pesos iniciales para un proceso de entrenamiento de ajuste fino (Fine-Tuning) específico de la tarea que deseamos realizar.

Esto simplifica y reduce el proceso de entrenamiento puesto que la necesidad de datos etiquetados es mucho menor, obteniendo como resultado un modelo ajustado a la tarea deseada y que por lo general da buenos resultados.

A la hora de entrenar utiliza dos estrategias de entrenamiento para no limitar el aprendizaje, como ocurre con modelos con un enfoque direccional que intentan predecir la siguiente palabra en una secuencia.

La primera de las estrategias es hacer uso de un modelo de lenguaje enmascarado o MLM (Masked Language Model). Con el cual antes de introducir la secuencias en BERT, un porcentaje variable de las palabras de cada secuencia se enmascaran reemplazándose con un token [MASK], como vemos en la Figura 2.11. Para más tarde intentar predecir la palabra original que se encontraba en esa posición, según el contexto proporcionado por las palabras no enmascaradas de la secuencia. Esta predicción requiere una capa de clasificación (Classification Layer) a continuación de las salidas del codificador (Outputs), multiplicar los vectores de salida por la matriz de incrustación para transformarlos en la dimensión del vocabulario y un cálculo de la probabilidad de cada palabra en dicho vocabulario.

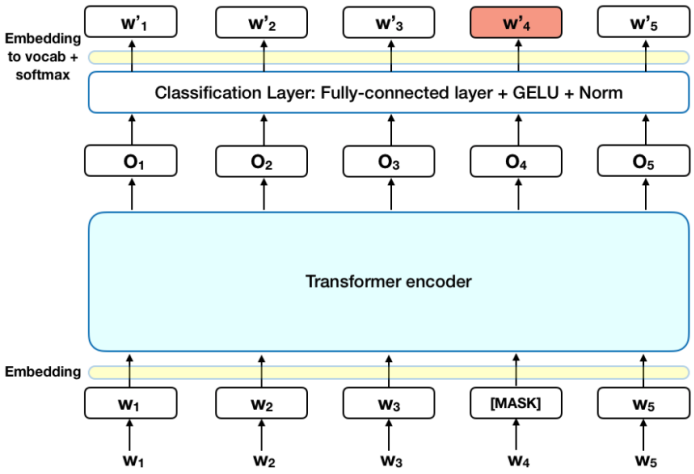


Figura 2.11: Modelo de lenguaje enmascarado. [35]

Debido a que la función de pérdida de BERT solo tiene en cuenta la predicción de los valores enmascarados y no las predicciones de las palabras no enmascaradas, se produce el inconveniente de que el modelo converge de forma más lenta en comparación con modelos direccionales. Aun así, esto se compensa debido a su mejor conocimiento del contexto y sus resultados.

En cuanto a la segunda de las estrategias utilizadas, se basa en la predicción de la siguiente oración o NSP (Next Sentence Prediction). El modelo recibe, mientras entrena, pares de oraciones como entrada y aprende a predecir si la segunda oración del par es la siguiente oración en el documento original. El 50 % aproximadamente de las entradas, son un par en el que la segunda frase es realmente la siguiente del documento, mientras que para el otro 50 % de pares restantes, se selecciona una frase aleatoria como segunda frase.

Para ayudar al modelo en el proceso de distinción entre dos oraciones, mientras entrena, se sigue un mecanismo similar al mostrado en la Figura 2.12. Inicialmente, la capa Positional Embedding agrega a cada token la posición que representa en la secuencia (incrustación posicional), uno de los mecanismos utilizados también por la técnica de Transformer. Seguidamente, la capa de Sentence Embedding añade al token un nuevo identificador que representa la oración (A o B), similar a lo anterior pero con un vocabulario diferente. Finalmente, la capa Token Embeddings inserta un token ([CLS]) al comienzo de la primera oración y otro token ([SEP]) al final de cada oración, obteniendo como resultado la secuencia de entrada.

Para decidir si la segunda oración está realmente conectada con la primera, de manera simplificada, toda la secuencia de entrada pasa por el modelo, la salida del token [CLS] se transforma a un vector utilizando una capa de clasificación que hace uso de matrices de pesos y sesgos, y un cálculo de probabilidad determina que realmente esa entrada es la siguiente oración.

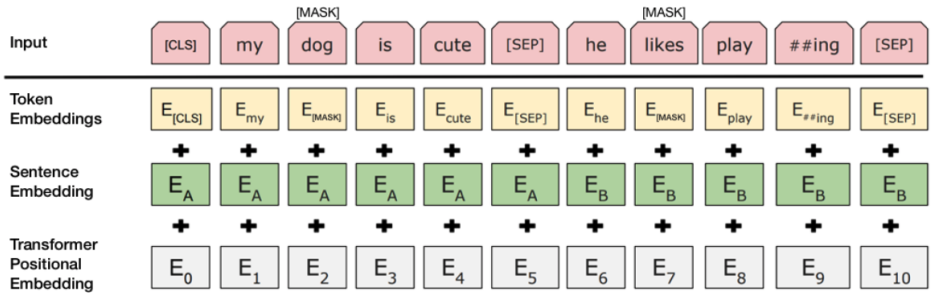


Figura 2.12: Predicción de la siguiente oración. [35]

Todas estas mejoras, cambios y nuevos mecanismos utilizados, son los que hicieron que BERT diera sorprendentes resultados al evaluarlo en diferentes tareas del lenguaje. De entre las que destacan el análisis de sentimientos. Siendo estos algunos de los motivos por los que ha día de hoy gran cantidad de modelos siguen estando basados en su arquitectura o tratan de mejorarla.

2.1.2. RoBERTa

Presentado en 2019 , RoBERTa [43], se basa en el modelo BERT de Google. Modifica hiperparámetros clave y la codificación de pares de bytes, elimina uno de los aspectos clave comentados anteriormente el el punto 2.1.1 sobre predecir la siguiente oración, y entrena utilizando batchs y tasas de aprendizaje (learning rates) más grandes.

Experimentaron sustituyendo el enmascaramiento estático realizado por BERT, donde el enmascaramiento se realizaba una única vez durante el preprocesamiento de los datos, por un enmascaramiento dinámico. En el cual, se duplican y enmascaran los datos de entrenamiento 10 veces, cada una de estas veces con una estrategia de enmascaramiento diferente durante 40 epochs con 4 de estos epochs con la misma máscara. De esta forma se busca evitar enmascarar la misma palabra varias veces. A su vez, también realizaron pruebas agregando/eliminando en varias versiones, la pérdida producida al predecir la siguiente oración, y concluyeron que eliminar esta pérdida mejoraba ligeramente el rendimiento en tareas posteriores. Por otro lado, BERT originalmente entrenaba con 1 millón de steps y un tamaño de batch de 256 secuencias. En el artículo presentado, los autores entrenaron el modelo tanto con 125 steps y un tamaño de batch de 2000, como con 31.000 steps y tamaño de batch 8.000. Este aumento en el tamaño del batch, proporcionaba dos ventajas principalmente. Una de ellas es que mejoraba el enmascaramiento producido al utilizar un modelo del lenguaje enmascarado (MLM), además de la precisión final obtenida. Siendo de esta forma, más fácilmente paralelizables.

	MNLI	**QNLI**	**QQP**	**RTE**	**SST**	**MRPC**	**CoLA**	**STS**	**WNLI**	**Avg**
Single-task single models on dev										
$\text{BERT}_{\text{LARGE}}$	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-	-
$\text{XLNet}_{\text{LARGE}}$	89.8/-	93.9	91.8	83.8	95.6	89.2	63.6	91.8	-	-
RoBERTa	**90.2/90.2**	**94.7**	**92.2**	**86.6**	**96.4**	**90.9**	**68.0**	**92.4**	**91.3**	-
Ensembles on test (from leaderboard as of July 25, 2019)										
ALICE	88.2/87.9	95.7	**90.7**	83.5	95.2	92.6	**68.6**	91.1	80.8	86.3
MT-DNN	87.9/87.4	96.0	89.9	86.3	96.5	92.7	68.4	91.1	89.0	87.6
XLNet	90.2/89.8	98.6	90.3	86.3	**96.8**	**93.0**	67.8	91.6	**90.4**	88.4
RoBERTa	**90.8/90.2**	**98.9**	90.2	**88.2**	96.7	92.3	67.8	**92.2**	89.0	**88.5**

Figura 2.13: RoBERTa vs BERT. [43]

Como podemos ver en la Figura 2.13, Roberta supera a BERT notablemente. Se observó que incrementar los datos de entrenamiento mejoraba el rendimiento en gran medida, por lo que se incrementó a 160GB de texto sin comprimir. Utilizándose diferentes conjuntos de datos preparados para diferentes tareas como el análisis de sentimientos, en la que se enmarca el corpus SST (Stanford Sentiment Treebank) [55].

2.1.3. ELECTRA

El modelo ELECTRA fue propuesto en 2020 en el artículo *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators* [13], por investigadores de Google AI. En el cual se criticaban ciertos aspectos negativos de métodos de pre-entrenamiento MLM como BERT, también comentadas anteriormente en el punto 2.1.1, como es la lenta convergencia del mismo. Ya que reemplazar tokens de la entrada como [MASK] y luego entrenar para reconstruirlos requiere grandes cantidades de cómputo para llegar a ser efectivo.

Como alternativa, mostraron un nuevo enfoque de pre-entrenamiento que entrena dos modelos de Transformers: el generador y el discriminador, mostrados en la Figura 2.14. El objetivo del generador es reemplazar tokens en una secuencia, de forma similar al proceso de enmascaramiento visto anteriormente, por lo que está entrenado como un MLM. El discriminador, que es el modelo discriminativo que ELECTRA aporta, trata de identificar qué tokens fueron reemplazados originalmente por el generador de la secuencia.

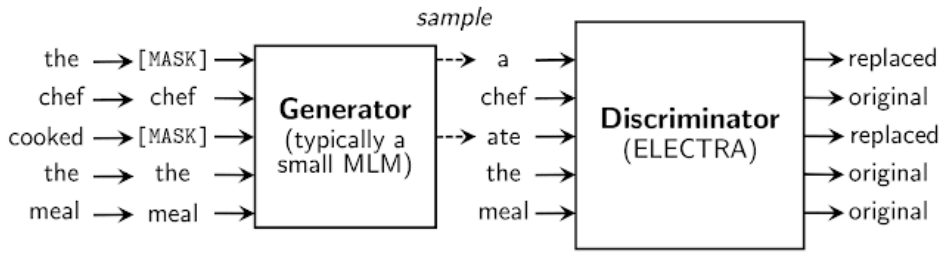


Figura 2.14: Proceso general de detección de tokens reemplazados. [13]

Los resultados demostraron que este nuevo mecanismo incorporado al pre-entrenamiento es más eficiente que un MLM ya que la tarea se define sobre todos los tokens de la secuencia de entrada que se enmascararon y no solamente sobre un pequeño subconjunto. Lo que da lugar a una mejor representación y entendimiento del contexto que aprende el modelo, llegando a superar incluso los resultados obtenidos por BERT, como vemos en la Figura 2.15.

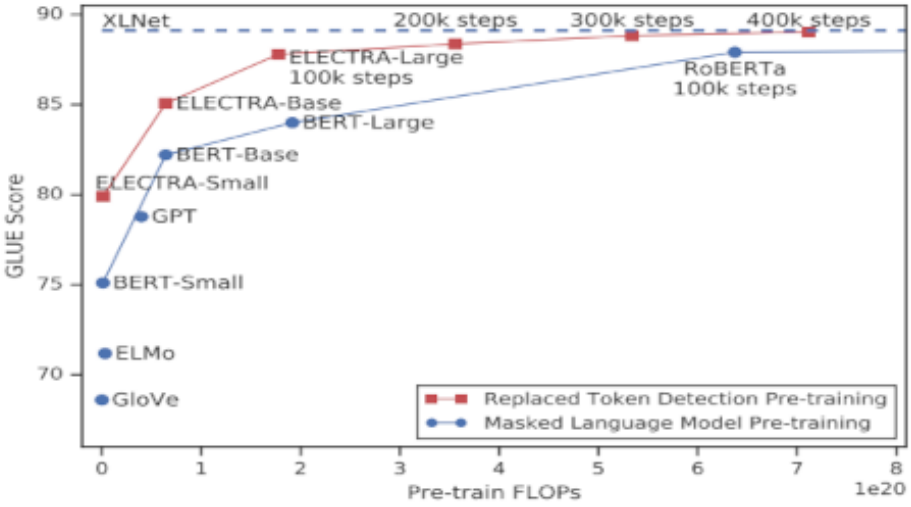


Figura 2.15: Redimiento ELECTRA. [13]

2.2. Procesamiento del Lenguaje Natural y Clasificación de texto

El Procesamiento de Lenguaje Natural es el encargado en la actualidad del estudio del lenguaje humano, con el objetivo de proporcionar herramientas para una mejor compresión del mismo. Dentro de este, encontramos el campo de clasificación de texto, cuya finalidad es asignar a una oración o documento una o varias categorías, las cuales varían dependiendo del contexto proporcionado por los datos. En este campo encontramos categorías como la clasificación de noticias, de sentimientos y de emociones, la detección de odio y la ironía, o evaluar la coherencia de los textos, entre otras. Motivos como la gran cantidad de noticias nuevas que se publican cada día o el incremento de las compras online, provoca la necesidad de identificar si una noticia es real o fake, conocer las opiniones de los clientes por parte de las empresas y lo que piensan de sus productos para saber qué ofrecer o cómo mejorar.

En el caso de las redes sociales, el incremento de su uso y que ha día de hoy gran cantidad de personas las utilicen para propagar discursos de odio, de diferente tipo, ha provocado que el interés por sistemas de detección de Hate-Speech crezca como mecanismo para erradicar estos mensajes y que este tipo de personas se detecten rápidamente. Estos sistemas de clasificación están basados principalmente en técnicas de aprendizaje automático, partiendo generalmente de recursos anotados por expertos en el campo. Algunos de estos recursos podrían ser Haternet [56], que analiza el odio en la red social Twitter. Así como corpus en nuestro idioma como MisoCopus [28] o AMI (Automatic Misogyny Identification) [25], ambos centrados en la detección de la misoginia en textos.

Debido a motivos similares al caso anterior, en cuanto al uso de las redes sociales, y retornando al tema en el que se enmarca este trabajo. La clasificación de emociones y la identificación del estado de ánimo de una persona en un determinado momento, también cobra más interés. Puesto que puede llevar desde una mejor recomendación de contenido online, hasta conocer el estado de salud actual de una persona. Para esta tarea, los diferentes enfoques en los que se enmarcan los distintos trabajos de reconocimiento de emociones suelen estar basados en:

- **Palabras clave** (Keywords):

  Esta perspectiva trata de explotar las estructuras de las oraciones mediante el conocimiento de las características clave de las palabras combinado con etiquetas de emociones. Utilizando lexicones como Word-Net Affect y SentiwordNet, y aplicando reglas lingüísticas.

Trabajos como el de Rahman et.al [22], utilizan este enfoque de análisis de palabras clave junto con un análisis de emoticonos, negación de palabras clave, y palabras cortas, entre otros. Para la creación de un conjunto de 25 clases para de detección de emociones en oraciones.

- **Corpus**:

  Este enfoque utiliza el aprendizaje supervisado y corpus clasificados con un conjunto de emociones extraídas de diferentes taxonomías para la identificación de lexicones de palabras y emociones. De esta forma es posible obtener las tendencias sintácticas y semánticas de un texto. En base a este enfoque, destaca el trabajo de Plaza del arco, et al. [52], en el se realizó un análisis de emociones, utilizando distintos lexicones de sentimientos como EmoLEX [47] y SEL [68], entre otros. Así como el trabajo de Rachman et al. [58]. En el que se desarrolló el corpus CBE (Corpus Based Emotion) mediante el corpus WNA (Wordnet Affect Emotions) [70] y ANEW (the Affective Norms for English Word) [6]. Demostrando que el uso del corpus CBE mejoraba el rendimiento de detección de emociones.

- **Machine Learning y Deep Learning:**

  El aprendizaje automático se puede enmarcar en técnicas de aprendizaje supervisado y no supervisado, con técnicas como SVM (Support Vector Machines) o K-means respectivamente. El machine learning clásico depende más de la intervención humana para aprender. Son los expertos quienes determinan el conjunto de características necesarias para comprender las diferencias entre las entradas de datos y, por lo general, requieren más datos estructurados para aprender. Mientras que el deep learning es un campo dentro del machine learning, cuya mayor diferencia es la forma en la que aprenden los algoritmos. Puesto que el deep learning automatiza gran parte de la extracción de características del proceso de aprendizaje, eliminando parte de la intervención humana y permitiendo el uso de conjuntos de datos más grandes.

  Trabajos como el de Hassan M. et al. [33], en el que utilizan técnicas como SVM, Naive Bayes y Árboles de decisión para crear modelos de detección de emociones, y crean un framerwork (EmotexStream) para clasificar en tiempo real textos en streaming. Llegan a obtener resultados con un porcentaje de aciertos del 90 %.

- **Híbrido**:

  Por último, los enfoques híbridos utilizan una combinación de los diferentes enfoques vistos anteriormente. Intentando suplir las limitaciones de uno con los beneficios del otro. Trabajos como el de Riahi et al. [60], proponen un enfoque basado en la combinación de tres subsistemas. El primero basado en un algoritmo de ML, el segundo con un modelo VSM (Vector Space Model) fundado en métodos matemáticos, y el tercero es un modelo basado en palabras clave (keywords). De esta forma si los tres subsistemas están de acuerdo en el tipo de emoción, el texto es etiquetado con ella.

Caben destacar trabajos como el de Sun et al. [71], el cual muestra cómo modelos de lenguaje pre-entrenados basados en Transformers vistos anteriormente, como BERT, han sido usados con éxito en tareas de clasificación de texto. Donde se concluye, entre otros puntos, que de las distintas capas que componen su estructura y que capturan diferentes niveles de información semántica y sintáctica. La última capa, es la que captura más información y resulta más útil para el ámbito de la clasificación de texto al mostrar una menor tasa de error. Como podemos en la capa 11 mostrada en la Tabla 2.1.

Layer	**Test error rates (%)**
Layer-0	11.07
Layer-1	9.81
Layer-2	9.29
Layer-3	8.66
Layer-4	7.83
Layer-5	6.83
Layer-6	6.83
Layer-7	6.41
Layer-8	6.04
Layer-9	5.70
Layer-10	5.46
Layer-11	**5.42**

Cuadro 2.1: Fine-tuning de BERT utilizando el dataset IMDb. [13]

Finalmente, en cuanto a la clasificación de emociones con modelos basados en Transformers, trabajos recientes como el de Tavirum el at. [1] que trata de clasificar emociones fuera del inglés, se encuentra con un problema similar de este trabajo. Como es la falta de datos. Aun así, el ajuste fino de modelos multilingües concluyó que se obtienen resultados superiores a técnicas de machine learning tradicionales o redes neuronales profundas como CNN y LSTM. Con una mejora de rendimiento desde un 5 % a un 29 %, dependiendo del tipo de tarea. Como podemos ver en la imagen 2.2, en este caso con un incremento del 17.7 %.

	Model	**Accuracy**	**F1 Score**
Others	LSTM [27]	**59.2**	**52.9**
	CNN [27]	54.0	53.5
	NB [27]	52.5	52.5
	SVM [27]	49.3	49.8
Ours	BERT-base	60.4	59.1
	XLM-RoBERTa-base	69.8	66.6
	XLM-RoBERTa-large	**72.7**	**70.6**

Cuadro 2.2: Resultados sobre el dataset para detección de emociones de Youtube. [1]

2.3. Análisis y Taxonomías de emociones

Como ya se ha comentado anteriormente, el análisis de emociones es una tarea que se centra en la identificación de emociones a partir del texto. A diferencia del análisis de sentimientos cuyo objetivo es determinar la polaridad de un texto (positivo, negativo o neutral), al clasificar emociones el número de clases por lo general aumenta y con ello la complejidad para determinar de qué emoción se trata. La definición de estas clases es un tema comúnmente abordado en la psicología, siendo la identificación de las emociones básicas un tema ampliamente discutido.

En ellas, encontramos emociones agradables como el amor (felicidad, alegría) y otras dos que hacen referencia a sensaciones desagradables como la ira (furia, enfado, enojo) y el temor (miedo, terror, pánico). Siendo el enfoque propuesto por Ekman [21] con seis emociones básicas, el más extendido y ampliamente utilizado con frecuencia. Sin embargo, también se han diseñado diferentes modelos que explican la diversidad de expresiones emocionales como combinación de las emociones básicas, o más bien, de los estados emocionales básicos. Denominándose emociones complejas

El debate en los últimos años se centra en clasificar las emociones a partir de "categorías difusas" de estados emocionales y de modelos factoriales que convergen en un modelo circular polar (circumplejo) que plantea extremos opuestos de categorías emocionales. Permitiendo así, explicar la variedad de estados emocionales.

La primera propuesta a destacar basada en este enfoque, es la Teoría circumpleja de las emociones de Plutchik [53], mostrada en la Figura 2.16. En la cual, formuló un modelo que identifica las emociones primarias y la combinación o fusión de las mismas, derivando en emociones y sentimientos más complejos.

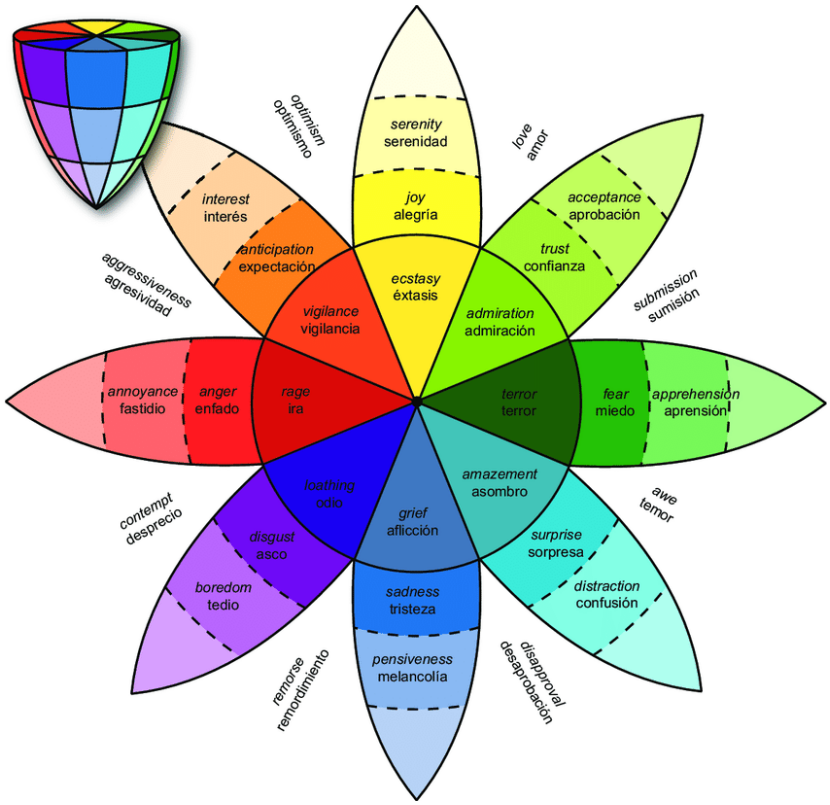


Figura 2.16: Modelo circumplejo tridimensional de Plutchik. [50]

En segundo lugar destaca el Modelo Circumplejo de Russel [63], en el que trató de identificar los factores que se encuentran involucrados en la respuesta emocional o estado emocional específico, describiéndolas como "bipolares e independientes" [4].

Este modelo, mostrado en la Figura 2.17, propone dos ejes a partir de los cuales es posible considerar expresiones emocionales o estados específicos así como la intensidad de éstas en base al nivel de activación y el grado de placer que se experimenta.

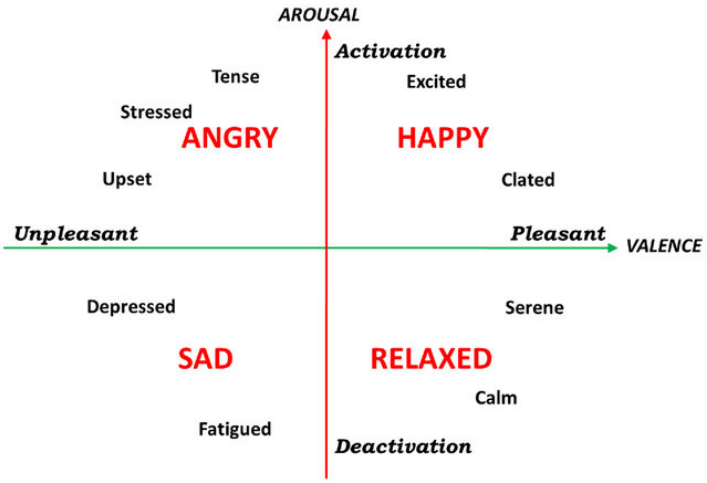


Figura 2.17: Modelo Circumplejo de Russel. [32]

De igual forma que los enfoques comentados anteriormente han servido en este trabajo como base para la identificación de emociones de interés como la tristeza, el miedo, y la ira, así como para estados que se enmarcan a partir de las combinaciones de las mismas. Otros trabajos también hacen uso de ellos, como el de Plaza-del-Arco et. al [51], donde se realizó la compilación de un corpus basado en las seis emociones básicas de Ekman junto con otras adicionales. Con este mismo enfoque, en 2020, el TASS (Taller de Análisis Semántico en la Sociedad Española para el Procesamiento del Lenguaje Natural) realizó un trabajo de extracción de emociones básicas de tweets en español [74].

Como se ha comentado anteriormente en el documento, la mayoría de conjuntos de datos en español se centra en las emociones básicas de Ekman. Entre ellos cabe destacar el generado para la tarea EmoEvalES, organizado por IberLEF 2021 [65]. Este hecho hace que se echen en falta conjuntos en nuestro idioma que incluyan más clases fuera de las emociones básicas, que permitan realizar pruebas con diferentes modelos y con ello una mayor investigación. Como podrían ser datasets en inglés similares a GoEmotions [16] con 27 categorías de emociones, o CancerEMO [69]. Un conjunto de datos creado a partir textos de una comunidad de salud online y anotado con ocho emociones.

Capítulo 3

Análisis de objetivos y metodología

Una vez conocida la importancia del PLN a lo largo de nuestra historia más reciente, pasamos a establecer los objetivos que se pretenden conseguir en este trabajo y que se irán desgranando a lo largo del documento, junto con la metodología seguida para su resolución.

3.1. Objetivos

El principal objetivo que se persigue en este trabajo es la utilización de diferentes modelos del lenguaje para el análisis de emociones de textos en español. De esta manera se puede obtener una primera aproximación de cómo es el funcionamiento y desempeño de este tipo de modelos en nuestro idioma y sobre una tarea tan específica. Ya que actualmente los resultados que se obtienen de su utilización en diferentes áreas como modelado del lenguaje o respuesta de preguntas, dan resultados realmente buenos e impulsa su continua investigación y desarrollo.

Para poder conseguirlo surgen otra variedad de tareas a realizar previamente. La primera y principal de todas ellas es la creación de un nuevo corpus de datos con el que entrenar y evaluar los modelos. Esto se debe a que en la actualidad no existen datos suficientes en español etiquetados con las emociones que se pretenden utilizar en este trabajo, y los que existen cuentan únicamente con algunas de las emociones más comunes como son la tristeza y la ira, o se centran en una única muy concreta como el odio (hate-speech). Es por ello que este objetivo se centra en conseguir un nuevo conjunto inicial de textos en español, que puede ser incrementado y mejorado con el tiempo con el que realizar las pruebas, y con el que aportar algo nuevo y de utilidad a los datos existentes en nuestro idioma.

Otra de las metas perseguidas, es el estudio de las técnicas de aprendizaje automático más actuales, utilizadas y extendidas en los últimos años. Como es BERT, un tipo especial de Transformer. Las cuales supusieron un gran avance en el campo del PLN por los resultados obtenidos en la resolución de diferentes tareas dentro del mismo. Por tanto, una explicación del funcionamiento de los mismos, las tecnologías que implementan y las diferencias existentes entre unos modelos y otros es a su vez uno de los objetivos perseguidos para una mejor resolución final del trabajo.

Finalmente, la visualización y el análisis del desempeño obtenido por los diferentes tipos de modelos a la hora de clasificar será también de importancia. Con ello se conseguirá identificar de entre todos, los que mejores resultados dan, cuales pueden ser los motivos de ello, y una posible selección de los mismos para futuras vías que se puedan desarrollar. De esta forma, se podrá observar cómo se desenvuelven y si puede ser de interés seguir investigando y profundizando en ellos para este tipo de tareas, la posibilidad de obtener nuevos datos a incluir en los existentes, ajustar los modelos de forma diferente o probar otros que han quedado fuera en este trabajo.

3.2. Metodología

En cuanto a la metodología seguida, puede observarse en la Figura 3.1 el proceso de la misma. Cada punto consta de las diferentes partes en las que se divide, algunos de ellos simplificados, puesto que se desarrollarán completamente más adelante en el documento. Este procedimiento, ayuda a una mejor estructuración de las tareas a realizar, facilitando la resolución de las mismas antes de avanzar a otra. Asegurando de esta forma que no queden tareas pendientes.

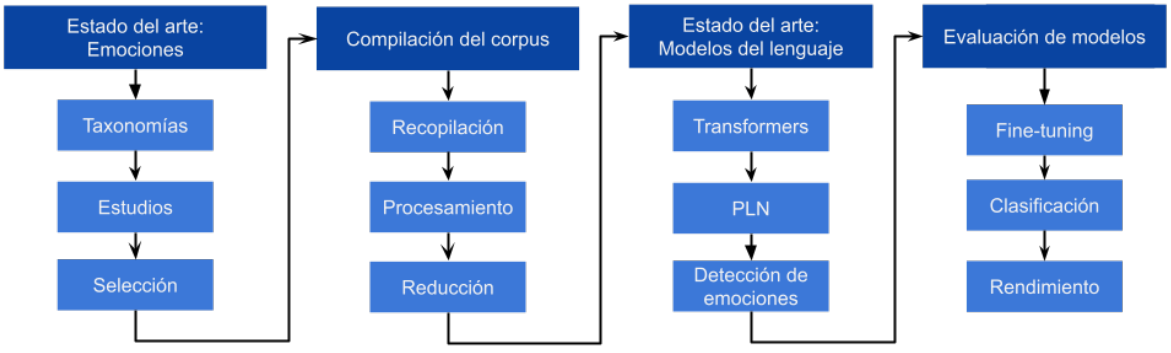


Figura 3.1: Metodología seguida.

Un primer paso para el desarrollo de este trabajo fue la búsqueda de diferentes taxonomías de emociones para decidir si alguna de ellas podría ajustarse con el tipo de emociones deseadas (soledad, depresión, tristeza, sentimientos suicidas, etc). Entre ellas, destaca la taxonomía de las emociones básicas propuesta por Ekman [21]. La cual a pesar de contar con pocas emociones de interés, sirvió para la selección de las más próximas a las deseadas, como son la tristeza y la ira. La siguiente taxonomía tenida en cuenta fue la rueda de las emociones de Plutchik's. La cual ayuda a clasificar las emociones en emociones primarias y las respuestas dadas a dichas emociones, se trata de una representación más amplia que la anterior y contempla otras emociones, con lo que sirvió de ayuda para la selección de emociones como grief, disgust o remorse. Además, el tratamiento de diferentes taxonomías sirvió para proporcionar ideas sobre cómo se estructuran las diferentes emociones para algunos autores, y cuáles podrían resultar de mayor interés para este trabajo.

Para la obtención del resto de emociones se hizo uso de diferentes estudios y artículos como el de Ángela Leis et. al [41], en el que se identificaba una lista de palabras que podrían sugerir signos de depresión. Con la idea en mente de que los términos proporcionados además de identificar signos de depresión podrían representar otros estados, se extrajo los de mayor puntuación y cuáles podrían ser algunos de estos estados o emociones alternativos que podían representar. Como puede ser soledad, desesperanza, pensamientos suicidas, etc.

El siguiente paso tras decidir cuales serían las emociones a utilizar, fue la búsqueda y recopilación de diferentes corpus que pudieran responder a estas necesidades. Para ello, se hizo uso tanto del motor de búsqueda de google para localización de datos (Google Dataset Search [30]), como de diferentes artículos relacionados con emociones concretas, como la soledad o el suicidio. Los cuales proporcionaban sus datos para libre uso. Tras finalizar este proceso y obtener todas las emociones deseadas, algunas en mayor o menor medida, fue necesario procesar cada uno de ellos individualmente. De esta forma se asegura la eliminación de emociones descartadas y textos duplicados, la limpieza de los mismos, y la traducción de los corpus en inglés. Seguido de la reducción de las clases que provocan un desbalanceo muy notable en el conjunto.

Una vez finalizadas todas la tareas anteriores se contaba tanto con las emociones a utilizar como con un conjunto final de tweets. Por tanto, se pasó a un comprobación del estado actual del arte para identificar cuales eran las técnicas más extendidas tanto en el desarrollo de artículos como en su elección para tareas del lenguaje natural y detección de emociones. Identificando algunas como Transformers, BERT, RoBERTa o ELECTRA. Con esto en mente, se pasó a explorar cómo funcionan estas arquitecturas, las mejoras que proporcionan y qué modelos podrían ser de interés utilizar basados en ellas.

Debido al desempeño de las mismas en este tipo de tareas se decidió usar modelos base tanto de BERT, de RoBERTa como de Electra, todos ellos basados en Transformers pero entrenados con distintos corpus en diferentes idiomas. En la siguiente lista se pueden ver todos los modelos finalmente seleccionados, comentados más adelante.

- Roberta-base
- Roberta-base-bne
- Bertin-roberta-base
- BETO
- RuPERTa
- Robertuito
- Electra
- Electricidad

Antes de continuar con el proceso de evaluación, se tuvo en cuenta las especificaciones de la máquina en la que se harían las pruebas, mostradas en la Figura 3.2, y se decidió en base a ello, como se comentará a continuación.

Especificaciones de la máquina de pruebas	
Sistema	Windows 10
Procesador	AMD Ryzen 7 1700 Eight-Core 3.00 GHz
RAM	16,0 GB
GPU	ASUS GeForce GTX 1050ti

Figura 3.2: Máquina de pruebas utilizada.

Por último, para la evaluación de cada modelo se utilizó la herramienta Huggingface la cual proporciona todos los modelos pre-entrenados con sus respectivos pesos finales. Puesto que ya se cuenta con un conjunto de datos específico para esta tarea de clasificación tan solo es necesario realizar el proceso de fine-tuning. El corpus creado se dividió en tres subconjuntos: Training y Validation utilizados para hacer fine-tuning, y un conjunto de Test para las pruebas finales mostradas en el punto 5 de resultados.

Obteniendo la distribución de clases por subconjunto mostrada en la tabla 3.1.

Emoción	**Label**	**Training**	**Validation**	**Test**
Neutral	10	4246	1357	1389
Depressed	1	3032	999	1048
Suicidal	13	2897	1031	1020
Disappointment	2	2883	949	933
Lonely	8	2733	947	847
Disgust	3	1362	462	465
Fear	5	1252	425	459
Anger	0	728	212	242
Embarrassment	4	662	222	210
Sadness	12	595	201	205
Remorse	11	572	193	191
Hopeless	7	545	177	185
Nervousness	9	375	125	111
Grief	6	186	56	48

Cuadro 3.1: Distribución de instancias para Training, Validation y Test

Realizada esta división, se observa un claro desbalanceo en el número de instancias de las últimas clases. Para tratar de corregirlo en la medida de lo posible y que el modelo no caiga en un error sistemático (Bias) con estas clases, se utilizó una función de pérdida basada en pesos. Para ello, se calcula la frecuencia de distribución de las clases en el conjunto y se establece un coeficiente entre 0 y 1 para la función de pérdida. El objetivo de esto es proporcionar un mayor peso a las clases menos comunes y uno más bajo a las de mayor aparición.

El paso siguiente fue definir los hiperparámetros. El principal motivo por el que se descartó una búsqueda de los mejores hiperparámetros es debido a que el artículo original de BERT [17], muestra el procedimiento de fine-tuning y un rango de posibles valores de hiperparámetros que funciona bien para la mayoría de tareas, como son:

- Batch size: 16, 32
- Learning rate (Adam): 5e-5, 3e-5, 2e-5
- Number of epochs: 2, 3, 4

Por tanto, teniendo esto en cuenta esto junto con las especificaciones de la máquina en la que se iban a realizar las pruebas, se seleccionaron los valores de hiperparámetros que aparecen en la Figura 3.3:

Hiperparámetro	Valor
Batch size	16
Learning rate	5e-5
Number of epochs	2
Weight decay	0.01
Maximum Sequence length	64

Figura 3.3: Valores de hiperparámetros.

- **Batch size**: Determina el número total de muestras de entrenamiento existentes en un único batch.
- **Learning rate**: Es la tasa de aprendizaje, y determina cuánto debe cambiar el modelo en respuesta al error estimado cada vez que se actualizan sus pesos. Un valor demasiado pequeño podría suponer un entrenamiento muy largo hasta converger en un resultado o atascarse. Mientras que un valor demasiado grande puede resultar en un proceso de entrenamiento inestable incrementando la tasa de error en lugar de reducirla.
- **Number of epochs**: Indica el número de veces que todo el conjunto de entrenamiento se pasa a través del modelo, en este caso 2 veces. Ya que un mayor número se observó que no se producía mejoras sustanciales sino que llegaba incluso a producir peores resultados de clasificación.
- **Weight decay**: Esta técnica de regularización de pesos se estableció a 0.01 para agregar una pequeña penalización a la función de perdida.
- **Maximum Sequence length**: Representa el número máximo de tokens de la entrada. Los tokens de entrada se truncan o se rellenan según su valor. Establecida en 64, ya que el incremento de la misma desembocaba en errores al entrenar por falta de memoria en la GPU.

Una vez establecidos estos hiperparámetros para el proceso de fine-tuning. Se utilizaron los mismos sin variarlos para todas las pruebas, independientemente del modelo. El objetivo es observar el desempeño producido por los modelos seleccionados, puesto que algunos de ellos se basan en el mismo tipo de arquitectura, como pueden ser los basados en RoBERTa. De esta forma se podrá identificar cuáles producen mejores resultados habiendo partido todos de la misma línea.

Finalmente, tras el entrenamiento de cada modelo se realizó un proceso de análisis de los resultados obtenidos junto con el planteamiento de algunas cuestiones. Para ello, se hizo uso de la matriz de confusión. Esta gráfica permite ver cuáles son las predicciones realizadas por un modelo y compararlas con las etiquetas reales, ayudando a identificar la confusión producida de unas clases con otras. Además, se obtuvo el reporte de clasificación de cada modelo, donde queda reflejada la precisión (precision), la exhaustividad (recall) y la F1 del modelo para cada clase. La métrica de precisión proporciona una idea de la calidad del modelo a la hora de clasificar cada emoción, mientras que la métrica de exhaustividad informará sobre la cantidad que el modelo es capaz de identificar de ese tipo. Por último, la F1 se trata de la media armónica de precisión y exhaustividad, por lo que facilitará la comparación de rendimiento entre modelos. Todo esto, se verá más en profundidad llegado el punto 5, donde se mostrarán y analizarán los resultados.

Capítulo 4

Resolución del trabajo

La finalidad de este apartado es definir cómo se han resuelto las diferentes tareas en las que se dividió el trabajo comentadas en el punto 3.2 de la metodología. Mostrar en qué ha consistido la de selección de emociones, el proceso de compilación del corpus así como el tratamiento realizado sobre el mismo. Ofreciendo finalmente una breve explicación de los modelos del lenguaje seleccionados para su evaluación sobre el conjunto de datos.

4.1. Emociones seleccionadas

Las diferentes taxonomías mostradas anteriormente en el punto 2.3, asentaron las bases para una selección inicial de algunas de las mostradas en estos enfoques como son: **anger** (ira), **disgust** (repulsión), **fear** (miedo), **grief** (pena), **sadness** (tristeza) y **remorse** (remordimiento). Sin embargo, quedaban fuera otros estados de interés relacionados con la salud mental y que podrían sugerir, por ejemplo, signos de depresión.

Diferentes trabajos que tratan este tema se centra en la identificación de palabras en el texto que puedan sugerir estos estados sobre una persona. Como el de Angela Leis et al. [42], en el que se detectan signos de depresión en tweets en español. O el trabajo de Jianhong Luo et al [37] que explora patrones de comportamiento suicida en las redes sociales. Con está idea en mente sobre la clasificación de palabras relacionadas con un estado, se utilizó una lista de 255 palabras vinculadas con la depresión publicadas en el articulo de Leis A. et al [41]. La selección de palabras fue elaborada por psiquiatras expertos con experiencia clínica, miembros del Instituto de Neuropsiquiatría y Adicciones (INAD) del Parc Salut Mar Barcelona, España. En ella se incluye tanto las palabras como una puntuación asignada a las mimas, obtenida a través de la suma de las puntuaciones que cada evaluador proveía mediante una escala tipo Likert (de 1 a 5) según la relevancia de esa palabra para ser usada por un paciente con depresión.

Ordenadas por puntuación de mayor a menor, las 30 primeras posiciones muestras palabras como las que aparecen a continuación:

- **Deprimido**/a 97
- Triste 97
- Tristeza 96
- **Desanimado/a** 94
- **Depresión** 92
- Depresivo/a 92
- Ansiedad 91
- Cansado/a 91
- Lloro 90
- Agobio 89
- Ansioso/a 89
- Insomnio 89
- **Nervioso**/a 89
- Agobiado/a 87
- Angustiado/a 87
- Angustia 87
- Agotado/a 86
- Decaído/a 86
- Preocupado/a 86
- **Desesperado**/a 85
- **Desmotivado/a** 84
- **Solo**/a 84
- **Desilusionado**/a 82
- Incapaz 82
- **Pena** 82
- Intranquilo/a 81
- **Vacío/a** 81
- Fatigado/a 80
- Sueño 80

En base tanto a los términos marcados en negrita como el resto que aparece en dicha lista, los cuales muestran puntuaciones realmente altas e incluso aparecen reflejados como sinónimos del mismo término. Se extrajo el estado o emoción alternativo que podrían reflejar algunos de ellos. Como pueden ser: **depressed** (deprimido), **disappointment** (decepcionado), **embarrassment** (avergonzado), **hopeless** (desesperanzado), **lonely** (solo), **remorse** (remordimientos), **nervousness** (nervios) y **suicidal** (suicida).

Las clases ahora mostradas junto con las anteriores emociones más la clase neutral, dan lugar a las 14 emociones/estados finales que aparecen a continuación.

- Neutral
- Depressed
- Suicidal
- Disappointment
- Lonely
- Disgust
- Fear
- Anger
- Embarrassment
- Sadness
- Remorse
- Hopeless
- Nervousness
- Grief

4.2. Compilación del corpus

Como se comentó al inicio del documento, la selección de emociones que no entren dentro de las siete emociones básicas de Ekman, o que no sean emociones muy concretas como el odio o la toxicidad, cuyo interés de análisis ha crecido debido a la gran cantidad de mensajes de este tipo que a día de hoy se siguen produciendo en redes sociales. Provoca que no dispongamos de datos suficientes para su estudio en castellano. Por ello, en la siguiente sección se comentarán los diferentes corpus de mensajes seleccionados, tanto en español como en inglés, en su mayoría provenientes de la red social Twitter y una pequeña parte de Reddit. Además del motivo de su selección, las emociones que aportan, el procesamiento realizado sobre ellos para obtener los datos de mayor interés y la traducción final de los que originalmente estaban en inglés.

4.2.1. Corpus utilizados

Conocidas las emociones finales seleccionadas, el siguiente paso derivó en la búsqueda de corpus que contuvieran mínimo una o varias de estas emociones. Debido a la pandemia y el incremento del uso de las redes sociales, también aumentó durante este tiempo la cantidad de conjuntos de datos publicados relacionados con este tema, como ocurre con el primer dataset obtenido. Proviene del artículo *An Approach using BETO on Spanish Tweets* [3], presentado en el International Workshop on Software Engineering Automation: A Natural Language Perspective (NLP-SEA), en 2021, en el cual se proporcionaban todos los recursos utilizados en el mismo. Entre ellos, el recurso de mayor interés y seleccionado, contiene un conjunto de 5260 oraciones extraídas de Twitter, relacionados con el COVID-19 y en castellano. Cada tweet marcado con tan solo una de las siguientes emociones: sadness, not-relevant, fear, happiness, anger o surprise.

A continuación se muestran las emociones seleccionadas junto con un ejemplo:

Sadness

> *"Creo q nadie va a olvidar la pandemia "COVID-19" ... ni los niños."*

Fear

> *"Tengo tanto miedo de ir a mi cita con el médico mañana porque yo no quiero coger el virus de la corona espero que nadie allí tiene mañana cuando vaya."*

Anger

> *"La Agencia Europea del medicamento en breve hará un comunicado. Sí hay relación entre la vacuna ASTRA-ZENECA y los trombos. Más vale tarde que nunca."* 😠

El segundo corpus pertenece de un conjunto de datos creado para el análisis de sentimientos. Este tipo de análisis se centra en la detección de la polaridad de un texto (positivo, neutral o negativo). De ahí que los únicos datos de interés extraídos fueran los tweets neutrales, con el objetivo de darle al modelo la capacidad de identificar textos sin una emoción específica. Lo cual suele ocurrir la mayoría de las veces que se escribe. Un texto no tiene necesariamente una emoción implícita y puede que simplemente transmita una información sin ir más allá, por tanto, es de interés incluirlos en el conjunto final.

Neutral

> *"Hay que aprender a darles a las personas el mismo valor e importancia que ellos nos dan."*

En cuanto al tercer y último conjunto de datos obtenido en español, pertenece al artículo *Detecting Signs of Depression in Tweets in Spanish* [42], y fue creado para la detección de signos de depresión en tweets en español. Promovido por la misma idea que la creación del corpus resultante de este documento, y es que la mayoría de estudios de trastornos mentales se centran en mensajes escritos únicamente en inglés. La metodología seguida para su obtención se basó en la extracción de tweets de 90 usuarios que habían mencionado explícitamente sufrir depresión y que incluía expresiones indicativas de este estado.

Depressed

> *"A veces quiero desaparecer y ver a quien le importo,pero luego pienso en que no le importo a nadie,y se me quitan las ganas de todo."*

La unión de estos tres conjuntos y la extracción de las emociones de interés como son miedo, tristeza, enfado, depresión y neutral, nos deja con un primer conjunto de datos de 12.535 tweets en castellano sin tratar, y la siguiente distribución de emociones.

- neutral: 7042
- sadness: 1667
- fear: 801
- anger: 2025
- depressed: 1000

Para la obtención del resto de emociones fue necesario la utilización de corpus en inglés. El primero y más destacado de ellos es GoEmotions [16]. Un conjunto de datos creado por Google, extraído de subreddits populares en inglés y etiquetado manualmente con 27 categorías de emociones, incluyendo 12 categorías de emociones positivas, 11 negativas, 4 ambiguas y 1 neutral. Para validar que las elecciones taxonómicas que realizaron coincidían con las emociones dadas, se llevó a cabo un análisis de componentes principales o PCA (Principal component analysis), que permitió comparar dos conjuntos de datos mediante extracción de combinaciones lineales con mayor variabilidad. GoEmotions, es uno de los corpus más destacables en este campo y resulta de gran utilidad para este tipo de tareas de clasificación, proporcionando las siguientes emociones: Remorse, Disappointment, Nervousness, Embarrassment, Grief, Fear y Disgust.

Algunos ejemplos de oraciones etiquetadas con estas emociones son:

Remorse

> *"sorry I completely forgot about these, but yeah you definitely changed my view"*

Disappointment

> *"I go to a neighborhood mexican restaurant monthly for the past 2.5 years, but nobody knows my name 😕"*

Nervousness

> *"Literally shaking, I hope you're getting the support you need with the ptsd this has obviously caused."*

Mostrando todas las emociones seleccionadas en este conjunto, obtenemos como resultado la siguiente distribución:

- disappointment: 6769
- disgust: 3420
- fear: 2514
- embarrassment: 1720
- remorse: 1648
- nervousness: 946
- grief: 494

Los siguientes cuatro conjuntos de datos provienen de la publicación *Detecting Depression in Social Media* [54]. Fue realizada, como su nombre indica, para la detección de signos de depresión en las redes sociales y contiene las siguientes emociones de interés: Depressed, Hopeless, Lonely y Suicide. Para su obtención se utilizó la herramienta TWINT, la cual permite extraer tweets sin utilizar directamente la API de Twitter. Además permite hacer uso de sus operadores de búsqueda, proporcionando así la capacidad de obtener Tweets de usuarios específicos, relacionados con ciertos temas, hashtags o tendencias.

En este caso, su extracción se realizó mediante la búsqueda de términos relacionados con la depresión, y puesto que en este proceso se pueden obtener tweets no relevantes, se verificó el etiquetado de cada tweet manualmente. Algunos tweets de ejemplo son los siguientes:

Depressed

> *"This show broke my heart it was so intense. I didn't mean to binge it but couldn't stop and went to bed late and depressed."*

Hopeless

> *"Sometimes I just want to scream about how unjust the world is, how heartless people can be, and how hopeless it makes me feel sometimes."*

Lonely

> *"i feel so lonely on here sometimes so if anyone ever wants to talk just please let me know :) i've just moved away from my entire life so i'm feeling very isolated and friendless."*

Suicide

> *"Every night I cry in my bed, thinking of ways to kill myself. I badly wanted to end my life for the past years but I couldn't. If I didn't know that suicide is a sin, I would've done it long ago."*

Los anteriomente mostrados tweets, son algunos de los ejemplos que implican directamente la emoción sobre el creador del tweet. Serán el tipo de oración de interés y más adelante se comentará el proceso seguido para la obtención de las mismas, ya que la distribución de emociones inicial es la mostrada a continuación.

- depressed: 20.932
- hopeless: 9021
- lonely: 53.772
- suicidal: 25.128

Por último, el tercer corpus utilizado es "Life!' [38], creado con fines de investigación sobre el suicidio. Se seleccionó un corpus procedente de mensajes suicidas en la red social Reddit, el cual cuenta con 2084 tweets etiquetados con con las categorías de "Risk" o "No Risk". Siendo los etiquetados con "Risk" los seleccionados únicamente.

Suicidal (Risk)

> *"That's all I want. It hurts too much. Stop telling me what I already know. I don't want to hurt anyone but I can't keep doing this shit."*

La agrupación inicial de todos los corpus mostrados sin tratar, ha permitido obtener sobre las 14 clases decididas los resultados mostrados en la Tabla 4.1.

Emoción	**Instancias**
Lonely	58.795
Depressed	21.932
Suicidal	15.851
Hopeless	10201
Neutral	7042
Disappointment	6769
Disgust	3420
Fear	3315
Anger	2025
Embarrassment	1720
Remorse	1648
Sadness	1667
Nervousness	946
Grief	494

Cuadro 4.1: Recuento de clases sin tratar.

4.2.2. Procesamiento realizado

A pesar de que los corpus mostrados anteriormente hayan sido creados en base a criterios de selección bien formados, fue necesario un procesamiento individual de algunos de ellos como paso intermedio antes de la unión final de todos en un solo conjunto y de realizar prueba alguna.

Motivos como que los datos se encuentran en idiomas diferentes o la procedencia de los mismos no es de una única red social y por tanto la longitud de los textos puede variar. Además, aunque los mensajes de un mismo conjunto estén etiquetados con una única emoción, dentro del mismo pueden haber oraciones que reflejen mejor dicha emoción que otras, por lo que una selección de ellas beneficiaría al modelo a la hora de clasificar y poder conseguir un mejor entendimiento del contexto.

Traducción

Como se ha comentado al final del apartado anterior, la mayor diferencia que existe entre los textos es el idioma. Una parte importante del trabajo fue realizar la traducción, para la cual se contemplaron varias opciones. El uso de un traductor profesional como *SYSTRANS* o un traductor "normal" como *Google Traductor* o *DeepL*, que permitieran la traducción de archivos. Su uso facilitaría la capacidad de obtener datos en nuestro idioma, pero puede surgir la duda de que al utilizar este mecanismo no se obtengan buenos resultados y que, en el proceso de traducción, se puede perder el sentido original de una oración, y por tanto los resultados no serían igual de buenos que con textos originalmente en el idioma deseado.

En la actualidad esto deja, en gran medida, de ser cierto. Como se comentó en la sección 2 (Estado del arte), los modelos actuales son capaces de entender el contexto en el que se enmarca un texto y cuando está gramaticalmente bien formado, es decir, obedece las reglas de morfología y sintaxis, son capaces de realizar traducciones al nivel de un humano.

Podemos ver un ejemplo de lo que ocurre para la emoción lonely a continuación:

Lonely (Original)

> *"I try to surround myself with as much as people as I can that's possible so I can at least forget how lonely I always feel but when everyone has their own thing to do and just leave I instantly get in a bad mood."*

Lonely (Traducción Humana - Traductor e interprete de inglés)

> *"Intento rodearme de tanta gente como sea posible para así poder olvidar lo solo/a que me siento siempre, pero cuando los demás tienen cosas que hacer y se van, me pongo de mal humor."*

Lonely (Traducción Máquina - Google traductor)

> *"Trato de rodearme de tanta gente como sea posible para que al menos pueda olvidar lo solo que siempre me siento, pero cuando todos tienen sus propias cosas que hacer y simplemente se van, instantáneamente me pongo de mal humor."*

Vemos que el resultado de traducir una oración bien formada por parte de una máquina es muy similar al de un humano. Se producen cambios en la selección de palabras sinónimas pero el significado original de la frase se sigue manteniendo. La máquina incluso llega al nivel de conocer las reglas ortográficas y saber dónde debe añadirlas, como vemos que hace al separar con comas. Sin embargo, la buena formación gramatical no implica la buena formación semántica, es decir, aquella oración o expresión que tiene sentido. Con lo que si una mala formación gramatical y semántica ocurren a la vez o solamente no se da que una oración esté bien formada, los resultados de una traducción pueden ser pésimos. Como se puede ver en el siguiente ejemplo.

Lonely (Original)

> *"anyone ever want talk u always dm bc sometimes even people anitwt may feel lonely anyways like talk anyone everyone feel free talk."*

Lonely (Traducción Máquina - Google traductor)

> *"alguien alguna vez quiere hablar siempre dm bc a veces incluso las personas pueden sentirse solas de todos modos como hablar cualquiera todos se sienten libres de hablar."*

La traducción realizada deja mucho que desear, pero de igual forma ocurriría con una traducción humana ya que no es sencillo entender el sentido de una oración al estar mal formada. No obstante, este ejemplo proviene de un conjunto de datos descartado debido a que la mayoría de sus textos contenían oraciones de este tipo, mal formadas gramatical y semánticamente.

Mientras que, tanto en el conjunto de datos del cual proviene el ejemplo anterior a este como en el resto de los seleccionados, donde se mostraban ambas traducciones de humano y máquina. La gran mayoría de sus oraciones están bien formadas y por tanto su traducción será lo mas fiel posible al significado original. Siendo este junto con lo comentado anteriormente, el principal motivo por el cual la traducción no supone un impedimento para no ser usada junto con textos que originalmente no estaban en el idioma deseado.

Selección de datos

Algunos de los conjuntos de datos obtenidos son notablemente más grandes en comparación con otros. Esto supondría un problema si se hiciera la unión de todos ellos directamente, ya que provocaría un desbalanceo entre emociones muy amplio, y nos encontraríamos en el caso de contar con muchas de un tipo y muy pocas de otro. Pudiendo llevar a obtener resultados muy buenos por parte del modelo, pero la realidad es que solo sería bueno para el tipo de emoción más existente. Se haría experto en la detección de ese tipo, pudiendo llegar a fallar con emociones que aparecen de forma inferior en el conjunto. Por tanto lo que se busca en este siguiente punto es explicar el proceso seguido para reducir y obtener los tweets más relevantes de los conjuntos con mayores datos, con el objetivo de conseguir un mejor equilibrio que el actual.

Como se ha visto en el apartado anterior, la cantidad de tweets en algunos conjuntos incluso llega a superar los 50.000. Apareciendo en mayor número los que corresponden a las emociones depressed, lonely, hopeless y suicidal. Es por ello que con el objetivo de obtener los textos que representen mejor dicha emoción, son los que van a sufrir un proceso similar de reducción. Dicho proceso consiste la selección de conjuntos de palabras que estén altamente relacionados con la emoción a reducir, procedentes de estudios que buscan la identificación de patrones y factores de riesgo en un texto para identificar palabras clave que son comúnmente usadas al escribir sobre una emoción en concreto.

Comenzando con el conjunto que contiene las oraciones etiquetadas con la emoción depressed, inicialmente cuenta con 20.932 tweets. Algunos de ellos con una longitud superior a la permitida por twitter (280 caracteres), lo cual puede ser debido a la concatenación de tweets dentro de un mismo hilo de mensajes. Mientras que el conjunto de palabras que podrían sugerir signos de depresión, proviene del artículo *Detecting Signs of Depression in Tweets in Spanish* [42].

Comentado en la sección 4.2.1 ya que fue utilizado también su corpus, recordemos que su objetivo fue la identificación de características lingüísticas y patrones de comportamiento de usuarios de Twitter que podrían sugerir signos de depresión. Por lo que el proceso de selección de estos términos incluyó la verificación de un psicólogo de las declaraciones dadas por 90 usuarios seleccionados que admitían sufrir depresión.

Dicho conjunto lo conforman las siguientes palabras:

- overwhelmed
- exhausted
- anxiety
- anxious
- tired
- low
- depression
- depressed
- discouraged
- desperate
- demotivated
- insomnia
- cry
- nervous
- worried
- lonely
- sad
- empty

En el proceso de reducción, se buscaba la condición de que mínimo una de las palabras presentes en la lista se encontrara en la oración tratada junto con que la longitud de dicho texto fuera superior a los 100 caracteres pero no superara los 280 caracteres permitidos por Twitter. Consiguiéndose reducir de 20.932 a 4.993 tweets.

El siguiente conjunto que se redujo fue el que contenía los tweets etiquetados como hopeless y lonely. Para ello, se utilizaron tres grupos diferentes de palabras que podrían sugerir soledad de algún tipo (loneliness, lonely y solitude). Esto es debido a que no es lo mismo estar solo de forma elegida y deseada, a sentirse solo por motivos sociales, por tanto, los términos utilizados para comunicar una forma u otra forma de soledad varían. Estos términos fueron extraídos del artículo *SOLO: A Corpus of Tweets for Examining the State of Being Alone* [39], cuyo objetivo fue realizar un análisis y exploración de los términos asociados al estado de soledad utilizados en nuestra forma de comunicarnos en internet.

En el proceso de obtención, se utilizó la API de twitter para extraer tweets desde el 28 de agosto de 2018 al 10 de julio de 2019 en base a un lista de palabras y frases cortas relacionadas con el estado de soledad, y creada con ayuda de psicólogos. De los tweets obtenidos, se descartaron los duplicados, los tweets cortos y los tweets con url's externas, para finalmente dar una puntuación a las palabras más relacionadas con los términos utilizados para la búsqueda de tweets.

De esta forma, se obtuvieron 3 listas con las 25 palabras más frecuentes para loneliness, lonely y solitude, de las cuales se muestran solamente algunas de las palabras que las componen a continuación:

lonelines_terms:

- alone
- lonely
- pain
- isolation
- killing
- feeling
- depression
- sadness
- fear
- feelings

lonely_terms:

- feel
- feeling
- friends
- single
- bored
- sad
- alone
- sometimes
- felt
- feels

solitude:

- alone
- peace
- loneliness
- quiet
- lonely
- enjoy
- silence
- fortress
- hundred
- enjoying

Cabe destacar cómo algunos de los términos presentes en la lista *solitude* son positivos, y esto es debido al motivo indicado anteriormente sobre que este tipo de soledad es "positiva" generalmente. En cuanto a la reducción, se realizó mediante un proceso similar al anteriormente indicado. Una selección por longitud del texto y en este caso, que contuviera alguna de las palabras de entre las tres listas para hopeless y mínimo una de las palabras de cada lista para lonely. Consiguiendo de esta forma un total de 5102 tweets para lonely y 1191 para hopeless.

Por último, los términos relacionados al suicidio fueron extraídos del artículo *Exploring temporal suicidal behavior patterns on social media* [37]. Cuyo objetivo fue examinar posibles patrones de comportamientos suicidas mediante la evaluación de un conjunto de tweets relacionados con el suicidio. Se descubrieron 13 factores de riesgo clave y se identificaron patrones de comportamiento en diferentes días. Además tras examinar esta gran cantidad de datos, se extrajeron diferentes expresiones con pensamiento suicida en base a los alta frecuencia de aparición de algunos términos, entre las que se incluyen palabras clave como las mostradas a continuación.

- suicide
- suicidal
- suic
- self-harm
- self-injury
- self harm
- self injury
- hang myself
- hung myself
- kill myself
- kills myself
- take my life
- takes my life
- want to die
- wanted to die
- wants to die
- want death
- wants death
- wanted death
- to be dead

Mediante la comprobación de que mínimo uno de los anteriores términos estuviera contenido en la oración examinada, se redujo el conjunto inicial de tweets de 25.128 a 6734.

Limpieza de texto

El proceso de limpieza se realizó como último paso tras la unión de todos los datos en un solo conjunto. El objetivo de esto fue crear una nueva colección con la que probar el mismo modelo y observar si se obtenía una mejora considerable de resultados al utilizar un conjunto u otro.
Los cambios aplicados al texto son lo siguientes:

- Normalización de caracteres. (Convertir caracteres especiales a una forma normalizada)
- Eliminaciones en el texto
  - Retweets (“RT TUSRUSER”, “TUSERUSER”)
  - Saltos de linea
  - Menciones (“@person”)
  - Caracteres no alfabéticos
  - Enlaces (“http, https”)
  - Imágenes (“img”)
  - Espacios en blanco
- Transformación a minúsculas

- Eliminación de tweets repetidos

Como resultado final de todo este procesamiento, se obtuvo el total mostrado en la Figura 4.2 para cada emoción.

Emoción	**Instancias**
Neutral	6995
Suicidal	6830
Depressed	5949
Lonely	5023
Disappointment	4765
Disgust	2289
Fear	2235
Anger	1388
Sadness	1201
Hopeless	1180
Embarrassment	1094
Remorse	956
Nervousness	611
Grief	290

Cuadro 4.2: Recuento final de clases.

4.3. Modelos de lenguaje evaluados

Existen infinidad de modelos con los que entrenar basados en diferentes tecnologías. En este caso, todos los modelos seleccionados están basados en las comentadas anteriormente (Transformers, BERT y Electra), previamente entrenados con diferentes textos en español o en inglés. Todos ellos se encuentran disponibles desde la herramienta Huggingface.

Este punto se centra en dar una breve explicación de cual es la tecnología usada, los datos utilizados y la técnica de entrenamiento utilizada, tan solo sobre ciertos modelos. Puesto que ya ha sido explicada su arquitectura anteriormente y las diferencias con las que cuenta cada uno.

Roberta-base

Bertín

Bertín pertenece a una serie de modelos basados en BERT para español [62]. El objetivo de este proyecto era entrenar previamente un modelo basado en RoBERTa desde cero durante el evento de la comunidad Flax/JAX, en el que Google Cloud proporcionó TPUv3-8 gratis para realizar el entrenamiento usando las implementaciones Flax de Huggingface de su biblioteca. Este proyecto cuenta con varios modelos utilizables, en este caso se ha seleccionado *roberta-base-spanish* para observar el funcionamiento de otro modelo basado en roberta pre-entrenado con un corpus en español.

Roberta-base-bne

Al igual que Bertin, se basa en el modelo base de RoBERTa, con la diferencia de que ha sido pre-entrenado utilizando el corpus en español más grande conocido hasta la fecha, con un total de 570 GB de texto limpio y deduplicado procesado para este trabajo, compilado a partir de los rastreos web realizados por la Biblioteca Nacional de España de 2009 a 2019, [10].

RuPERTa-base

Este modelo creado por la comunidad de Huggingface [8], está de nuevo basado en ROBERTA y entrenado con un gran corpus en español [61]. Al utilizar tres modelos distintos basados en la misma arquitectura, con dos de ellos entrenados en español, podremos ver los resultados y la importancia de un corpus a la hora de pre-entrenar un modelo y si esto afecta a la hora de clasificar.

BETO (bert-base-spanish-wwm-uncased)

Modelo basado en BERT entrenado con un gran corpus en español [8]. Tiene un tamaño similar a BERT-Base y fue entrenado con la técnica de enmascaramiento de palabras completas (Whole Word Masking)[9].

Robertuito-base-uncased

Modelo de lenguaje pre-entrenado para contenido generado por usuarios en español, entrenado siguiendo las pautas de RoBERTa en 500 millones de tweets. Supera a otros modelos de lenguaje pre-entrenados para este lenguaje como BETO, Bertin y RoBERTa-BNE. Las 4 tareas seleccionadas para la evaluación fueron: detección de discurso de odio, análisis de sentimientos, de emociones y detección de ironía.[49]

Electra-base-discriminator

Electricidad-base-discriminator

Este modelo proporcionado por la comunidad de Huggingface [61], está basado en Electra con la diferencia de haber sido entrenado con el corpus de BETO. De esta forma podemos visualizar el desempeño realizado por el discriminador de ELECTRA con textos en español y ver si existen diferencias notables con el discriminador original de Electra-base.

Capítulo 5

Evaluación y análisis de resultados

Este apartado tiene como objetivo la comprobación y discusión de los resultados obtenidos por cada modelo seleccionado. Para ello, se mostrarán las métricas de Accuracy, Macro F1 y Weighted F1. En este caso, puesto que existe un desbalanceo entre clases, la proporción de coincidencias correctas (precisión) no es recomendable utilizarla como evaluador del rendimiento del modelo. En cuanto a la métrica Macro F1, se calcula tomando la media aritmética de todas las puntuaciones F1 por clase. Tratando a todas las clases por igual, independientemente del número de clases que aparezcan de cada una. Por tanto, la penalización del modelo será mayor cuando el modelo no pueda clasificar las clases minoritarias.
Lo cual no implica un mal o pobre rendimiento del modelo, simplemente puede que sea necesario incluir más apariciones de estas clases si se desea y espera que el modelo llegue a clasificarlas correctamente. Por último, la métrica Weighted F1 se calcula tomando la media de todas las F1 obtenidas por clase teniendo en cuenta el número de ocurrencias de cada clase. Con lo que las clases de mayor número contribuyen más positivamente.

Por lo general, trabajando con un conjunto de datos desbalanceado donde todas las clases tienen la misma importancia usaríamos la métrica Macro F1 como referencia del desempeño obtenido por el modelo. Sin embargo, en este caso en particular no todas las clases tienen la misma importancia, puesto que clases como depressed, suicidal o lonely tienen una importancia de clasificación mayor y se espera que se clasifiquen correctamente antes que clases como grief, la cual incluso algunas taxonomías vistas en el punto 2.3 la enmarcaban junto con otras como la tristeza. Además, a pesar de esta posible penalización producida por este tipo de clases, se decidió incluirlas para comprobar que sucede con ellas, de qué forma afectan y si son suficientes instancias para conseguir cierto nivel de clasificación.

Por tanto, teniendo todo esto en cuenta, se dará mayor importancia a la métrica Weighted F1 ante Macro F1, sin dejar a esta última completamente a un lado.

Podemos ver lo anteriormente comentado en la Figura 5.1, donde por lo general todos los modelos obtiene resultados entre 72 % y 82 % para la métrica Weighted F1, mientras que se produce una penalización cercana al 20 % para la todos los modelos en la Macro F1. Además, como se comentó anteriormente en el punto 2.2, se realizó un procesamiento de limpieza sobre los datos. Aun así, no supone una mejora considerable puesto que la mayoría de modelos obtiene resultados muy similares utilizando un conjunto de datos u otro. Incluso se observa que para la mayoría de modelos los resultados empeoran entre un 1 % y un 5 % aproximadamente, con lo que el análisis siguiente se centrará en los resultados obtenidos para los modelos entrenados con los datos sin limpiar.

Dataset	Métrica	Roberta-base	Roberta-base-bne	RuPERTa	BETO	Bertin	Robertuito	Electra	Electricidad
Sin procesar	Accuracy	75.9%	82.9%	77.9%	80.2%	81.8%	83%	74.4%	79.4%
	Macro F1	51.1%	68.2%	56.4%	65.0%	61.5%	69.3%	52.8%	59.9%
	Weighted F1	72.1%	82.2%	76.1%	79.7%	80.2%	82.5%	72%	77.9%
Procesado	Accuracy	77.6%	81.2%	73.8%	79.2%	80.9%	79.2%	70.3%	76%
	Macro F1	55.9%	67.3%	56.1%	64.0%	65.9%	65.6%	46.9%	56%
	Weighted F1	75.8%	81.1%	72.9%	79.1%	80.5%	79%	66.5%	75%

Figura 5.1: Resultados obtenidos.

Observando de nuevo la Figura 5.1 vemos como todos los modelos basados en RoBERTa que han sido entrenados con corpus en español, obtienen mejores resultados que el propio RoBERTa-base. Ocurriendo de igual forma con Electricidad (ELECTRA español) al compararlo con ELECTRA-base. Además, en cuanto a resultados obtenidos destacan los modelos de Roberta-base-bne, BETO, Bertín y Robertuito, tanto en la métrica Macro como Weighted F1. Con lo que nos centraremos en ellos para observar el desempeño obtenido por cada uno, en relación a las emociones seleccionadas.

Una primera visualización de sus matrices de confusión, mostradas en las Figuras 5.2, 6.1, 6.2, 6.6 respectivamente, manifiestan resultados en general muy similares para todas las emociones. Es por ello que a continuación se muestra tan solo la Figura 5.2, el resto de Figuras (6.1, 6.2 y 6.6), aparecen en el Anexo 1 del punto 6.1.

Entre estas emociones destacan depressed, hopeless, lonely, neutral y suicidal. Tanto a ser emociones de mayor importancia como a clases que mejores resultados de clasificación proporcionan. Donde la confusión de predecir estas clases con otras, es muy bajo. Siendo muy llamativo y destacable en la clase lonely, neutral y suicidal debido a ser todavía más inferior al resto.

Por el contrario, se observa que clases como grief, nervousness y embarrassment son las que peores resultados ofrecen y producen un mayor error de clasificación.

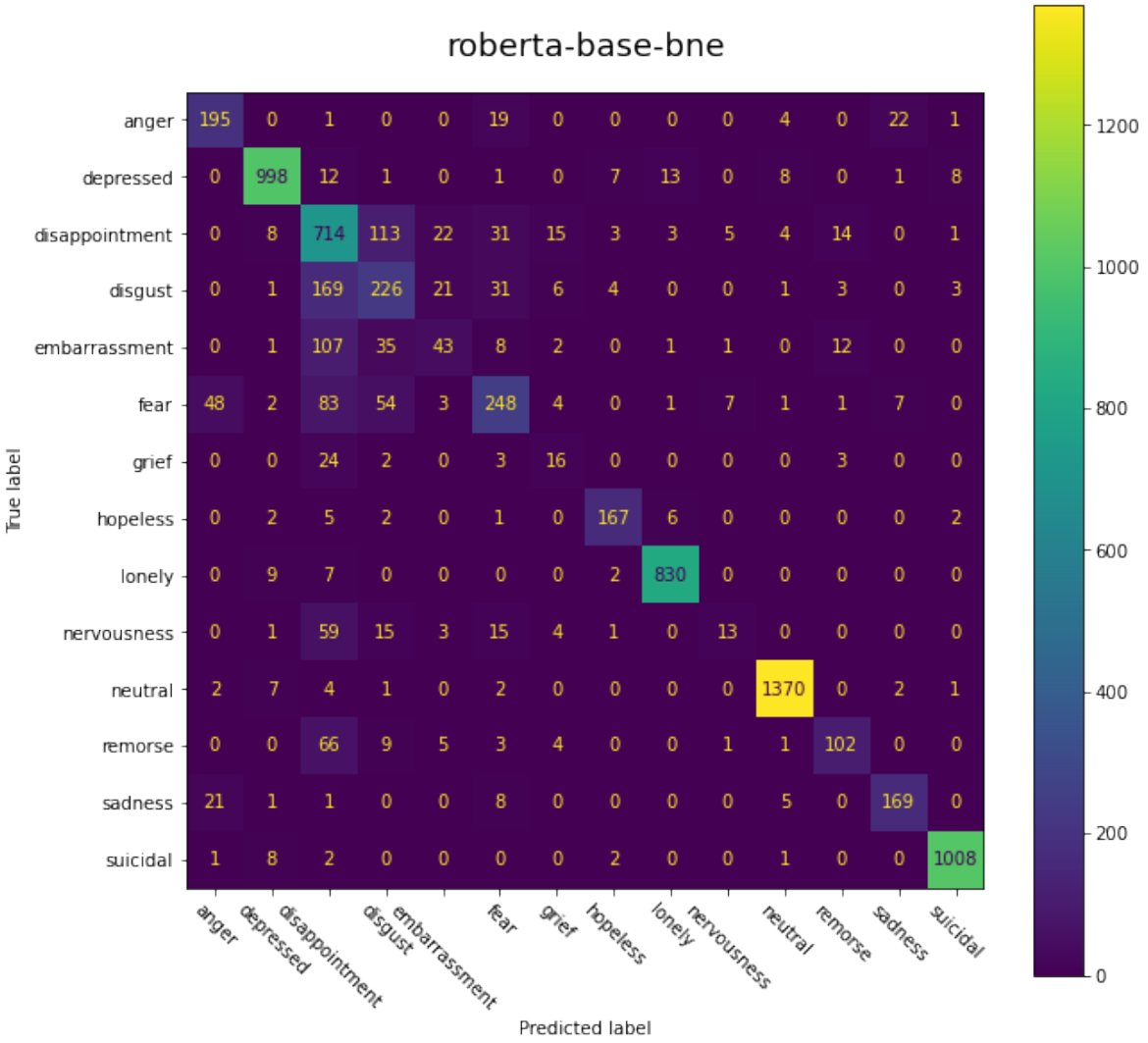


Figura 5.2: Matriz de confusión RoBERTa-base.

Son estas 3 clases las que provocan principalmente la penalización en la Macro F1 por su bajo número de aparición en el conjunto de datos. Sin embargo, embarrassment, habiendo contando con un número de ocurrencias (662) cercano a remorse (572), la cual junto con grief (186) y nervoussness (375) son de las que menor número contaban a la hora de entrenar. Ofreciendo incluso peores resultados que remose.

También cabe destacar que para la clase remorse se produce, sin importar el modelo, cierta confusión con la clase disappointment. Dicha confusión, analizando de nuevo las Figuras, se observa que se produce de igual forma en clases como disgust, embarrassment, fear y nevousness para todos los modelos. Siendo la clase disgust, la que mayor confusión con la clase disappointment produce en todos ellos.

Antes de pasar a comentar cual puede ser el motivo de dicha confusión de clases, mediante la Figura 5.3 se proporcionará una visualización del reporte de clasificación obtenido para cada modelo en cada una de las clases, haciendo uso de las métricas Precision, Recall y Weighted F1. De esta forma se obtendrá una idea el desempeño obtenido de manera individual para cada emoción.

	RoBERTa-base-bne			BETO			Bertín			Robertuito		
Emoción	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
neutral	98.2%	98.6%	98.4%	96.3%	93.5%	94.9%	98.6%	98.9%	98.8%	98.4%	98.6%	98.5%
suicidal	98.4%	98.6%	98.5%	98.0%	98.6%	98.3%	97.9%	98.1%	98.0%	98.3%	98.5%	98.4%
lonely	97.1%	97.8%	97.5%	95.9%	97.7%	96.8%	97.1%	97.4%	97.2%	97.0%	97.2%	97.1%
depressed	96.1%	95.1%	95.6%	93.4%	93.2%	93.3%	94.9%	94.5%	94.7%	94.1%	94.5%	94.3%
hopeless	89.7%	90.2%	90.0%	90.2%	90.2%	90.2%	90.0%	87.5%	88.7%	95.9%	89.7%	92.7%
sadness	84.0%	82.4%	83.2%	80.1%	72.6%	76.2%	76.6%	75.1%	75.8%	85.6%	81.4%	83.5%
disappointment	56.9%	76.5%	65.2%	54.2%	70.4%	61.2%	53.6%	78.4%	63.7%	58.6%	73.7%	65.3%
anger	73.0%	80.5%	76.6%	68.6%	71.4%	70.0%	72.5%	73.1%	72.8%	75.1%	80.1%	77.6%
remorse	75.5%	53.4%	62.5%	72.1%	55.4%	62.7%	72.0%	56.5%	63.3%	74.5%	61.2%	67.2%
fear	67.0%	54.0%	59.8311	58.6%	54.0%	56.2%	65.1%	52.5%	58.1%	62.7%	53.5%	57.8%
disgust	49.3%	48.6%	48.9%	48.7%	49.2%	48.9%	46.6%	53.5%	49.8%	51.7%	56.7%	54.1%
embarrassment	44.3%	20.4%	28.0%	32.1%	16.6%	21.9%	0%	0%	0%	45.0%	21.4%	29.0%
grief	31.3%	33.3%	32.3%	24.3%	20.8%	22.4%	0%	0%	0%	35.7%	31.2%	33.3%
nervousness	48.1%	11.7%	18.8%	28.8%	11.7%	16.6%	0%	0%	0%	44.4%	14.4%	21.7%

Figura 5.3: Reporte de clasificación por modelo.

Se aprecia como las 6 primeras clases (neutral, suicidal, lonely, depresed, hopeles y sadness), son las que mejor clasifican, con los cuatro modelos proporcionando resultados muy similares. Entre el 80 y el 90 por ciento. Destacando entre ellos Robertuito, ya que además de proporcionar tales resultados para estas primeras clases, es el que mantiene junto con RoBERTa-base-bne, una mayor F1 para el resto de clases.

Con disappointment y anger, ambos son capaces todavía de identificar un alto número de las mismas. Como refleja la métrica Recall, entre el 76 y el 88 por ciento.

Sin embargo, para remose, fear y disgust, se aprecia como la métrica Recall confirma lo visto anteriormente en las matrices de confusión, y es que la cantidad que el modelo es capaz de identificar de ese tipo de clases desciende hasta valores entre el 48 y el 68 por ciento, haciendo que se comiencen a sufrir mayores errores al clasificar.

Por último, las 3 emociones finales (embarrassment, grief y nervousness) son las que peores resultados ofrecen por parte de los cuatro modelos. Bertín no llega a clasificar correctamente ninguna de ellas, mientras que en el resto de modelos la correcta identificación de las mismas tan solo se encuentra entre un 11 y un 30 por ciento.

En cuanto a RoBERTa-base, RuPERTa, ELECTRA y Electricidad, ninguno de ellos es capaz de clasificar embarrassment, grief o nervoussness correctamente, aunque fuera en menor medida de forma similar a Robertuito o Roberta-base-bne, como se puede ver en la Figura 5.4 más abajo. O en las Figuras 6.7, 6.8 y 6.12 que aparecen en el Anexo 1 del punto 6.1.

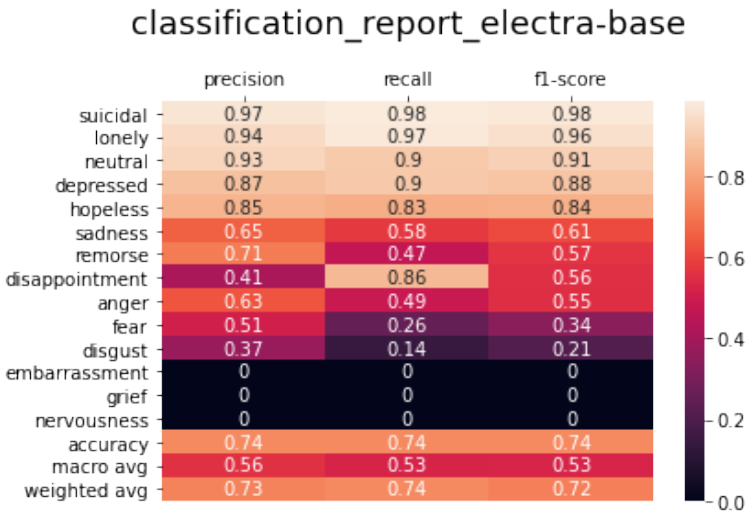


Figura 5.4: Reporte de clasificación para ELECTRA.

Además, a pesar de que para las primeras 5 clases (suicidal, lonely, neutral, depressed y hopeless) los resultados llegan a ser comparables a los modelos anteriormente comentados como Robertuito o Roberta-base-bne. A partir de sadnes, se produce un descenso en las métricas de clasificación obtenidas bastante amplio, aproximadamente de un 20 por ciento en comparación con dichos modelos.

Con lo que el rendimiento proporcionado no ha sido el esperado por ninguno de ellos. De RoBERTa-base podía esperarse algo menos de rendimiento puesto que fue pre-entrenado con datos en inglés y por tanto el entendimiento del contexto que pudiera llegar a extraer del texto, podría ser inferior a otros pre-entrenados con datos en español. Siendo este a su vez, uno de los motivos del bajo rendimiento de ELECTRA, al que se le añade el hecho de que a pesar de haber sido entrenado en una gran variedad de tareas de GLUE (General Language Understanding) entre ellas no se encontraba la clasificación de emociones.

Comentados los resultados obtenidos tanto de los que mejores resultados ofrecen como los que no. Se han podido identificar varios escenarios por parte de los comentados en la Figura 5.3 (Roberta-bne, BETO, Bertín y Robertuito), como son:

- **Alta precisión y alto recall**: Este escenario es el ideal, implica que cuando el modelo predice una clase esta suele ser la esperada y generalmente acierta con ella para la mayoría de sus casos. Con lo que se puede considerar al modelo como un buen identificador de estas clases. Como ocurre con neutral, suicidal, lonely, depressed, hopeless, sadness y anger. Conformando un total de 7 clases las cuales con una alta seguridad, el modelo es capaz de clasificar correctamente.

- **Alta precisión y bajo recall**: Este caso implica que cuando el modelo tiene que decidir clasificar una clase difícil de etiquetar y decide hacerlo, la mayoría de las veces la decisión es correcta. Sin embargo, una gran cantidad de las mismas no lo es. Como puede ser el caso de la clase remorse, fear o disgust. Las cuales, pese a no tener una precisión realmente alta, el hecho de ser mayor que la métrica recall provoca que se produzca este caso. Con una gran cantidad de aciertos pero de igual forma una alta cantidad de fallos de clasificación con diferentes clases.

- **Baja precisión y bajo recall**: Vemos como último caso, el escenario en el cual se encuentran las clases embarrasment, grief y fear. En este caso, la decisión por parte del modelo de etiquetar a una de estas clases, no es del todo segura. Lo que provoca la asignación de etiquetas incorrectas y por consiguiente la caída de precisión final obtenida.

Otro aspecto que ocurre de forma similar en todos, independientemente de sus resultados, es la anteriormente comentada confusión que sufren algunas clases con disappointment. Esta confusión podría entenderse sobre clases que pudieran estar ligadas como grief y sadness. puesto que grief también puede reflejar la pena que se siente. Sin embargo, entre las clases en las que se produce esta confusión, se encuentran emociones como disgust, embarrassment o fear. Las cuales deberían tener una relación directa menor con disappointment.

Un primer pensamiento puede hacer creer que realmente sí existieran ciertas similitudes en los textos que provocan esta confusión. En cambio, si observamos la Figura 5.5 mostrada al completo en el Anexo 1, Figura 6.13. Se muestra la frecuencia de aparición de las 20 palabras más comunes en las clases disappointment, disgust, embarrassment y fear, tras haber eliminado stopwords. Vemos que a pesar de que existen términos compartidos y repetidos por las clases, realmente la cantidad de los mismos no es sustancialmente importante y no debería suponer un problema mayor para producir tal confusión con estas clases.

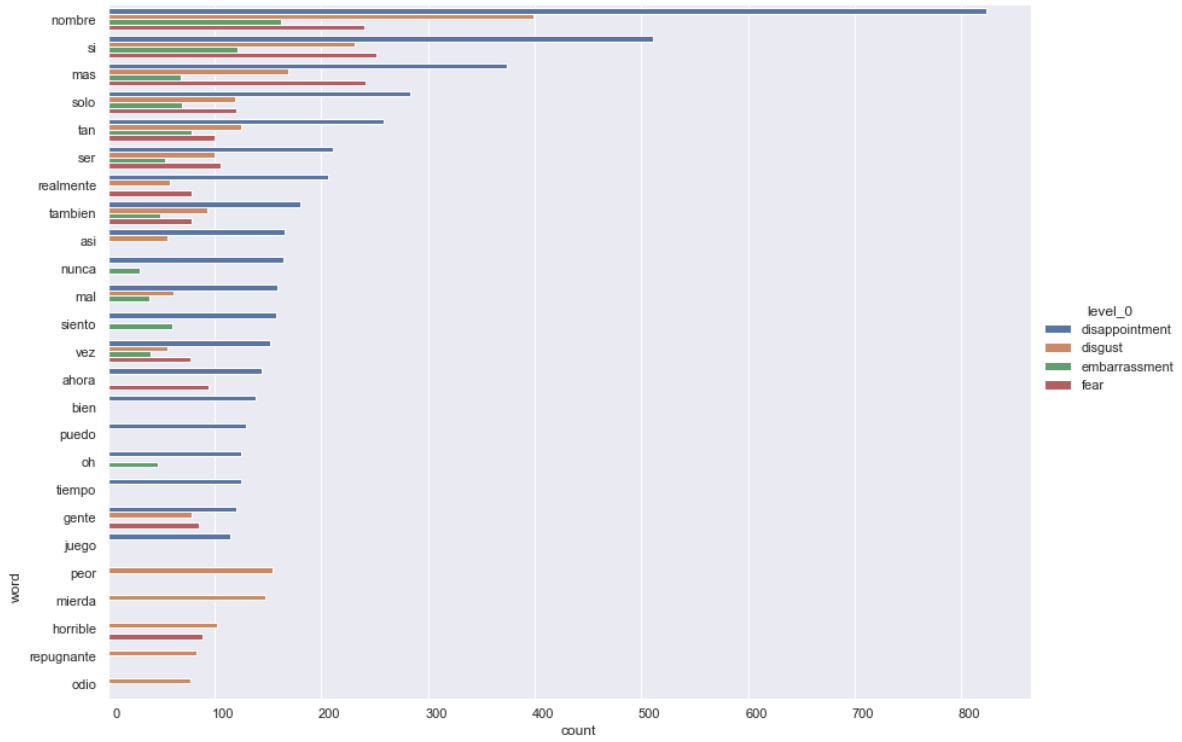


Figura 5.5: Frecuencia de términos (disappointment, disgust, embarrassment y fear).

Es más, si comparamos disappointment con clases que se clasifican realmente bien, como hopeless, así como con clases que no llegan a clasificar correctamente como nervousness. Se observa mediante la Figura 5.6 que hopeless, en verde, comparte incluso más términos que nervousness con disappointment, y sin embargo su clasificación es completamente superior a ella. Por tanto, esta primera idea de que el suceso de confusión de algunas clases con disappointment es debido a la cantidad de términos que comparten, realmente no es uno de los motivos clave por los que sucede este fenómeno.

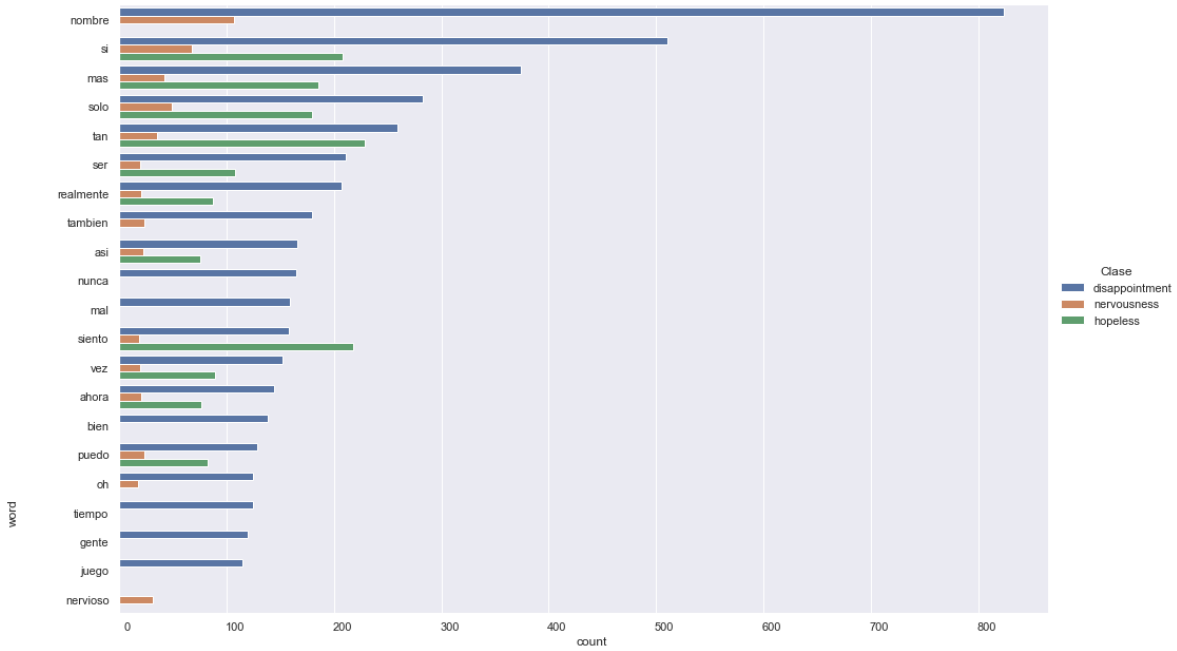


Figura 5.6: Frecuencia de términos (disappointment, hopeless y nervousness).

Un escenario no contemplado anteriormente ha sido el de baja precisión y alto recall. En el que se enmarca disappointment. Este escenario lo que provoca es que nuestro modelo piense que existen muchas clases de un determinado tipo, y a su vez que realmente son de ese tipo. Provocando tanto aciertos porque muchas de ellas realmente están enmarcadas en esa clase, como fallos puesto que otro tanto no lo está. En este caso, esta "creencia" podría provocar más clasificaciones del tipo disappointment por parte del modelo cuando a la hora de tener que clasificar sin total seguridad una clase, decida enmarcarla sobre la que cree que puede tener un mayor número de apariciones en el conjunto de datos. Siendo este un posible motivo de los errores de clasificación que se producen con clases que inicialmente no deberían suceder, mostrados anteriormente en la Figura 5.2.

Capítulo 6

Conclusiones y vías futuras

Como punto final a este trabajo, se considera que el desempeño obtenido por una buena parte de de los modelos para el análisis de emociones, sobre emociones y estados más complejos que las emociones básicas ha sido considerablemente bueno. Teniendo en cuenta que 11 de las 14 emociones son clasificadas correctamente, con 7 de las emociones de mayor importancia siendo reconocidas con gran exactitud y las 4 restantes con capacidades de poder llegar a obtener métricas similares a las 7 anteriores.

A esto se añade que por lo general el número de clases seleccionadas para este tipo de trabajos suele ser inferior, y que alguna de las clases utilizadas podrían haberse obviado a sabiendas de que se obtendrían mejores métricas. Con todo, se decidió dejarlas para observar el desempeño obtenido.

Además, se suma la traducción realizada sobre los textos. La cual a pesar de haber sido defendida en el punto 4.2.2 y ser considerada una muy buena herramienta, puede haber provocado en algunos casos cierta perdida del sentido original de un texto. Lo que resultaría en un peor entendimiento del contexto por parte del modelo y una peor identificación de la emoción correspondiente, debido a las diferencias entre el texto final traducido y la original. Sin embargo, se ha podido observar cómo los textos traducidos y las emociones que representan (depressed, hopeless, lonely, suicidal) son de los que mejores resultados proporcionan y mejor clasifican. Pudiendo haber sido de gran importancia tanto el procesamiento como la reducción aplicada a los mismos.

Por tanto, teniendo ambas cosas en cuenta se puede considerar que los resultados proporcionados son altamente buenos y generan un interés para continuar con la investigación sobre la clasificación de emociones sobre emociones relacionadas con las seleccionadas en este trabajo.

De igual forma, queda clara una necesidad por parte de los modelos de incluir más datos al conjunto inicial como trabajo futuro, tanto por parte de las clases que obtienen buenos resultados como principalmente de las que aparecen en menor número, ya que son las que mayor penalización provocan. De esta forma se podrá identificar si el problema de clasificación de las mismas radica en su bajo número de aparición, viene dado por la complejidad que ciertas emociones pueden suponer para la correcta identificación por parte del modelo, o se trata de otro motivo.

La realización de este proceso de obtención de nuevos datos, podría realizarse de una forma similar a la mostrada en este trabajo. Haciendo uso de corpus ya creados o directamente extrayendo textos de interés desde las propias redes sociales. No obstante, durante la realización de este trabajo, se consideró la idea de utilizar este tipo de modelos basado en Transformers de forma distinta.

Como se comento anteriormente en el documento, este tipo de modelos son ampliamente utilizados para diferentes tipos de tareas en el ámbito del PLN, entre ellas la generación de texto. Donde actualmente se puede destacar HuggingTweets [15], un proyecto que permite hacer fine-tuning de diferentes modelos basados en Transformers para la generación de tweets sobre un usuario específico de twitter. De forma que dichos tweets pudieran ser considerados como propios del autor.

Con este mismo pensamiento en mente, la idea surgida consistía en entrenar este tipo de modelos sobre una emoción en específico. De esta forma, si los resultados obtenidos son considerablemente buenos y se consiguen tweets que podrían haber sido escritos por una persona que, por ejemplo, se siente sola o con pensamientos depresivos. Podría ser una buena opción para conseguir de forma automática más textos sobre un determinado tipo de clase, sin recurrir a las herramientas más típicas y extendidas actualmente para la obtención de información. De esta forma, igual que los modelos de HuggingTweets llegan a ser capaces de entender el modo con el que se expresa una determinada persona en base a sus tweets, podrían llegar a ser capaces de identificar los tipos de expresiones más comúnmente utilizadas por textos que comparten la emoción que expresan, y de la misma forma generar textos que la reflejen.

Siendo estos algunos de los principales motivos por los que se considera como una vía de interés y con potencial a explorar en un futuro próximo.

Bibliografía

[1] Tanvirul Alam, Akib Khan y Firoj Alam. *Bangla Text Classification using Transformers*. 2020. arXiv: 2011.04446 [cs.CL].

[2] Jay Alammar. *The Illustrated Transformer*. URL: http://jalammar.github.io/illustrated-transformer/.

[3] Ariadna de Arriba, Marc Oriol y Xavier Franch. *Merging Datasets for Emotion Analysis. An Approach using BETO on Spanish Tweets - Supporting material*. Ver. 1.0. Ago. de 2021. DOI: 10.5281/zenodo.5191344. URL: https://doi.org/10.5281/zenodo.5191344.

[4] Lisa Barrett y James Russell. «Independence and Bipolarity in the Structure of Current Affect». En: *Journal of Personality and Social Psychology* 74 (abr. de 1998), págs. 967-984. DOI: 10.1037/0022-3514.74.4.967.

[5] Yoshua Bengio y col. «A Neural Probabilistic Language Model». En: *J. Mach. Learn. Res.* 3.null (mar. de 2003), págs. 1137-1155. ISSN: 1532-4435.

[6] Margaret M. Bradley y col. *Affective Norms for English Words (ANEW): Instruction manual and affective ratings*. 1999.

[7] Tom B. Brown y col. *Language Models are Few-Shot Learners*. 2020. arXiv: 2005.14165 [cs.CL].

[8] José Cañete. *Compilation of Large Spanish Unannotated Corpora*. Zenodo, mayo de 2019. DOI: 10.5281/zenodo.3247731. URL: https://doi.org/10.5281/zenodo.3247731.

[9] José Cañete y col. «Spanish Pre-Trained BERT Model and Evaluation Data». En: *PML4DC at ICLR 2020*. 2020.

[10] Casimiro Pio Carrino y col. *Biomedical and Clinical Language Models for Spanish: On the Benefits of Domain-Specific Pretraining in a Mid-Resource Scenario*. 2021. arXiv: 2109.03570 [cs.CL].

[11] Noam Chomsky. *Syntactic Structures*. The Hague: Mouton y Co., 1957.

[12] CIBERSAM. *La tendencia del suicidio en 2020*. URL: https://www.cibersam.es/noticias/la-tendencia-del-suicidio-en-2020-cambio-en-los-meses-mas-relevantes-de-la-pandemia.

[13] Kevin Clark y col. *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*. 2020. arXiv: 2003.10555 [cs.CL].

[14] Papers with Code. *Transformers*. URL: https://paperswithcode.com/methods/category/transformers.

[15] Boris Dayma. *HuggingTweets*. URL: https://github.com/borisdayma/huggingtweets.

[16] Dorottya Demszky y col. «GoEmotions: A Dataset of Fine-Grained Emotions». En: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, jul. de 2020, págs. 4040-4054. DOI: 10.18653/v1/2020.acl-main.372. URL: https://aclanthology.org/2020.acl-main.372.

[17] Jacob Devlin y col. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[18] Ketan Doshi. *Transformers Explained*. URL: https://towardsdatascience.com/transformers-explained-visually-part-1-overview-of-functionality-95a6dd460452.

[19] Cho Yoshua Bengio Dzmitry Bahdanau KyungHyun. «NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE». En: 2015, pág. 6.

[20] IBM Cloud Education. *Recurrent Neural Networks*. URL: https://www.ibm.com/cloud/learn/recurrent-neural-networks.

[21] Paul Ekman. «Lie Catching and Micro Expressions». En: *The Philosophy of Deception*. Ed. por Clancy Martin. Oxford University Press, 2009, págs. 118-133.

[22] Romana Ema, Tajul Islam y Md. Humayan Ahmed. «Detecting Emotion from Text and Emoticon». En: 17 (oct. de 2018), págs. 8-13.

[23] Mark Edward Epstein. «Statistical Source Channel Models for Natural Language Understanding». Tesis doct. USA, 1996. ISBN: 0591132230.

[24] Instituto Nacional de Estadística. *La salud mental en la pandemia*. URL: %5Chttps%7Bhttps://www.ine.es/ss/Satellite?L=es_ES&c=INECifrasINE_C&cid=1259953225445&p=1254735116567&pagename=ProductosYServicios%2FINECifrasINE_C%2FPYSDetalleCifrasINE#:~:text=Seg%C3%BAn%20la%20reciente%20Encuesta%20europea,en%20los%20problemas%20para%20dormir.%7D.

[25] Elisabetta Fersini, Debora Nozza y Paolo Rosso. «Overview of the Evalita 2018 Task on Automatic Misogyny Identification (AMI)». En: *EVALITA@CLiC-it*. 2018.

[26] N. Galanis y col. «Machine Learning Meets Natural Language Processing – The story so far». En: mar. de 2021.

[27] N. Galanis y col. «The Candide System for Machine Translation». En: 1991, págs. 1-6.

[28] José Antonio García-Díaz y col. «Detecting misogyny in Spanish tweets. An approach based on linguistics features and word embeddings». En: *Future Generation Computer Systems* 114 (2021), págs. 506-518. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2020.08.032. URL: https://www.sciencedirect.com/science/article/pii/S0167739X20301928.

[29] Google. *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*. https://huggingface.co/docs/transformers/model_doc/electra. 2020.

[30] Google. *Google Data Search*. URL: https://datasetsearch.research.google.com/.

[31] Graphcore. *Pre-Training and Fine-Tuning BERT for the IPU*. URL: https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html.

[32] Ardi Handojoseno. *Modelo Circumplejo de Russel*. URL: https://www.researchgate.net/profile/Ardi-Handojoseno.

[33] Maryam Hasan, Elke A. Rundensteiner y Emmanuel O. Agu. «Automatic emotion detection in text streams by analyzing Twitter data». En: *International Journal of Data Science and Analytics* 7 (2018), págs. 35-51.

[34] Mallory Hightower. *High-Level History of NLP Models*. URL: https://towardsdatascience.com/high-level-history-of-nlp-models-bc8c8b142ef7.

[35] Rani Horev. *BERT: State of the art language model for NLP*. URL: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

[36] Google - Huggingface. *ELECTRA*. URL: https://huggingface.co/docs/transformers/model_doc/electra.

[37] Luo Jianhong y col. «Exploring temporal suicidal behavior patterns on social media: Insight from Twitter analytics». En: *Health Informatics Journal* (2019). DOI: 10.1177/1460458219832043.

[38] Isabel Moreno José Manuel Gómez. *Life! corpus*. URL: https://github.com/PlataformaLifeUA.

[39] Svetlana Kiritchenko y col. *SOLO: A Corpus of Tweets for Examining the State of Being Alone*. 2020. DOI: 10.48550/ARXIV.2006.03096. URL: https://arxiv.org/abs/2006.03096.

[40] Prachi Kumar. *An Introduction to N-grams*. URL: https://blog.xrds.acm.org/2017/10/introduction-n-grams-need/.

[41] Angela Leis y col. «Clinical-Based and Expert Selection of Terms Related to Depression for Twitter Streaming and Language Analysis». En: *Studies in health technology and informatics* 270 (2020), págs. 921-925.

[42] Angela Leis y col. «Detecting Signs of Depression in Tweets in Spanish: Behavioral and Linguistic Analysis». En: *Journal of Medical Internet Research* 21 (jun. de 2019), e14199. DOI: 10.2196/14199.

[43] Yinhan Liu y col. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. 2019. arXiv: 1907.11692 [cs.CL].

[44] Yinhan Liu y col. «RoBERTa: A Robustly Optimized BERT Pretraining Approach». En: *CoRR* abs/1907.11692 (2019). arXiv: 1907.11692. URL: http://arxiv.org/abs/1907.11692.

[45] Dale Markowitz. *Understand the Model Behind GPT-3, BERT, and T5*. URL: https://daleonai.com/transformers-explained.

[46] Microsoft. *Machine Translation*. URL: https://www.microsoft.com/en-us/translator/business/machine-translation/.

[47] Saif M. Mohammad. «Sentiment Analysis: Automatically Detecting Valence, Emotions, and Other Affectual States from Text». En: *Emotion Measurement (Second Edition)*. Ed. por Herb Meiselman. Elsevier, 2021.

[48] Britney Muller. *BERT: State Of The Art NLP Model Explained*. URL: https://huggingface.co/blog/bert-101#1-what-is-bert-used-for.

[49] Juan Manuel Pérez y col. *RoBERTuito: a pre-trained language model for social media text in Spanish*. 2021. arXiv: 2111.09453 [cs.CL].

[50] Santiago Planet. *Modelo circumplejo tridimensional de Plutchik*. URL: https://www.researchgate.net/profile/Santiago-Planet.

[51] Flor Miriam Plaza del Arco y col. «EmoEvent: A Multilingual Emotion Corpus based on different Events». English. En: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, mayo de 2020, págs. 1492-1498. ISBN: 979-10-95546-34-4. URL: https://aclanthology.org/2020.lrec-1.186.

[52] Flor Miriam Plaza-del-Arco y col. «Improved emotion recognition in Spanish social media through incorporation of lexical knowledge». En: *Future Generation Computer Systems* 110 (2020), págs. 1000-1008. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2019.09.034. URL: https://www.sciencedirect.com/science/article/pii/S0167739X1931163X.

[53] «Theories of Emotion». En: *Theories of Emotion*. Ed. por Robert Plutchik y Henry Kellerman. Academic Press, 1980, pág. iv. ISBN: 978-0-12-558701-3. DOI: https://doi.org/10.1016/B978-0-12-558701-3.50003-X. URL: https://www.sciencedirect.com/science/article/pii/B978012558701350003X.

[54] Tulasi ram Ponaganti. *Detecting Depression in Social Media Via Twitter Usage*. URL: https://medium.com/swlh/detecting-depression-in-social-media-via-twitter-usage-2d8f3df9b313.

[55] Richard Socher - Alex Perelygin - Jean Wu - Jason Chuang - Christopher D. Manning - Andrew Ng - Christopher Potts. *Stanford Sentiment Treebank*. URL: https://paperswithcode.com/dataset/sst.

[56] Lara Quijano-Sanchez y col. *HaterNet a system for detecting and analyzing hate speech in Twitter*. Ver. 1.0. Zenodo, mar. de 2019. DOI: 10.5281/zenodo.2592149. URL: https://doi.org/10.5281/zenodo.2592149.

[57] QUISH. *Gradientes que desaparecen y explotan*. URL: https://es.quish.tv/vanishing-exploding-gradients.

[58] Fika Rachman, Riyanarto Sarno y Chastine Fatichah. «CBE: Corpus-based of emotion for emotion detection in text document». En: ene. de 2016, págs. 331-335. DOI: 10.1109/ICITACEE.2016.7892466.

[59] Colin Raffel y col. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. 2019. arXiv: 1910.10683 [cs.LG].

[60] Nooshin Riahi y Pegah Safari. «Implicit Emotion Detection from Text with Information Fusion». En: *Journal of Advances in Computer Research* 7.2 (2016), págs. 85-99. ISSN: 2345-606X. eprint: http://jacr.iausari.ac.ir/article_648851_53c8c1f92db4902658915191cdabeaf1.pdf. URL: http://jacr.iausari.ac.ir/article_648851.html.

[61] Manuel Romero. *Spanish Electra by Manuel Romero*. https://huggingface.co/mrm8488/electricidad-base-discriminator/. 2020.

[62] Javier De la Rosa y Eduardo G. Ponferrada y Manu Romero y Paulo Villegas y Pablo González de Prado Salas y María Grandury. «BERTIN: Efficient Pre-Training of a Spanish Language Model using Perplexity Sampling». En: *Procesamiento del Lenguaje Natural* 68.0 (2022), págs. 13-23. ISSN: 1989-7553. URL: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/6403.

[63] James Russell. «A Circumplex Model of Affect». En: *Journal of Personality and Social Psychology* 39 (dic. de 1980), págs. 1161-1178. DOI: 10.1037/h0077714.

[64] Moiz Saifee. *GPT-3: The New Mighty Language Model from OpenAI*. URL: https://medium.com/towards-data-science/gpt-3-the-new-mighty-language-model-from-openai-a74ff35346fc.

[65] Flor Miriam Plaza-del-Arco y Salud María Jiménez-Zafra y Arturo Montejo-Ráez y M. Dolores Molina-González y L. Alfonso Ureña-López y M. Teresa Martín-Valdivia. «Overview of the EmoEvalEs task on emotion detection for Spanish at IberLEF 2021». En: *Procesamiento del Lenguaje Natural* 67.0 (2021), págs. 155-161. ISSN: 1989-7553. URL: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/6385.

[66] Ministerio de Sanidad de España. *Encuesta Europea de Salud en España 2020*. URL: https://www.sanidad.gob.es/estadEstudios/estadisticas/EncuestaEuropea/Enc_Eur_Salud_en_Esp_2020.htm.

[67] FERDINAND DE SAUSSURE. *CURSO DE LINGÜÍSTICA GENERAL*. URL: https://fba.unlp.edu.ar/lenguajemm/?wpfb_dl=59.

[68] Grigori Sidorov y col. «Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets». En: *Advances in Artificial Intelligence*. Ed. por Ildar Batyrshin y Miguel González Mendoza. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, págs. 1-14.

[69] Tiberiu Sosea y Cornelia Caragea. «CancerEmo: A Dataset for Fine-Grained Emotion Detection». En: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, nov. de 2020, págs. 8892-8904. DOI: 10.18653/v1/2020.emnlp-main.715. URL: https://aclanthology.org/2020.emnlp-main.715.

[70] Carlo Strapparava y Alessandro Valitutti. «WordNet-Affect: an Affective Extension of WordNet». En: *Vol 4.* 4 (ene. de 2004).

[71] Chi Sun y col. *How to Fine-Tune BERT for Text Classification?* 2019. DOI: 10.48550/ARXIV.1905.05583. URL: https://arxiv.org/abs/1905.05583.

[72] Yu Sun y col. «ERNIE 2.0: A Continual Pre-Training Framework for Language Understanding». En: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.05 (abr. de 2020), págs. 8968-8975. ISSN: 2159-5399. DOI: 10.1609/aaai.v34i05.6428. URL: http://dx.doi.org/10.1609/AAAI.V34I05.6428.

[73] Ashish Vaswani y col. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

[74] Manuel García Vega y col. «Overview of TASS 2020: Introducing Emotion Detection». En: *IberLEF@SEPLN*. 2020.

[75] Wikipedia. *Expert Systems*. URL: https://en.wikipedia.org/wiki/Expert_system.

[76] Wikipedia. *Ferdinand de Saussure*. URL: https://es.wikipedia.org/wiki/Ferdinand_de_Saussure.

[77] Wikipedia. *LISP*. URL: https://en.wikipedia.org/wiki/Lisp_(programming_language).

[78] Wikipedia. *N-grams*. URL: https://en.wikipedia.org/wiki/N-gram.

[79] Wikipedia. *Turing test*. URL: https://en.wikipedia.org/wiki/Turing_test.

[80] Hmong - Wikipedia. *ALPAC - Automatic Language Processing Advisory Committee*. URL: https://hmong.es/wiki/ALPAC.

[81] Hmong - Wikipedia. *Experimento de Georgetown-IBM*. URL: https://hmong.es/wiki/Georgetown-IBM_experiment.

[82] Rachel Wolff. *Future of Natural Language Processing*. URL: https://monkeylearn.com/blog/nlp-trends/.

[83] Zhilin Yang y col. *XLNet: Generalized Autoregressive Pretraining for Language Understanding*. 2019. arXiv: 1906.08237 [cs.CL].

[84] Seunghyun Yoon, Seokhyun Byun y Kyomin Jung. «Multimodal Speech Emotion Recognition Using Audio and Text». En: *2018 IEEE Spoken Language Technology Workshop (SLT)* (dic. de 2018). DOI: 10.1109/slt.2018.8639583. URL: http://dx.doi.org/10.1109/SLT.2018.8639583.

Anexos

Los recursos adicionales a este documento con los que cuenta en este punto son: las imágenes de las matrices de confusión correspondientes al apartado 5, los reportes de clasificación obtenidos por modelo y el resto de Figuras que no han podido mostrarse completamente.

El código fuente utilizado tanto para el procesamiento de datos, limpieza, fine-tuning y evaluación de resultados. Así como los datos utilizados para la realización de este trabajo y el corpus finalmente generado, se encuentran alojados en GitHub: `https://github.com/AlejandroSalme/TFG-Corpus_AnalisisEmociones_UMU`

6.1. Anexo 1 - Imágenes de resultados

En este punto se muestran las matrices de confusión y los reportes de clasificación obtenidos por todos los modelos. Tanto los comentados en el documento como el resto de ellos.

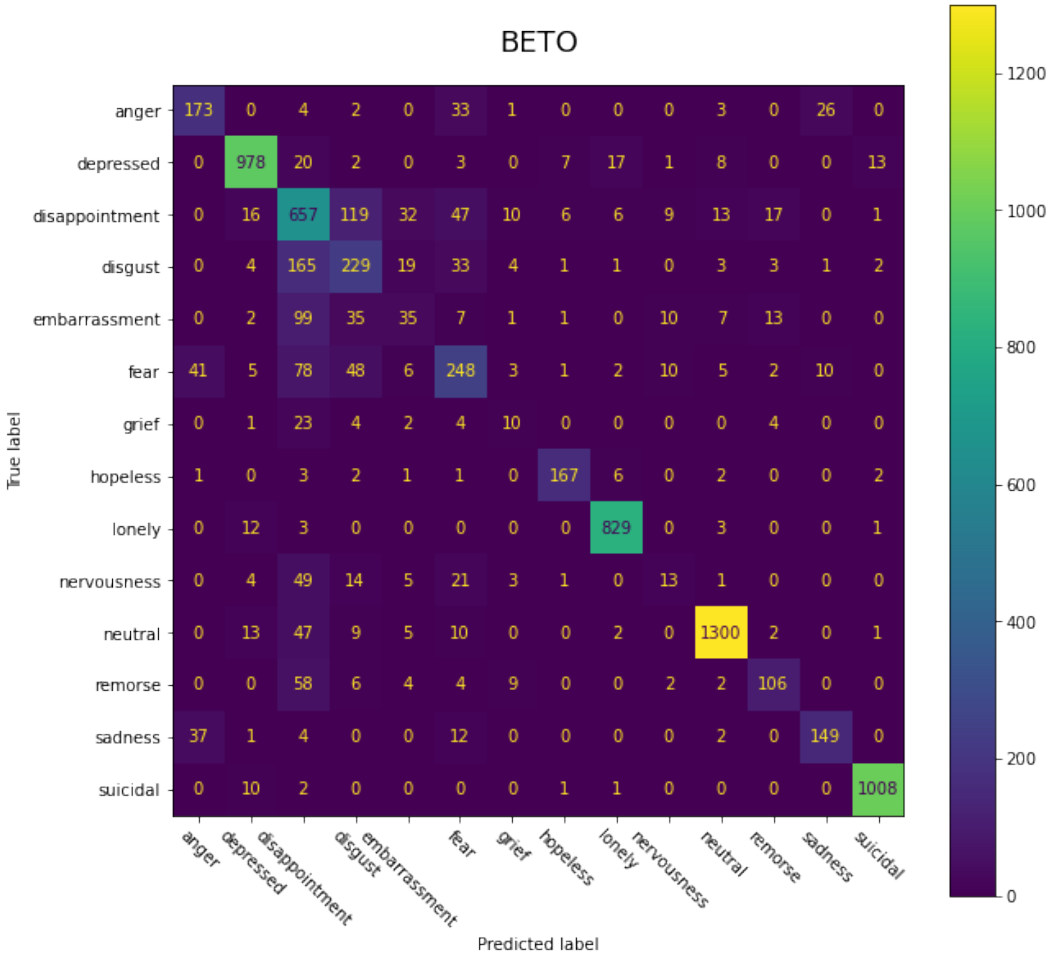


Figura 6.1: Matriz de confusión BETO.

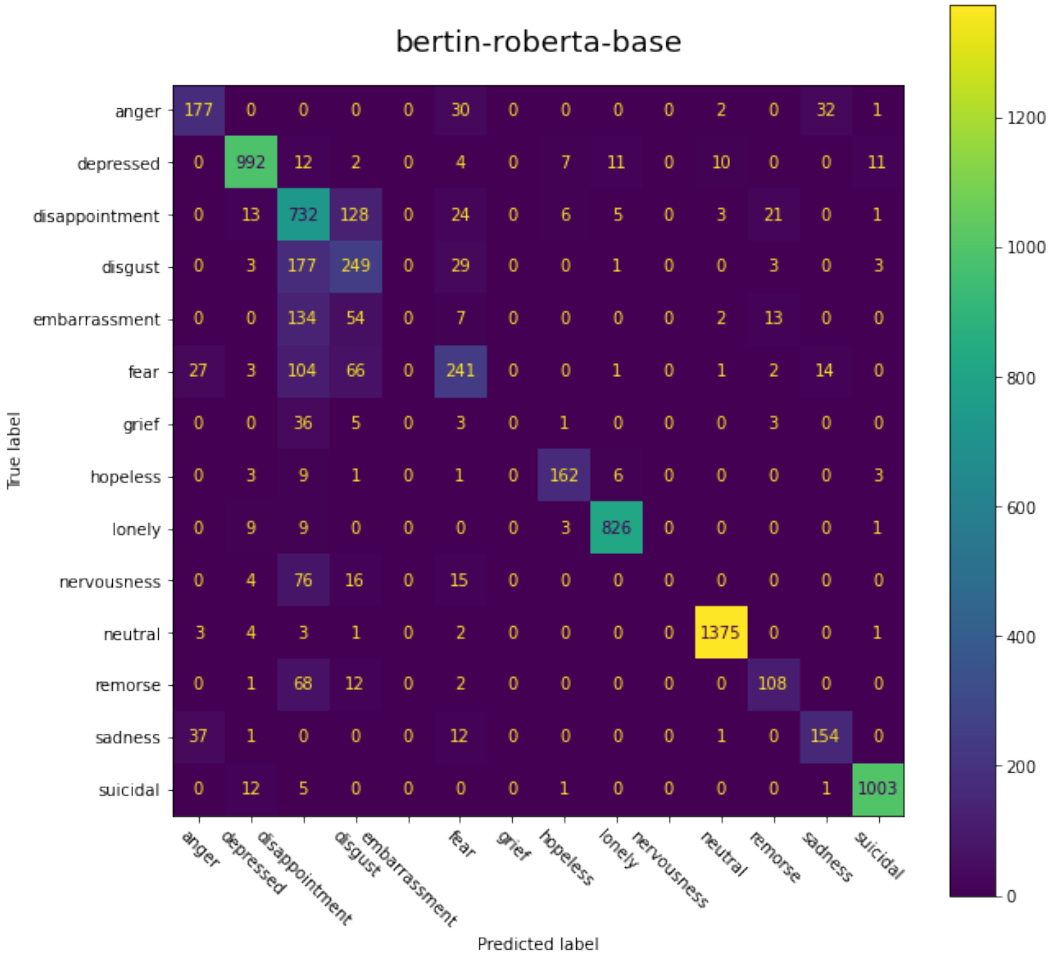


Figura 6.2: Matriz de confusión Bertín.

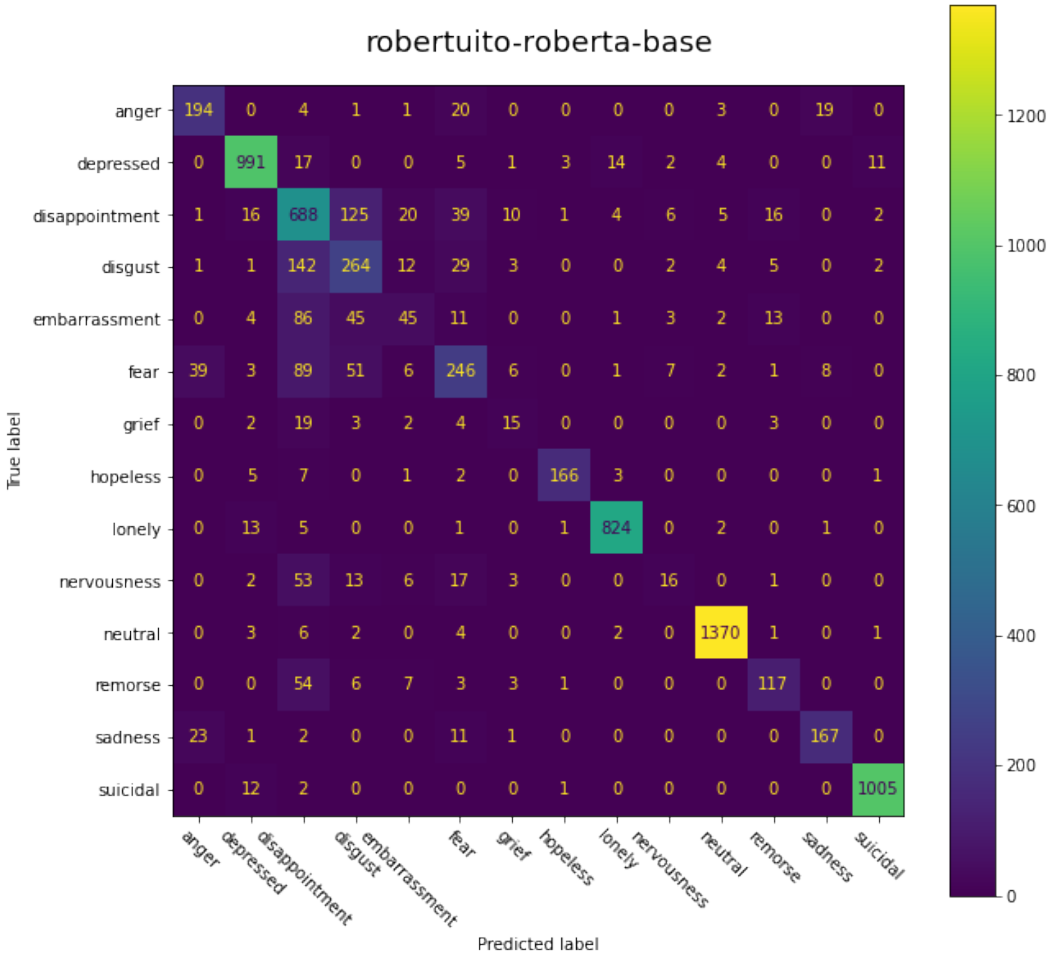


Figura 6.3: Matriz de confusión Robertuito.

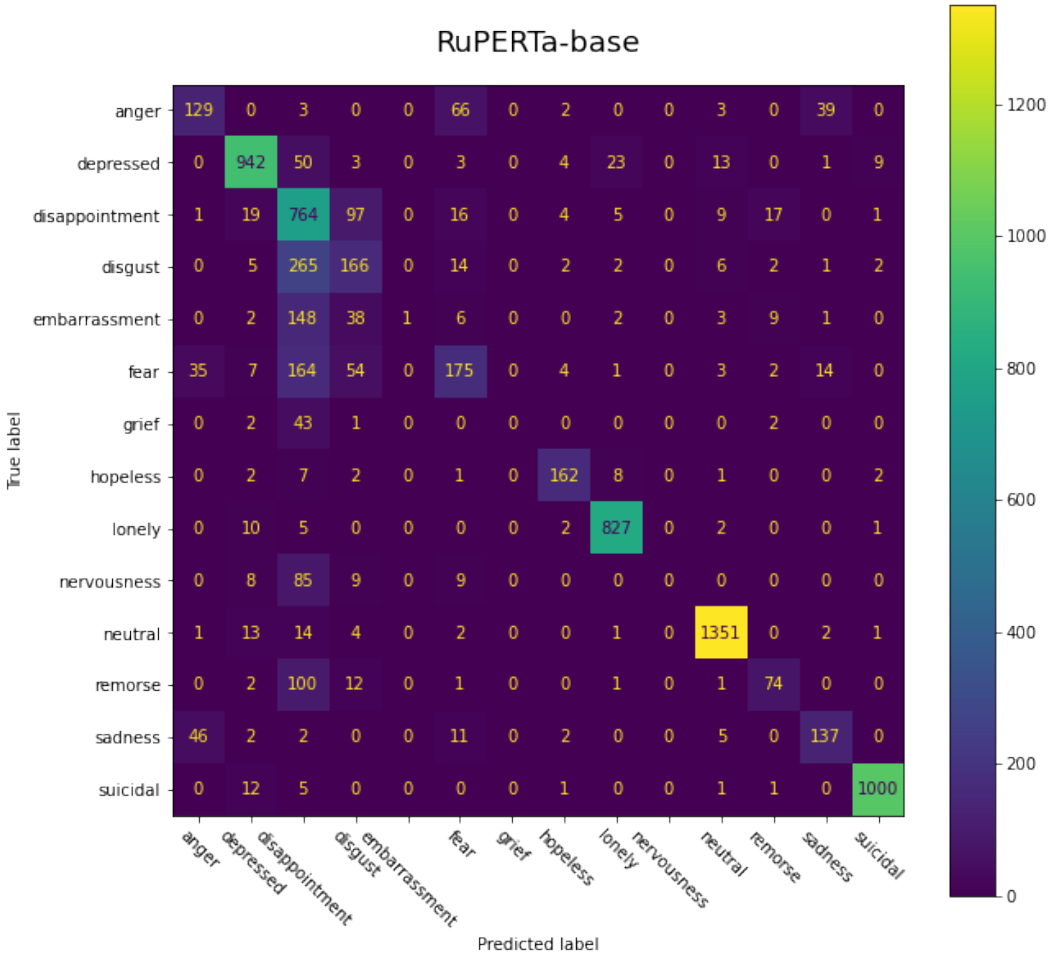


Figura 6.4: Matriz de confusión RuPERTa.

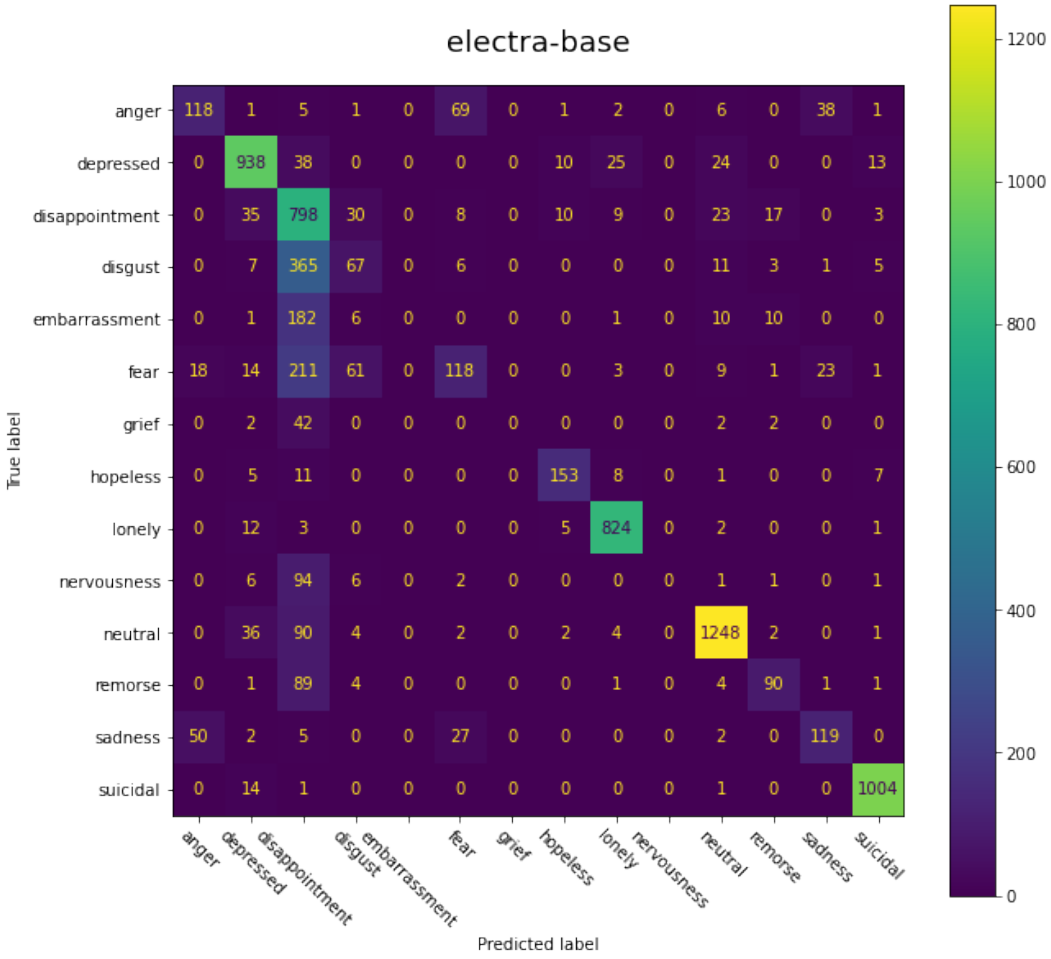


Figura 6.5: Matriz de confusión Electra.

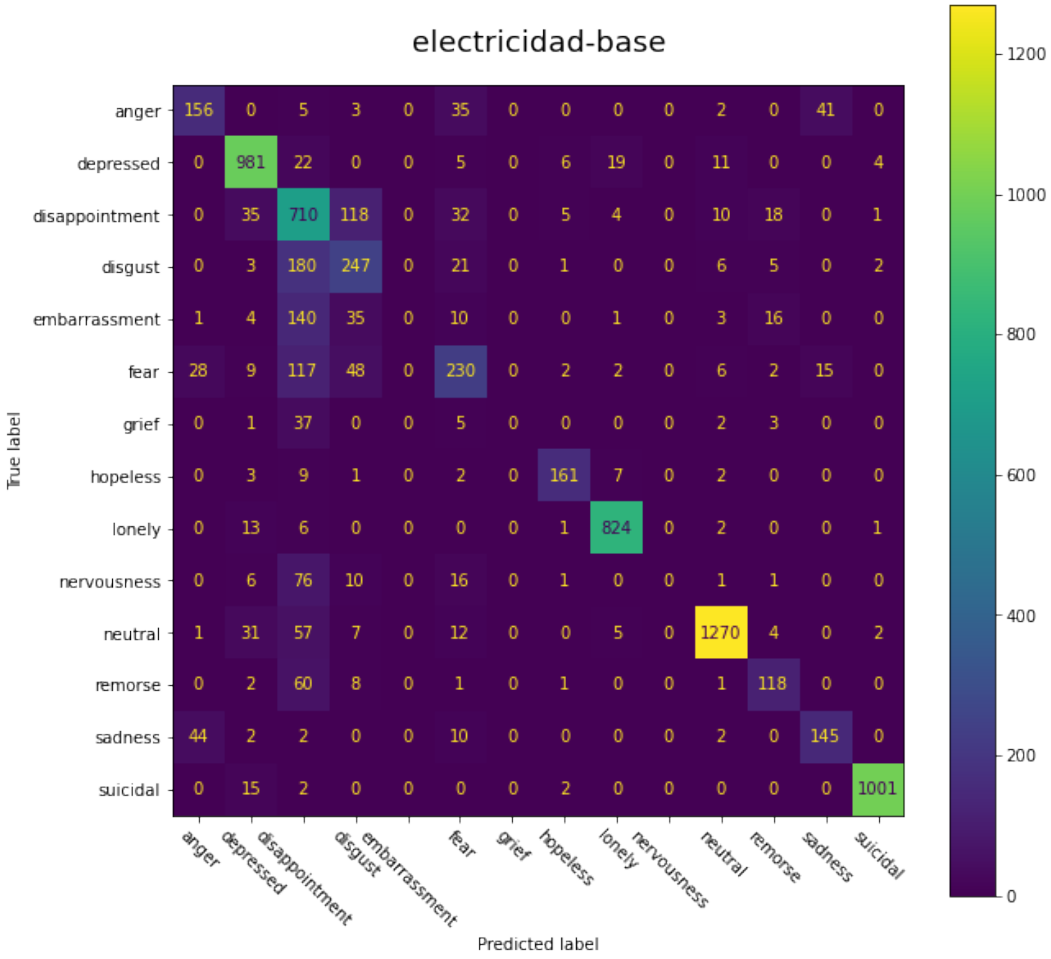


Figura 6.6: Matriz de confusión Electrididad (Electra ESP).

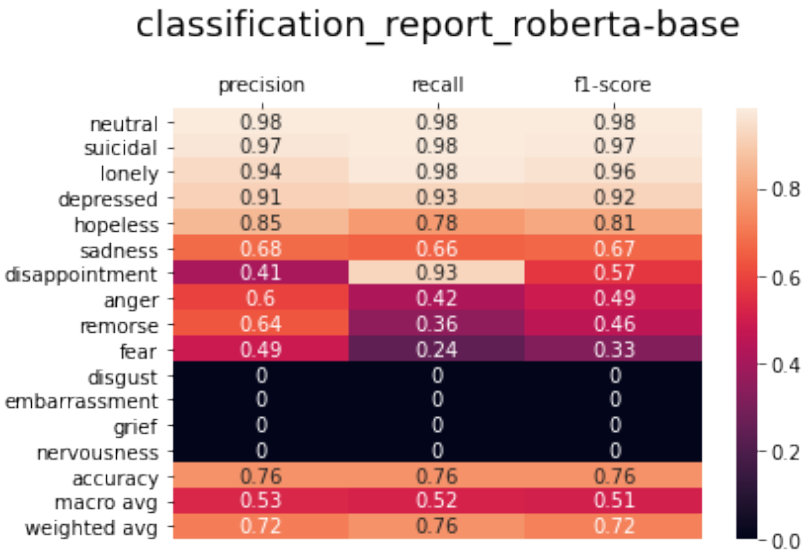


Figura 6.7: Reporte de clasificación para RoBERTa-base.

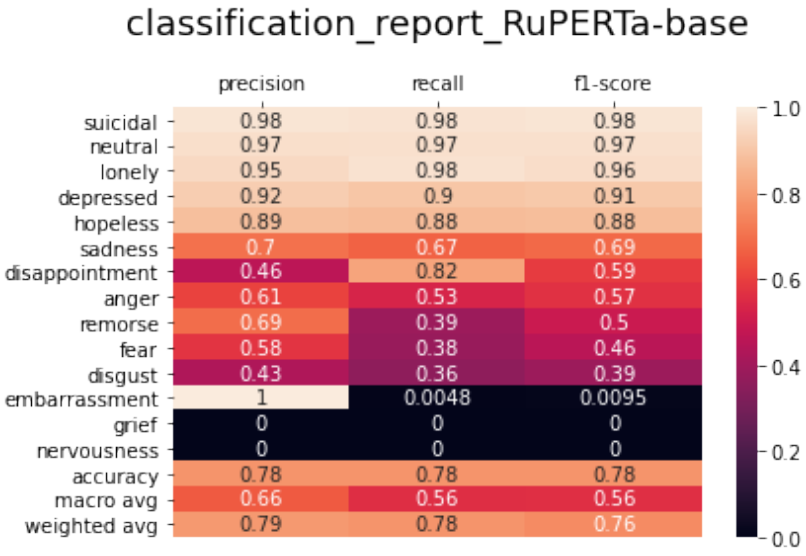


Figura 6.8: Reporte de clasificación para RuPERTa.

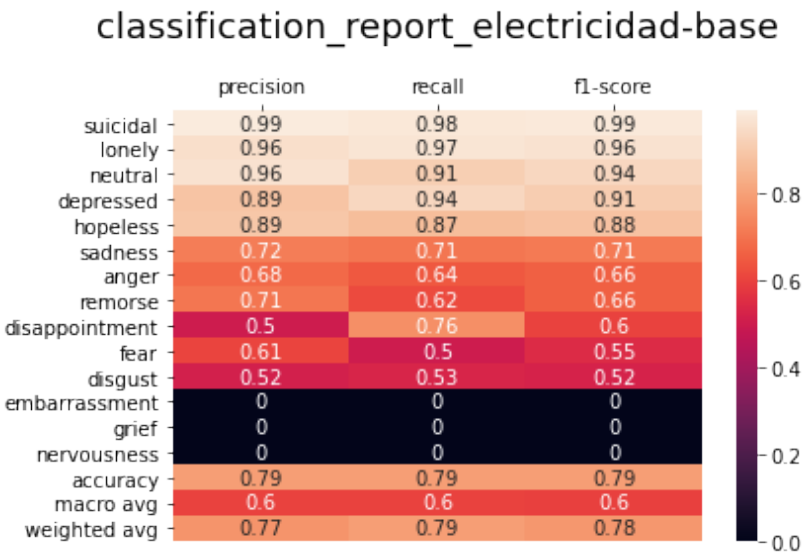


Figura 6.9: Reporte de clasificación para Electricidad.

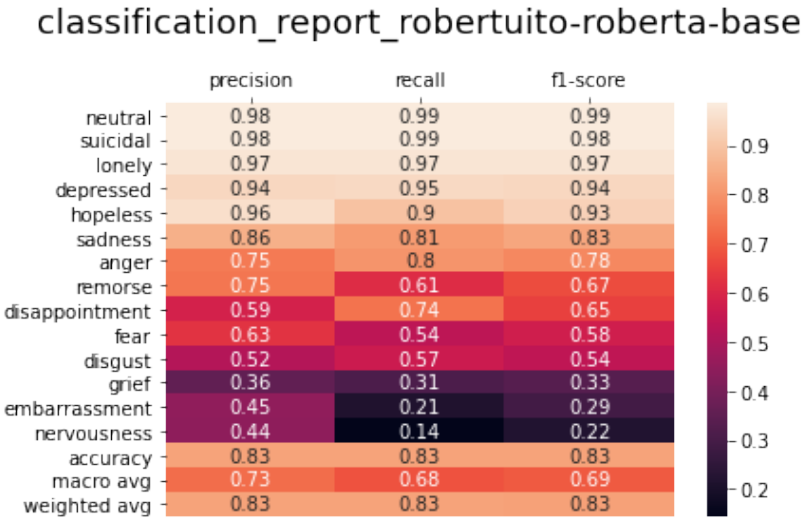


Figura 6.10: Reporte de clasificación para Robertuito.

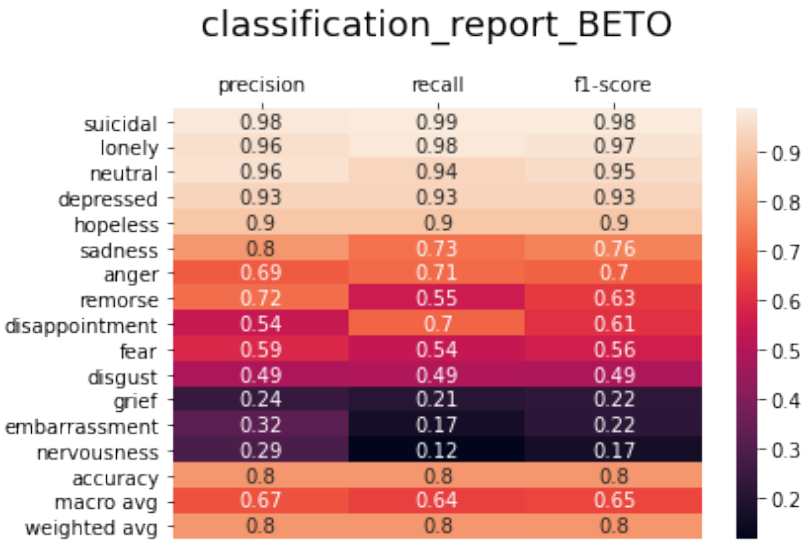


Figura 6.11: Reporte de clasificación para BETO.

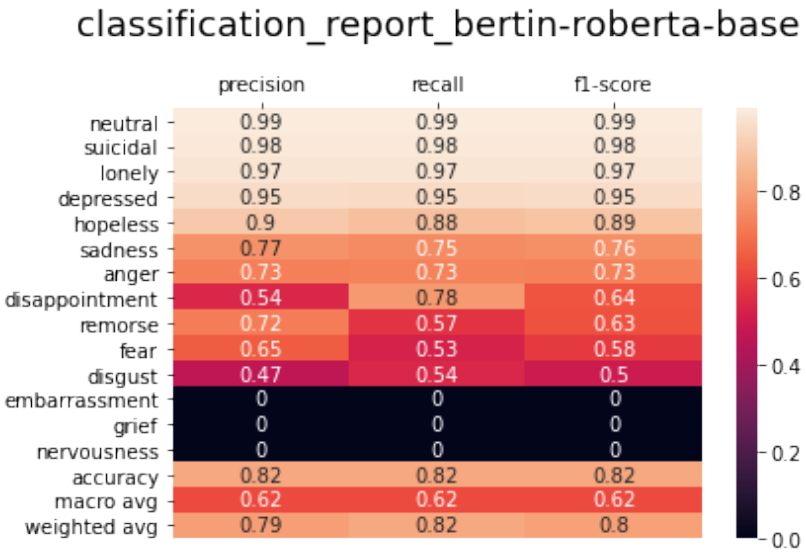


Figura 6.12: Reporte de clasificación para Bertín.

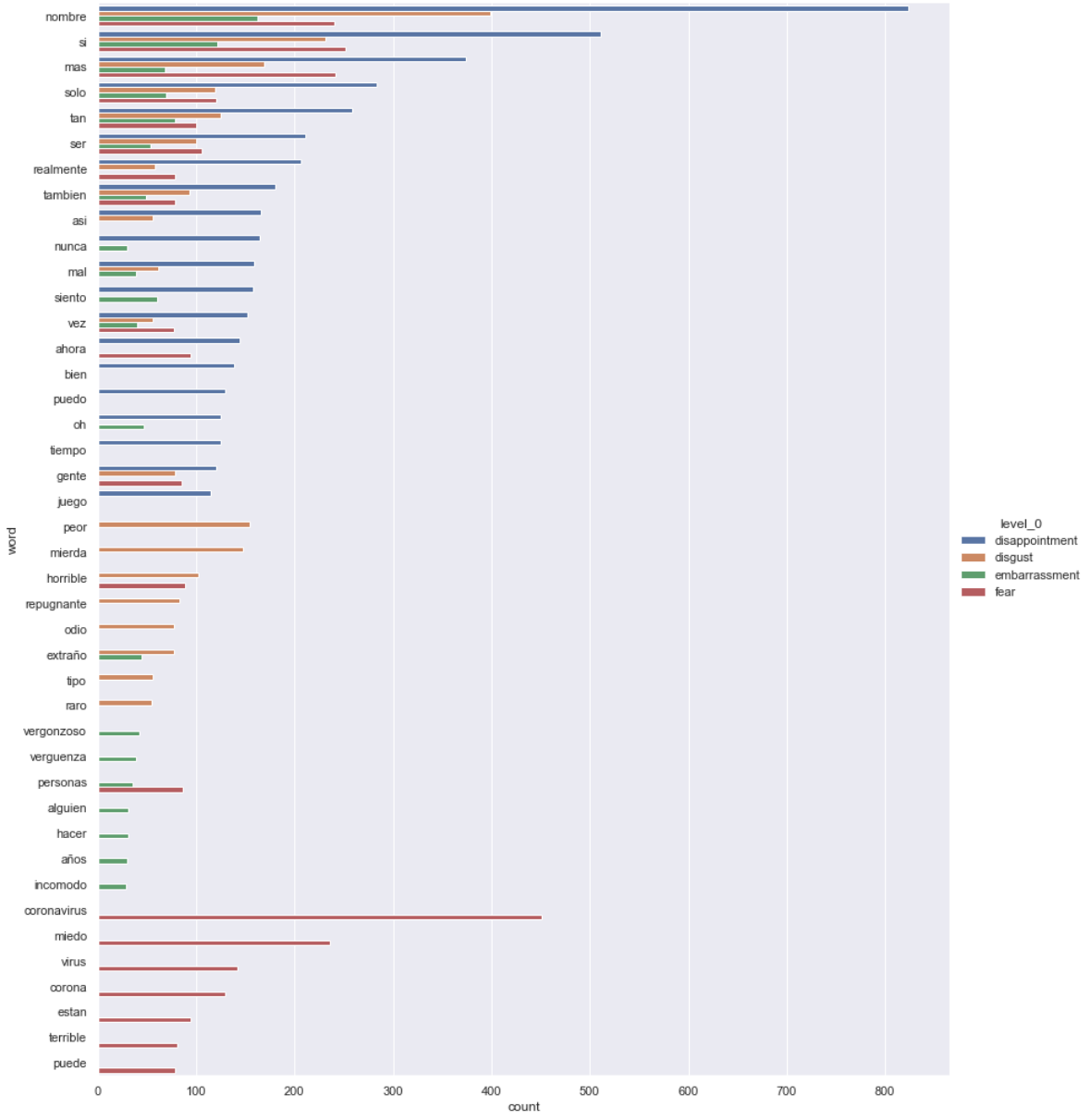


Figura 6.13: Frecuencia de aparición de términos completa.

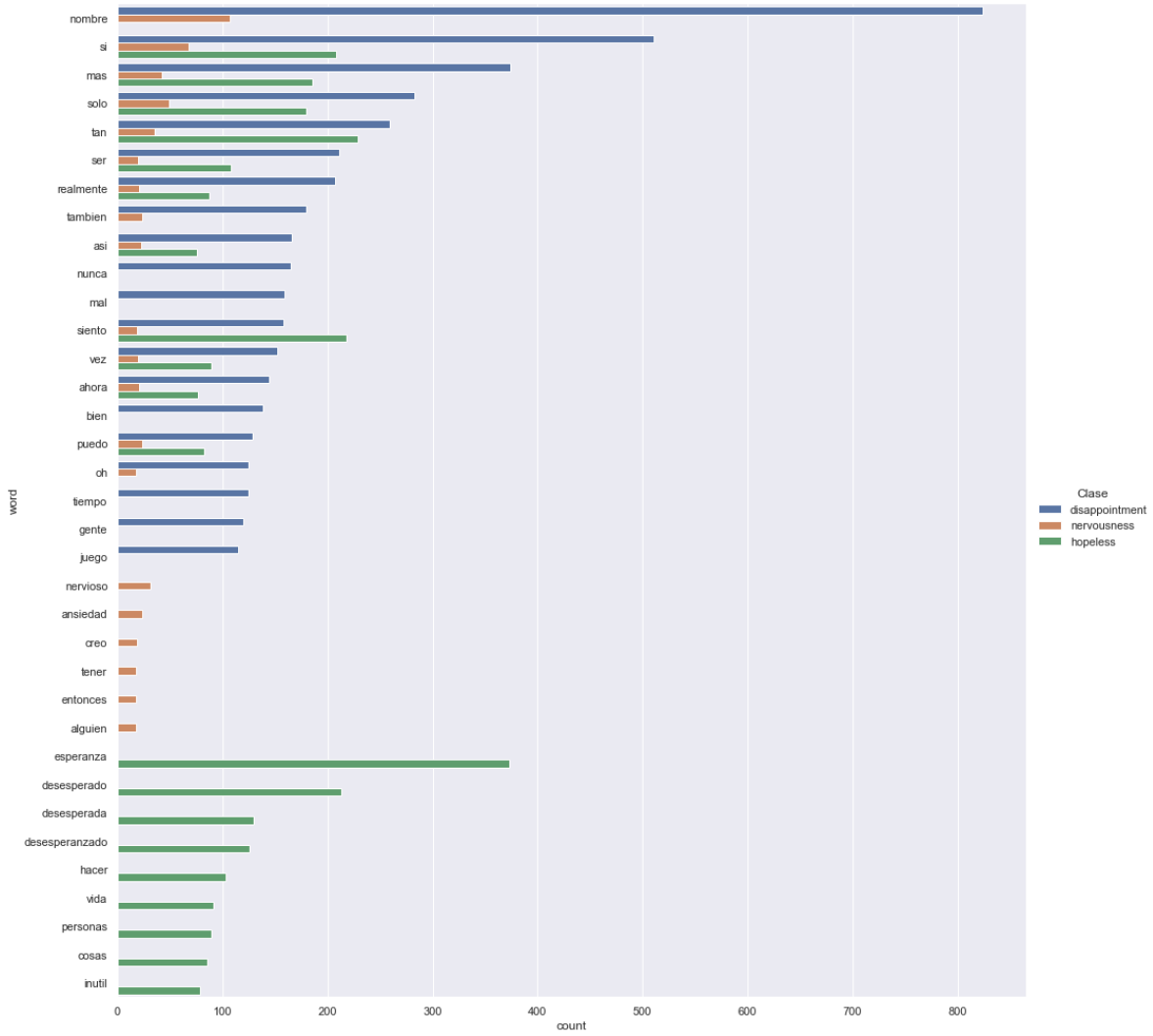


Figura 6.14: Frecuencia de términos completa (disappointment, hopeless y nervousness).