www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 16(4), Sep-Dec 2025, 515-526. ISSN: 2007-1558. https://doi.org/10.61467/2007.1558.2025.v16i4.574

Measuring the Performance of Large Language Models with Hyperparameters Calibration and Machine Learning Approaches for Sentiment Analysis in Spanish Texts

Mario A. Cruz[1], José A. Reyes-Ortiz[1,], Josué Padilla-Cuevas[1] and Leonardo D. Sánchez-Martínez[1]*
1 Systems Department, Autonomous Metropolitan University, Azcapotzalco, Mexico City 02200, Mexico.
al2221800024@azc.uam.mx, jaro@azc.uam.mx, jpc@azc.uam.mx, ldsm@azc.uam.mx

Abstract. Social networks have become an important source of information in recent years, offering a platform for visualizing people's opinions on various industries and research topics. These opinions may be expressed in Spanish, providing fresh and updated insights that can be analyzed using computational approaches. This research aims to understand the sentiment conveyed in ideas or opinions within Spanish text. As a result, various methods are evaluated to identify the most effective approach. Computational techniques, especially Natural Language Processing (NLP), have become essential for automating this analysis. While traditional machine learning algorithms have been employed for sentiment analysis, the rise of large language models since 2017 has introduced significant challenges in assessing their impact on various NLP tasks. This paper presents a comparison between machine learning approaches and calibrated large language models for sentiment analysis in Spanish texts, measuring their performance. The calibration process consists of two stages: a coarse calibration using exploratory methods and a fine calibration that involves an algorithm for searching hyperparameter values. The evaluation process showed that the ost effective machine learning approach combines unigrams and bigrams with the Bayes algorithm, along with exploratory parameter tuning and feature selection, achieving an accuracy of 72.72% and an F1 score of 72.81%. Furthermore, by applying LoRA, a technique that optimizes the fine-tuning of pre-trained models, it was found a best model that we call and store as twitter-xlm-roberta-base/SentUAM, which reached an accuracy of 71.99% and an F1 score of 71.78% in the sentiment analysis of Spanish texts.

Article Info
Received November 29, 2024
Accepted December 4, 2024

1 Introduction

The enormous amount of information generated daily on social media platforms has become a valuable source for understanding social opinion. This phenomenon has sparked significant interest in both industry and research. Unstructured texts from these platforms are particularly relevant for analyzing expressed sentiment. However, manual analysis of such texts is daunting, which drives the need for computational techniques like Natural Language Processing (NLP) to perform automatic sentiment identification. Sentiment analysis has been a significant focus of research in recent years, with its ability to determine text polarity being applied in various fields, including product reviews, social media monitoring, customer service, market analysis, and politics. While machine learning algorithms have been used to tackle this task, advancements in NLP through Large Language Models have further enhanced its effectiveness. This manuscript aims to evaluate the performance of both machine learning approaches and calibrated Large Language Models, specifically BERT, for sentiment analysis in Spanish texts. It is important to note that analyzing sentiment in Spanish texts introduces additional challenges due to higher lexical, syntactic, and semantic ambiguity levels compared to languages like English.

The complete process in this work involves collecting from the literature a set of texts labeled with sentiment categories and extracting word-level features through lemmatization and stemming techniques, as well as generating n-grams (unigrams and bigrams) to weigh them as inputs for machine learning algorithms. Additionally, hyperparameter calibration is performed on

four BERT-based language models, using the set of texts in their original form, i.e., without any preprocessing, to conduct a comparison. The calibration process was carried out in two ways: on the one hand, a coarse calibration using an exploratory method and on the other, a fine calibration based on an algorithm for the search for hyperparameter values. Finally, both approaches are evaluated using accuracy and F1 metrics to obtain the best algorithm and configuration based on their performance. This research allows us to compare how Large Language Models (LLM) perform in sentiment analysis compared to traditional machine learning methods. We want to determine if these models can outperform conventional approaches regarding accuracy and effectiveness. This comparison not only helps us better understand how LLMs work in this task but can also significantly impact the development of future technologies based on these models. It is crucial to understand the strengths and limitations of LLMs in sentiment analysis to use them effectively in practical scenarios. Furthermore, the results of this research will lay the groundwork for future studies in the field of sentiment analysis and natural language processing. These findings could serve as a starting point for further research on improving LLMs and integrating them with other machine learning techniques in practical applications.

Through experimentation, it was found that combining n-grams with feature selection and the Bayes algorithm, along with exploratory parameter tuning, yielded the best results, achieving 72.72% accuracy and 72.81% F1 score. Feature selection significantly improved the metrics by reducing term dictionary variability and selecting variables that enhance model effectiveness. Additionally, lemmatization and stemming also improved the metrics for all algorithms, highlighting their importance in sentiment analysis. Our twitter-xlm-roberta-base/SentUAM model stood out as the best performer due to its pretraining on tweets, which allowed it to adapt effectively to the Cardiff text set. This experiment, conducted using LoRA, achieved 71.99% accuracy and 71.78% F1 score, closely approaching the results obtained with unigrams+bigrams, feature selection, and the Bayes algorithm.

The rest of the paper is organized as follows. Section 2 presents a state-of-the-art study on large language models and machine learning approaches for NLP tasks such as Sentiment Analysis. Section 3 outlines the proposed methodology. Section 4 outlines the text corpus used. Section 5 presents the NLP approaches lemma, stemming, n-grams, and the machine learning algorithms used for sentiment analysis. Section 6 presents fine-tuning using large language models through exploratory hyperparameter calibration, adapting the model to the text corpus and performance optimization of the models. The results of the experiments are detailed in Section 7, followed by an analysis and discussion in Section 8. Finally, the conclusions derived from this study and potential areas for future research are presented in Section 9.

2 Related Work

In recent years, studies have emerged comparing different NLP techniques for sentiment analysis, such as (Chiruzzo, 2020), where a classification of neutral sentiment was conducted using the "SentimentAnalysisatSEPLN (TASS) Spain, Costa Rica, Peru" competition corpus. This work employed bag-of-words techniques, word polarity, and category markers. Results using the accuracy metric were 56% for SVM, 52.2% using CNN, and 52.1% in LSTM. Similarly, (Samuel, 2020) carried out a classification using 170,000 tweets related to COVID-19 downloaded from the Kaggle website. Naive Bayes and logistic regression algorithms were used for short tweets (less than 77 characters) and long tweets (more than 77 and less than 120 characters), resulting in Naive Bayes achieving 91% accuracy for short tweets and 57% for long tweets, while logistic regression achieved 74% for short tweets and 52% for long tweets.

Recently, with the advent of large language models (LLMs), these models have begun to be used for NLP tasks, comparing results with machine learning algorithms. (Chintalapudi, 2021) compared BERT with logistic regression models, support vector machines, and LSTM using tweets from Indian users during COVID-19. BERT achieved 89% accuracy, surpassing the other three models, which produced 75%, 75%, and 65%, respectively. Despite the emergence of new techniques, machine learning algorithms combined with NLP techniques continue to yield good results for sentiment analysis. For example, (Indulkar, 2021) compiled 3000 opinion tweets about the Uber and Ola taxi services with two classes (positive and negative) using Google Word2Vec for feature extraction. The best algorithm was Random Forest, with an accuracy of 96.3% for Uber and 83.4% for Ola. (Bhargav, 2021) collected satisfaction opinions on a product and performed cleaning and preprocessing, including replacing emojis with *EMO POS/EMO NEG*. Feature extraction was done using unigrams and bigrams, resulting in 76% precision for Random Forest, 75.89% for decision trees, and 81.97% for SVM method. (Yadav, 2021) conducted sentiment analysis using a dataset of labeled tweets from Kaggle. Naive Bayes, logistic regression, and support vector machine algorithms were used.

Logistic regression achieved F1 scores of 82.69% for negative and 82.246% for positive labels. In contrast, Naive Bayes achieved 81.46% for negative and 79.67% for positive labels, and the support vector machine achieved 88.88% for negative and 74.27% for positive labels. (Kaur, 2021) performed sentiment analysis using a set of IEEE Data Port texts, approximately over 500 tweets, implementing a Hybrid Heterogeneous Support Vector Machine (H-SVM), neural network, and support vector machine. Results showed 69% accuracy for SVM, 86% for MKH-SVM, and 72% for the neural network. Regarding the F1 metric, results were 57% for SVM, 77% for MKH-SVM, and 48% for the neural network. SVM obtained 49% recall, MKH-SVM 69%, and the neural network 37%.

COVID-based datasets have been an essential source for sentiment analysis tasks. (Alam, 2021) studied using the "All COVID-19 Vaccines Tweets" text corpus from Kaggle with 125,906 texts, applying VADER sentiment analyzer, resulting in 90.83% accuracy for Bi-LSTM and 90.59% for LSTM models. A study by Hidayat and (Sulistiyono, 2022) collected 3,588 data points for positive and negative sentiment of COVID increase, achieving 80% accuracy using Bayes with K-Fold Cross-Validation and 85% using SVM. (Parikh, 2022) used a dataset of 1,600,000 texts evenly divided into positive and negative, performing tokenization, lemmatization, and removal of stopwords, extracting features with TF-IDF and applying Word2Vec, using deep learning algorithms for classification and comparing with Bayes and support vector machines. (AlBadani, 2022) proposed a sentiment analysis approach on US airline tweets combining the ULMFiT technique (Universal Language Model Fine-tuning for Specific Tasks) with a neural network for text feature extraction and SVM for classification, achieving 99.78% accuracy and 99.01% F1 score. (Singh, 2022) conducted sentiment analysis of COVID-19-related reviews on Twitter using a deep learning approach based on an enhanced long short-term memory (LSTM-RNN) with attention layers for feature weighting, achieving 63.12% accuracy and 63% F1 score.

Recently, some works have focused on combining deep learning with various techniques to explore the effectiveness of sentiment analysis. In the work by (Sun, 2022), sentiment analysis was presented using a corpus of 209,000 messages generated between the system and the patient in English. They applied VADER (Valence Aware Dictionary and Sentiment Reasoner) and BERT, obtaining results of 72%, 82%, and 85% at three levels of fine-tuning for BERT. On the other hand, (Ashrita, 2023) proposed an enhanced model of LSTM with a bidirectional mechanism (BiLSTM) in conjunction with word2vec, and it is compared against the LSTM model, yielding accurate results of 94% for BiLSTM and 89% for LSTM. Also, a study conducted by (Xu, 2023) compared methods for sentiment analysis through traditional machine learning and deep learning. Various models were evaluated, such as SVM, RNN, Bayesian, CNN, LSTM, and BERT. Experimental results showed that combining BERT with CNN outperformed other models in text sentiment classification, with accuracy results of 98.34% and F1 score of 97.25%. Meanwhile, a study to enhance sentiment analysis in social media texts was conducted by (Matías-Cristóbal, 2023) using a hybrid model of deep learning and natural language processing to detect depression in social media texts. CNN and LSTM-RNN models were applied, and word embedding techniques such as Word2Vec, Glove, and FastText were used, achieving the best results with FCL (Fast-text Convolution Neural Network with Long Momentary) with an accuracy of 88% and F1 of 87% with a set of Twitter texts.

3 Proposed Methodology

This section presents the proposed methodology for conducting sentiment analysis on Spanish texts from social media. This general approach has five stages, as shown in Figure 1. Each stage is described below, followed by a detailed explanation in the subsequent sections.

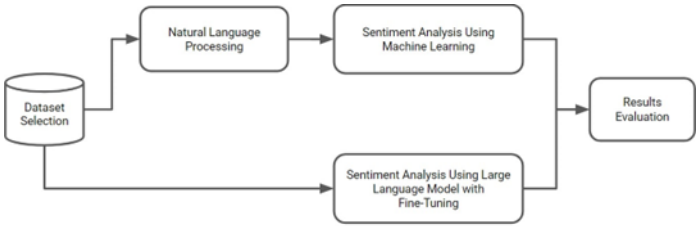

Figure1. Overview of the proposed methodology

Collection and Analysis of the Text Dataset. In the initial phase, various Spanish text datasets are explored and analyzed, classified into sentiment categories extracted from the literature. The objective is to identify a dataset that presents significant challenges in the classification task while maintaining a balance in the number of sentiment categories. In the subsequent stages, these texts serve as inputs for natural language processing techniques and large language models. The texts processed with NLP

are then applied to machine learning algorithms. Natural Language Processing. Once the text dataset is selected, various NLP techniques such as cleaning, lemmatization, stemming, and n-gram generation are implemented for extracting feature vectors to represent the texts in a structured manner using numerical values. Implementation of Machine Learning Algorithms. During this stage, machine learning algorithms are employed to analyze the generated feature matrix. Fine-Tuning of Various Large Language Models. Fine-tuning of various large language models is performed, adapting them for the classification task and calibrating hyperparameters to address the classification of the original texts. Evaluation and Model Comparison. The generated models undergo a rigorous evaluation using Accuracy and F1 metrics. Through a comprehensive comparative analysis, we identify the approaches that truly stand out in sentiment analysis. This thorough evaluation ensures that our methodology is effective and reliable, providing the confidence necessary for sentiment analysis tasks.

3.1. Dataset

A search and selection of texts in Spanish literature was conducted to apply the machine learning algorithms Bayes, SVM, and Logistic Regression and fine-tuning four large language models to compare the results. In this work, the text set *tweet-sentiment-multilingual* from Barbieri (2021), whose Spanish texts come from Díaz Galiano (2018), was used, which will be referred to simply as "Cardiff" hereafter, as it was created by the Cardiff NLP group at Cardiff University. This dataset is publicly accessible via CardiffNLP (2023) T*weet Sentiment Multilingual*, Hugging Face. Retrieved from https://huggingface.co/datasets/cardiffnlp/tweet_sentiment_multilingual. The text set contains different languages, such as French, English, and Spanish; for this study, only those corresponding to Spanish were used. The set consists of 3033 user messages and sentiment labels, divided into three files. Therefore, all files were unified, and a filter was applied to select only the texts in Spanish. A sample of the resulting text set is shown in Figure 2, with the "text" column containing the message and the "label" column containing values of "0" for negative, "1" for neutral, and "2" for positive sentiment.

texto	etiqueta
@user jajajaja dale, hacete la boluda vos jajaja igual a vos nunca se te puede tomar en serio te mando un abrazo desde Perú!	0
cada vez que cito un tweet se va la ubicación sin tampoco poder ponerla en el momento a uds les pasa? TE VEO Y ME PICA VICICONTE	1
@user MAAAAE RAJADO! Pero lo bueno es q uno se va independizando!y logrando metas	2
Bueno hoy fui a almorzar a Nanay con otras 3 dras xq la capacitación mal organizada no nos dió almuerzo y encima nos mandan a comer 2pm	0
Necesito seguir a mas cuentas camren shippers y fans de las armonías. Me recomendais alguna?	1
@user ¡Hola Tomás! ¿Habéis visto los nuevos #dinos de #TierraMagna? Es normal que haya colas antes de que comience el espectáculo	2

Figure 2. Samples of the Cardiff text dataset

It is observed that the set is balanced in terms of class distribution, the total number of records per label for this text set is shown in Table 1.

Table 1. Class distribution.

Class	Type	Total Tex
0	Negative	1011
1	Neutral	1011
3	Positive	1011

Natural Language Processing

This section outlines the Natural Language Processing tasks performed to process the Spanish texts and prepare them for subsequent stages. The tasks included text cleaning, lemmatization, n-gram generation, and feature extraction, all aimed at producing suitable input for machine learning algorithms and large language models.

Text Processing

The text was cleaned using a proposed approach and then restored to its base form using precise lemmatization and stemming techniques. Generating n-grams was applied to create a unified set of unigrams and bigrams from the clean texts.

Cleaning

Furthermore, a cleaning process of the texts was conducted by removing any elements that were considered unnecessary for the analysis task. In this study, the following elements were removed from the text set.

Mentions of @user.
Special characters, such as: ,\.\$*¿\?¡!\+\-[])(.
Words consisting of a single character or multiple repeated characters.
Emoticons, URLs, and numbers.
Multiple blank spaces, leaving only one.
Email addresses.

In informal texts, such as those found on social media, emoticons can be an important part of emotional expression, so their removal could negatively affect text interpretation. However, in approaches focused solely on textual content, such as in this study, emoticons may be considered noise, and their removal could improve result quality.

Lemmatization and Stemming

Lemmatization and stemming techniques have the same goal of simplifying words, but they differ in their approach. Lemmatization considers the canonical form of words, while stemming removes affixes. Both techniques are used to reduce variability in words and decrease the complexity of the text set. In this study, a procedure was used to bring the words to their base form by applying lemmatization and stemming techniques. Figure 3 illustrates the text set after applying the lemmatization technique.

text	label
jajajaj dal hacet la bolud vos jajaj igual a vos nunc se te pued tom en seri te mand un abraz desd peru	0
cad vez que cit un tweet se va la ubic sin tampoc pod pon en el moment a uds les pas te veo y me pic vicicont	1
maaaa raj per lo buen es q uno se va independizandoy logr met	2
buen hoy fui a almorz a nanay con otras dras xq la capacit mal organiz no nos dio almuerz y encim nos mand a com pm	0
necesit segu a mas cuent camr shippers y fans de las armon me recomendais algun	1

Figure 3. Samples derived from Figure 2 after undergoing the cleaning and stemming processes.

Generation of n-grams

N-grams refer to sequences of words extracted from a text, which can comprise individual words (unigrams), pairs of consecutive words (bigrams), or triples of consecutive words (trigrams). After cleaning the texts, this project employs unigrams and bigrams as features in a single set. The unigrams and bigrams are considered features based on the length of the collected texts. Figure 4 demonstrates how unigrams and bigrams are derived from clean text.

```
Processed Text (without stopwords): jajajaja dale hacete boluda vos
jajaja igual vos nunca puede tomar serio mando abrazo perú


Generated Unigrams in the Row 1: ['abrazo', 'boluda', 'dale',
'hacete', 'igual', 'jajaja', 'jajajaja', 'mando', 'nunca', 'perú',
'puede', 'serio', 'tomar', 'vos']


Generated Bigrams in the Row 1: ['abrazo perú', 'boluda vos', 'dale
hacete', 'hacete boluda', 'igual vos', 'jajaja igual', 'jajajaja
dale', 'mando abrazo', 'nunca puede', 'puede tomar', 'serio mando',
'tomar serio', 'vos jajaja', 'vos nunca']
```

Figure 4. Example of unigrams and bigrams.

Feature Weighting

The feature weighting method used in this work is based on the Term Frequency-Inverse Document Frequency (TF-IDF) technique, which determines the importance of words in a set of documents. Generally, TF measures the frequency of a word in a specific document, while IDF measures the rarity of a word in the document set. A matrix with TF-IDF weights is obtained by using lemmatization, stemming, and the combination of unigrams and bigrams, where a row represents each processed text. Each feature in this matrix is assigned real values that reflect its specific weight. The number of features generated by each of these characterization approaches is detailed in Table 2.

Table 2. Total number of terms generated by each NLP technique.

Type	Total Tex
Lemmatization	7122
Stemming	5749
Unigrams + Bigrams	27249

Feature Selection

Feature selection was conducted to identify the most relevant variables and compare the results obtained with and without this selection. (Fodor, 2002), it is mentioned that in many cases not all measured features are important in high-dimensional datasets, so it is advisable to reduce the dimensionality of the original data before any modeling. To address this problem, feature selection was performed to identify the most relevant elements in order to compare the results obtained with and without this selection. The SelectKBest library (Atkinson-Abutridy, J. 2023) was used to choose the terms, which selects features based on statistics to assess the relationship between each feature and the target variable, in order to reduce the dimensional space and select the most relevant features to improve the model's capacity. Empirical tests were conducted starting with a selection of 10,000 features of unigrams and bigrams, decreasing the quantity by 1,000 in each test. For lemmatization, the process began with 7,000 features, while for stemming, it started with 5,000. In all cases, performance improved until reaching 1,000 features. Starting from 1000 features, experiments were conducted by reducing 200 features in each test, observing that this reduction deteriorated performance. Therefore, it was decided to utilize 1000 features, incorporating lemmatization, stemming, and unigrams+bigrams.

Application of Machine Learning Algorithms

In this work, training was carried out using three machine learning algorithms selected based on the initial literature. The algorithms were initially trained with default parameters, followed by exploratory adjustment with manual calibration through trial and error as presented in (Birattari & Kacprzyk, 2009). The text set used in this process was previously processed using lemmatization, stemming, and unigrams+bigrams techniques, with both feature selection and no feature selection being considered.

The description of the parameters is detailed as follows:

1 Model Type (Bayes): Refers to the multinomial distribution, which is suitable for classification problems with multiple categories.
2 C (SVM): Regularization parameter controlling the trade-off between maximizing the margin and minimizing classification error.
3 Gamma (SVM): Defines the range of influence of a single training example.
4 Kernel (SVM): Specifies the type of kernel function used, such as 'rbf' (radial basis function).
5 Penalty (Logistic Regression): Type of regularization applied, such as L1 (lasso) or L2 (ridge).
6 Solver (Logistic Regression): Optimization algorithm used, such as 'lbfgs'.

Sentiment Analysis Using Large Language Models

This section provides an overview of the Large Language Models (LLM) used for sentiment analysis in Spanish, namely BERT and RoBERTa. The experimentation was based on these models using the Cardiff text set. It is important to note that the original text set was used in this work without preprocessing. The selected LLMs have specific adaptations and improvements on their base architectures to optimize performance in particular tasks. Below are the four selected models with a brief description of their characteristics:

1 bert-base-multilingual-uncased: This model is based on BERT and was trained on 102 languages using Wikipedia (Hugging Face, n.d.)..
2 roberta-base-bne: This model is based on RoBERTa. It was trained with a 570 GB Spanish corpus of text collected by the National Library of Spain between 2009 and 2019 (PlanTL-GOB-ES, n.d.).
3 twhin-bert-base: This model is also based on BERT and was trained with over 7 billion tweets from over 100 different languages (Twitter, n.d.).
4 twitter-xlm-roberta-base: This model is based on XLM-Roberta, a variant of RoBERTa. It was pre-trained with approximately 198 million multilingual tweets (CardiffNLP, n.d.).

The large language models were fine-tuned in three ways: task-specific fine-tuning for sentiment analysis through text classification, hyperparameter calibration using an exploratory technique, and fine calibration to improve the performance of the models using the LoRA technique. These model-tuning approaches are described in detail below.

Task-specific fine-tuning

The fine-tuning based on the task involves adapting the selected base models by modifying the LLM architecture and adding a final layer to suit the new task. The steps involved in fine-tuning the models are as follows:

1 Remove the "fill mask" head from the LLM, as it is irrelevant to the task performed in this work.
2 Incorporate the classification task into the LLM by adjusting the model architecture for text classification.

The model obtained from this task is further trained through exploratory hyperparameter tuning, which is described in the following section.

Coarse Hyperparameter Calibration of Large Language Models

In the context of large language models, hyperparameters are external parameters that affect the model's performance. They are not learned from the training set but can be adjusted to influence the model's capacity and behavior. This work explored various hyperparameter configurations through an exploratory method presented in (Birattari & Kacprzyk, 2009) to identify the suitable values. The following hyperparameters were focused on in this work, along with their default values:

1 Batch Size: the number of data samples used in each training iteration. A batch is a set of training examples processed simultaneously by the model before updating the weights. A larger batch size can speed up training but may require more memory. This hyperparameter is established with 8 as a default value, and no other range or value was explored.
2 Epochs: an epoch is a complete iteration through the entire training set. During one epoch, the model sees and processes all examples in the training set. The number of epochs was explored with values between 3 and 5.
3 Optimizer: an algorithm that adjusts the model's weights during training to minimize the loss function. The optimizers that were studied are: adamw\hf, adamw_torch_focused, adamw_torch, adafactor and adagrad.
4 Learning Rate: determines the size of steps taken during model optimization. The learning rate was used with the default value of 5e-5.
5 Weight Decay: a penalty applied to the model's weights. It helps prevent overfitting by adding a penalty term proportional to the magnitude of the weights in the loss function. The weight decay was used with the default value of 0.
6 Train test split: the number of training and testing elements are also considered hyperparameters of the LLM and were defined at fixed values using the complete text set for all experiments as follows: 80% for training (2426 texts) and 20% for testing (607 texts).

Fine Hyperparameter Calibration of Large Language Models using LoRA

A fine calibration using the Low-Rank Adaptation (LoRA) (Hu et al., 2021) technique was applied to optimize the hyperparameters to improve the performance of large-scale language models (LLMs) in sentiment analysis. LoRA is used to adapt the model weights efficiently by inserting low-rank matrices into the model architecture, allowing for fast and effective adaptation without the need for costly full adjustment of all model parameters. Unlike the model adjustments described in the previous section, which involve modifying hyperparameters, LoRA focuses on adapting the weights within neural networks. In this approach, the traditional hyperparameters of Large Language Models (LLMs) remain unchanged during the fine-tuning process. To carry out this adaptation, the Optuna (Akiba et al., 2019) library was used, which facilitates the search for optimal hyperparameters specific to the LoRA technique. Instead of adjusting default hyperparameters such as learning rate or weight decay coefficient, LoRA introduced hyperparameters related to the low-rank matrices. The optimized parameters included the size and configuration of the low-rank matrices.

4 Experimental Results

In this section, we will discuss the experiments conducted and the metrics used to compare machine learning approaches and large language models with fine-tuning. These experiments were executed in a Procesador 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz, RAM 16.0 GB, Graphical Card NVIDIA GeForce RTX 3050 Ti Laptop GPU 64 cores. For this work, we employed the Accuracy and F1 score metrics. This metric helps to indicate the proportion of correct predictions.

Results of machine learning approaches

The conducted experiments clearly demonstrate the differences in algorithm performance under various configurations of features and parameters. In particular, it is evident that feature selection has a significant impact on the analysis results. For instance, when comparing results without feature selection and with default parameters, a moderate performance in terms of precision and accuracy is observed. However, by applying feature selection and adjusting the parameters, a substantial improvement in algorithm performance is achieved, as evidenced by notable increases in F1 and accuracy metrics. These findings underscore the importance of a careful approach to feature selection and configuration to optimize the performance of machine learning models in sentiment analysis tasks.

Results of calibrated large language models

This section describes the experiments conducted to compare the accuracy and F1 metrics results of default hyperparameters and fine-tuning LLM. In addition, the new models created through fine-tuning in this study are labeled with the original name followed by *textit/SentUAM* to personalize and reference them as a contribution where the base model has been improved.

This work performs coarse calibration through various exploratory configurations by trial and error (Akiba et al., 2019) and an automatic model calibration using the Low-Rank Adaptation (LoRA) technique. The objective was to identify the best values for the hyperparameters and compare the results obtained with these configurations. The experiments conducted are detailed below:

1 Default parameters: The evaluation with the default hyperparameter configuration is presented to establish a baseline for comparison.
2 Exploratory calibration 1, 2, 3 and 4: Four experiments were conducted with coarse calibration of hyperparameters through an exploratory technique, varying optimizers and other configurations to assess their impact on model performance. The results of these experiments are summarized in the table below.
3 LoRA: The Low-Rank Adaptation (LoRA) technique was applied to adjust the model weights without changing default hyperparameters, aiming to improve model performance with a more efficient adjustment by introducing hyperparameters related to the low-rank matrices.

The number of training and test elements is also considered a hyperparameter and is defined with fixed values using the complete text set for all experiments as follows: 80% for training (2426 elements) and 20% for testing (607 elements). The values presented in Table 4 serve as a starting point for this calibration. From there, an exploratory search is conducted to identify values that optimize the performance of the LLMs. The initial experiment employed the *bert-base-multilingual-uncased* model with default parameters, achieving 49.25% accuracy and 49.40% F1 score. The results of fine-tuning experiments and the calibrated hyperparameters of the new model *bert-base-multilingual-uncased/SentUAM*.

For the *roberta-base-bne* model, default parameter results included an accuracy of 62.75% and an F1 score of 62.44%. Default parameters, four exploratory configurations and LoRA technique for the new model *roberta-base-bne/SentUAM*.

The *twhin-bert-base* model yielded default results of 62.43% accuracy and 61.72% F1 score. Results after experiments for the new model *twhin-bert-base/SentUAM* are presented in Table 3.

Table 3. Results of hyperparameter tuning by manual calibration and fine-tuning in *twhin-bert-base/SentUAM*

Hyperparameter tuning by manual calibration				Results	
Experiments	Batch	Epochs	Optimizer	Accuracy	F1
Default Parameters	8	3	adamw_hf	62.43	61.72
Exploratory 1	8	3	adamw_torch_fused	63.92	64.39
Exploratory 2	8	3	adamw_torch	62.76	62.74
Exploratory 3	8	3	adafactor	61.77	62.53
Exploratory 4	8	3	adagrad	63.09	63.60
LoRA	8	3	adamw_hf	65.50	63.47

Finally, the *twitter-xlm-roberta-base* model exhibited default results of 65.70% accuracy and 64.80% F1 score. The metrics for experiments with the new model *twitter-xlm-roberta-base/SentUAM* are shown in Table 4.

Table 4. Results of hyperparameter tuning by manual calibration and fine-tuning in *twitter-xlm-roberta-base/SentUAM*

Hyperparameter tuning by manual calibration				Results	
Experiments	Batch	Epochs	Optimizer	Accuracy	F1
Default Parameters	8	3	adamw_hf	65.70	64.80
Exploratory 1	8	3	adamw_torch_fused	67.71	67.67
Exploratory 2	8	3	adamw_torch	68.69	68.32
Exploratory 3	8	3	adafactor	67.05	67.21
Exploratory 4	8	3	adagrad	71.49	71.43
LoRA	8	3	adamw_hf	71.78	71.78

The best results for each LLM used in this study are depicted in Figure 5. The *twitter-xlm-roberta-base/SentUAM* model emerged as the best performer with experiment 4, yielding results as shown in Table 5.

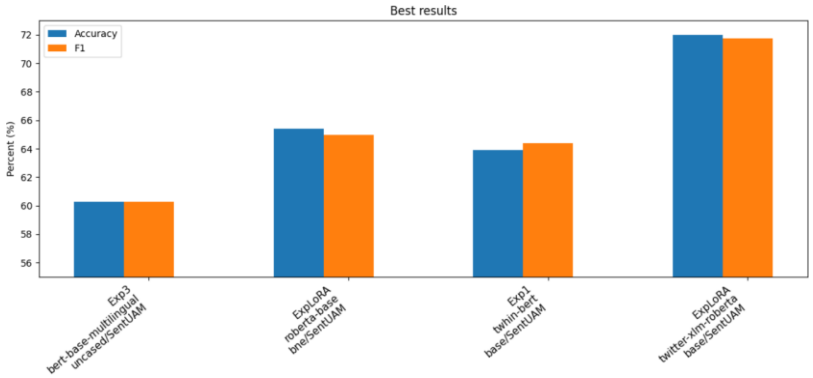

Figure 5. The best results for each fine-tuning large language model.

The experiments conducted with different LLM models and fine-tuning have yielded promising results. It is observed that careful parameter selection and manual calibration can significantly improve the models' performance. In particular, the last model leverages pretraining with tweets, allowing it to achieve better performance in terms of accuracy and F1. These findings underscore the importance of considering domain-specific pretraining when applying LLMs to sentiment analysis tasks.

Additionally, the LoRA technique has shown good performance, as its evaluation results are very close to manual calibration. However, LoRA performs this calibration automatically in less time than manual calibration would require.

The main objective of this work was to evaluate the impact of applying machine learning techniques and large language models on sentiment analysis tasks. The results of the techniques using machine learning without feature selection and with default parameter values in the algorithms were comparable, with a slight improvement when parameter calibration was employed. However, a significant improvement in metrics was observed when feature selection was applied, with the highest point being reached at 72.72% accuracy when n-grams were used with the Bayes algorithm and parameter calibration. The combination of Bayes with unigrams+bigrams and feature selection improves the classification performance. The importance of feature selection was further highlighted by improved metrics for all algorithms when lemmatization and stemming were used. In experiments with large language models, a considerable improvement was observed in accuracy and F1 results when parameter calibration was used, compared to default hyperparameters. Although no specific hyperparameter significantly influenced the models' performance, an exploratory search allowed some combinations to yield better results. The model based on pre-training with tweets was particularly effective when adapted to the Cardiff text set, resulting in 72.72% accuracy and 71.43% F1, especially after repeating the experiment with more epochs. A metrics analysis was conducted comparing the best results obtained using machine learning algorithms with NLP techniques and large language models. The results showed an enhancement with the tweet-based LLM, as fine-tuning leveraged the model's adaptation to Twitter message characteristics during retraining. Furthermore, hyperparameter calibration enhanced classification performance. Conversely, combining n-grams with feature selection reduced term dictionary variability and selected the most relevant features. Despite being distinct approaches, both models adapted to the text set's specific characteristics, resulting in comparable outcomes. The summarized results are presented in Table 5.

Table 5. Comparison between the best results of machine learning vs LLM.

Approach	Accuracy score	F1 score
Bayes & Unigrams+Bigrams	72.72	72.81
LoRA & *twitter-xlm-roberta-base/SentUAM*	71.99	71.78

The F1 score achieved in this work and shown in Table 9 surpasses the work presented in (Camacho-Collados et al., 2022) using XLM-T model based on RoBERTa large language model, where they achieved 68.52 F1 score from the same Spanish texts.

5 Conclusions

This paper presents an approach to evaluate the performance of machine learning algorithms and calibrated large language models for sentiment analysis in Spanish texts. It uses natural language processing (NLP) techniques to process Spanish texts in order to improve the input for machine learning algorithms; it is important to note that large language models (LLMs) do not need to directly employ NLP.

The main contributions of this work include a) a sentiment analysis approach that considers NLP tasks and feature selection in combination with traditional machine learning algorithms; b) a comparison of machine learning algorithms with and without feature selection, and exploratory calibration to achieve optimal results for the task; c) fine-tuning of hyperparameters combined with training from the text set used in large language models for sentiment analysis task; d) measuring the performance of machine learning and large language models to select the best option. The results obtained reveal that feature selection significantly enhances the performance of models within the machine learning approach. Notable improvements in performance are observed with lemmatization and stemming, with the combination of unigrams and bigrams emerging as the best option when used in conjunction with this technique.On the other hand, in the context of Large Language Models (LLMs), it is observed that pretraining with tweets leverages this characteristic to achieve better results on similar text datasets. Additionally, the Low-Rank Adaptation (LoRA) technique demonstrates that low-rank adaptation can enhance model performance without the need to adjust traditional hyperparameters.

The outcomes of our research hold significant implications for social media analysts, who rely on sentiment analysis to discern sentiment in texts and adjust product-based strategies based on feedback. Similarly, researchers focusing on NLP or large language models can benefit from our work. We provide performance measurements for each configuration (algorithms, techniques, and fine-tuning parameters), which serve as invaluable tools for selecting the most suitable approach for sentiment

analysis in Spanish texts. This practical application of our findings underscores the relevance and applicability of our research. Our future work holds exciting prospects. We plan to delve deeper into the potential of NLP techniques, exploring specific cleaning, fine-tuning, and hyperparameter calibration of large language models to further enhance evaluation metrics. While our current approach was tested with one available Spanish text set, we are eager to expand our experiments to other text sets, such as texts generated by university students, opinions from faculty members, or evaluations from company staff. This ongoing research promises to uncover new insights and further advance the field of sentiment analysis.

References

Birattari, M., & Kacprzyk, J. (2009). *Tuning metaheuristics: a machine learning perspective* (Vol. 197). Berlin: Springer.

Zheng, A., & Casari, A. (2018). *Feature engineering for machine learning: principles and techniques for data scientists*. " O'Reilly Media, Inc.".

Kenton, J. D. M. W. C., & Toutanova, L. K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT* (Vol. 1, p. 2).

Chiruzzo, L., Etcheverry, M., & Rosá, A. (2020). Sentiment analysis in Spanish tweets: Some experiments with focus on neutral tweets.

Samuel, J., Ali, G. M. N., Rahman, M. M., Esawi, E., & Samuel, Y. (2020). Covid-19 public sentiment insights and machine learning for tweets classification. *Information*, *11*(6), 314.

Chintalapudi, N., Battineni, G., & Amenta, F. (2021). Sentimental analysis of COVID-19 tweets using deep learning models. *Infectious disease reports*, *13*(2), 329-339.

Indulkar, Y., & Patil, A. (2021, March). Comparative study of machine learning algorithms for twitter sentiment analysis. In *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)* (pp. 295-299). IEEE.

Bhargav, M., Alaria, S. K., & Mukhija, M. K. (2021). Implementation of Sentiment Analysis and Classification of Tweets Using Machine Learning. *Turkish Online Journal of Qualitative Inquiry*, *12*(10).

Yadav, N., Kudale, O., Rao, A., Gupta, S., & Shitole, A. (2021). Twitter sentiment analysis using supervised machine learning. In *Intelligent data communication technologies and internet of things: Proceedings of ICICI 2020* (pp. 631-642). Springer Singapore.

Kaur, H., Ahsaan, S. U., Alankar, B., & Chang, V. (2021). A proposed sentiment analysis deep learning algorithm for analyzing COVID-19 tweets. *Information Systems Frontiers*, *23*(6), 1417-1429.

Alam, K. N., Khan, M. S., Dhruba, A. R., Khan, M. M., Al-Amri, J. F., Masud, M., & Rawashdeh, M. (2021). [Retracted] Deep Learning-Based Sentiment Analysis of COVID-19 Vaccination Responses from Twitter Data. *Computational and Mathematical Methods in Medicine*, *2021*(1), 4321131.

Hidayat, M. R., & Sulistiyono, M. (2022, August). Comparison of Accuracy and Time Of Naïve Bayes Algorithm with Support Vector Machine Algorithm in Twitter Sentiment Analysis of Peduli Lindungi Application. In *2022 5th International Conference on Information and Communications Technology (ICOIACT)* (pp. 172-176). IEEE.

Parikh, A., Pawar, R., Shelke, P., Gadhave, R., & Bagade, J. (2022, November). Comparison of Machine Learning Algorithms for Twitter Sentiment Analysis. In *2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)* (pp. 209-215). IEEE.

AlBadani, B., Shi, R., & Dong, J. (2022). A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM. *Applied System Innovation*, *5*(1), 13.

Singh, C., Imam, T., Wibowo, S., & Grandhi, S. (2022). A deep learning approach for sentiment analysis of COVID-19 reviews. *Applied Sciences*, *12*(8), 3709.

Sun, M. (2022). *Natural Language Processing for Health System Messages: Deep Transfer Learning Approach to Aspect-Based Sentiment Analysis of COVID-19 Content* (Master's thesis, Harvard University).

Xu, L., & Song, Y. (2023, April). Comparison of text sentiment analysis based on traditional machine learning and deep learning methods. In *2023 4th International Conference on Computer Engineering and Application (ICCEA)* (pp. 692-695). IEEE.

Ashrita, Y., Abhiram, S., Hemanth, V., Srinivas, A., & Vemula, P. R. (2023, September). Deep Learning Techniques for Sentiment Analysis on Social Media Text. In *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)* (Vol. 6, pp. 2294-2300). IEEE.

Matías-Cristóbal, O., Padilla-Caballero, J., Gonzales-Rivera, R., Benavente-Ayquipa, R., Pérez-Saavedra, S., & Cardenas-Palomino, F. (2023, September). Enhancing Sentiment Analysis In Text Of Social Media Texts Using Hybrid Deep Learning Model And Natural Language Processing. In *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)* (Vol. 6, pp. 1776-1780). IEEE.

Díaz Galiano, M. C., Martínez Cámara, E., García Cumbreras, M. Á., García Vega, M., & Villena Román, J. (2018). The democratization of deep learning in TASS 2017.

Barbieri, F., Anke, L. E., & Camacho-Collados, J. (2021). XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond. *arXiv preprint arXiv:2104.12250*.

Twitter/twhin-bert-base · Hugging Face. (s.f.-b). Hugging Face – The AI community building the future. https://huggingface.co/Twitter/twhin-bert-base

cardiffnlp/twitter-xlm-roberta-base · Hugging Face. (s.f.). Hugging Face – The AI community building the future. https://huggingface.co/cardiffnlp/twitter-xlm-roberta-base

google-bert/bert-base-multilingual-uncased · Hugging Face. (s.f.). Hugging Face – The AI community building the future. https://huggingface.co/google-bert/bert-base-multilingual-uncased

cardiffnlp/tweet_sentiment_multilingual · Datasets at Hugging Face. (s.f.). Hugging Face – The AI community building the future. https://huggingface.co/datasets/cardiffnlp/tweet_sentiment_multilingual

Fodor, I. K. (2002). *A survey of dimension reduction techniques* (No. UCRL-ID-148494). Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).

Birattari, M., & Kacprzyk, J. (2009). *Tuning metaheuristics: a machine learning perspective* (Vol. 197). Berlin: Springer.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.