applied sciences

MDPI

Article

Social Network Sentiment Analysis Using Hybrid Deep Learning Models

Noemí Merayo [1], Jesús Vegas [2,*], César Llamas [2] and Patricia Fernández [1]

[1] Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad de Valladolid, 47005 Valladolid, Spain; noemer@uva.es (N.M.); patfer@uva.es (P.F.)
[2] Escuela de Ingeniería Informática, Universidad de Valladolid, 47005 Valladolid, Spain; cesar.llamas@uva.es
* Correspondence: jvegas@uva.es

Featured Application: The system presented in this study was aimed at business, organisation, government, and consumer areas, whereas a real-time application is needed for the classification of social network messages written in Spanish through sentiment semantics metadata. Our model improved the performance of other existing machine learning techniques by up to 20 percentage points. These high levels of accuracy are crucial for obtaining real-time ratings from thousands of people for the effective monitoring of social media discourse in decision-making and strategy implementations.

Abstract: The exponential growth in information on the Internet, particularly within social networks, highlights the importance of sentiment and opinion analysis. The intrinsic characteristics of the Spanish language coupled with the short length and lack of context of messages on social media pose a challenge for sentiment analysis in social networks. In this study, we present a hybrid deep learning model combining convolutional and long short-term memory layers to detect polarity levels in Twitter for the Spanish language. Our model significantly improved the accuracy of existing approaches by up to 20%, achieving accuracies of around 76% for three polarities (positive, negative, neutral) and 91% for two polarities (positive, negative).

Keywords: deep learning; hybrid strategies; sentiment analysis; social networks; Twitter; Spanish

check for updates

Citation: Merayo, N.; Vegas, J.; Llamas, C.; Fernández, P. Social Network Sentiment Analysis Using Hybrid Deep Learning Models. *Appl. Sci.* **2023**, *13*, 11608. https://doi.org/10.3390/app132011608

Academic Editor: Duy-Tai Dinh and Uday Kiran RAGE

Received: 25 September 2023
Revised: 11 October 2023
Accepted: 13 October 2023
Published: 23 October 2023

1. Introduction

Social networks have become the most representative tools of Web 2.0 (the Internet), allowing millions of users to post and share information in a fast and seamless way, permitting a continuous flow of information. According to figures from different studies [1], the latest data from 2022 state that more than 9 out of 10 Internet users already use social media every month, and the percentage of social media users amounts to 75% of the world's population. In this scenario, 17 social media platforms accounted for at least 300 million active users in January 2022. Facebook is the most used, with over 2.91 billion monthly active users, followed by YouTube with 2.562 billion users, and WhatsApp with 2 billion users. Twitter, which combines the features of blogging, social networking, and instant messaging, has about 436 million active users worldwide, standing out not only for its popularity but also for its great monetisation and business potential. As a matter of fact, Twitter seems to be the seventh-favourite social network among Internet users in the 16–64 years age group [1]. Although this social network has recently changed its name, we will keep the original nomenclature in order to facilitate the reading and understanding of this article.

In any case, due to the impressively growing amount of information on the Internet and social networks, as well as the large number of sources and the high number of opinions about any given content, it is essential to have automatic methods that allow us to classify and analyse information quickly and efficiently; all this is used to feed any decision-making system. Natural language processing (NLP) seems to be a more rational choice to tackle complexity in the area of artificial intelligence (AI) [2]. The well-known specific fields required in this area are opinion mining (OM) and sentiment analysis (SA). These fields combine NLP and computational linguistics, which involve uncovering words and contexts to understand the opinions they reveal. More precisely, SA deals with determining the emotional tone behind a series of words in order to classify them via their polarity (positive, negative, or neutral) or emotion (sadness, joy, disappointment, etc.). Additionally, contents on social networks such as Twitter have different natures depending on the language employed, which has its own peculiarities, making it different from other uses of language [3]. In fact, some important problematic features of tweets are related to their short length (280 characters), the data sparsity, the lack of context, the low concern for grammar, and the use of an informal linguistic style (idioms, slang, and abbreviations). These characteristics make it difficult to integrate fully effective SA systems into social networks such as Twitter.

Moreover, the application of SA in social networks is very useful in many cases, such as measuring the impact of social media actions, helping understand what consumers think about brands/companies, knowing what users think about certain topics, making better decisions on marketing strategies for product/service development, following trends in real-time, or even assisting in predicting the behaviour of users. For these reasons, the application of NLP strategies and SA to social networks is an active area of applied and basic research in order to make social networks and the web more usable and profitable.

However, the rise in the use of Spanish continues to increase each day. Currently, it is the second most widely spoken native language, with nearly 493 million native speakers. Therefore, the Spanish-speaking population occupies an important place in the dissemination of information worldwide; about 7.9% of Internet users usually communicate in Spanish, so Spanish is the third most used language on the Internet, followed by English and Chinese [4]. The use of Spanish on the Internet is also growing; for example, in the period from 2000 to 2020, it experienced a growth of 1511% compared to the 743% increase recorded for English. This increase could be explained via the preference of the USA-based Hispanic community to consume and create digital content in Spanish, rather than in English. In this way, regarding the use of Spanish on websites with multilingual content, Spanish is used on 4.1% of these websites, which places this language in fourth position, ahead of German and French [4]. The use of Spanish on social networks is also very high, and right now, it is the second most used language on digital platforms and social networks, such as YouTube, Facebook, Netflix, LinkedIn, Wikipedia, Instagram, and so on, after English. In particular, if we analyse the percentage of active social network users in relation to the total population in 2022, Spain was eleventh place with respect to the number of social network users, with 87.1% of its population using social network users, followed by other Spanish-speaking countries, such as Argentina, with 83.3% (thirteenth place), and Colombia, with 81.3% (twenty-third place) [1].

Regarding this, there are many studies that have focused on SA, but most of them are related to documents written in English. Although there are studies in Spanish on the SA of social networks [5,6], more research is needed due to the high complexity of detecting, handling, and processing certain aspects of Spanish, such as negation, sarcasm, or ambiguity, which is even more complex in the context of social networks. These are, therefore, major challenges that are still open and need to be addressed. In SA, two main approaches have been combined for the task: (i) lexical-resource-based approaches and (ii) supervised-learning-based approaches. The first approach requires a dictionary of words associated with a sentiment (positive, negative, or neutral). Such dictionaries are compilations that capture prior knowledge of the words that appear in them. In supervised-learning-based approaches, no lexical resources are required, but a set of previously labelled examples of opinions is required. Finally, there are hybrid approaches that combine the two previous techniques. As for the first approach, the number of lexicons in Spanish is limited, and they need to be developed for each specific domain. Moreover, this approach must be combined with additional techniques to detect negation, sarcasm, or ambiguity in language, making them more complex strategies. However, the supervised learning approach needs a labelled corpus, but multiple machine learning algorithms can be applied to the same corpus with greater versatility. In terms of the effectiveness of these two techniques, some works show that supervised approaches could improve the accuracy of lexicon-based approaches in some cases and contexts [7].

However, in addition, of all the different machine learning models that can be applied, deep learning algorithms have been gaining prominence in recent years, as they can achieve better results than traditional models and are able to run faster thanks to the high performance of graphics processing units (GPUs).

In addition, in deep learning approaches, the feature vector is composed of word embeddings (WEs) [8], which are much richer in word-related information than the features used in traditional algorithms, as they capture more information about similarity and semantic features of words. However, despite these important benefits, there are few proposals in Spanish for SA in social networks using deep learning, specifically in Twitter, and those that do exist have room for improvement in their accuracy performance. Therefore, we propose the design of a novel hybrid model based on deep learning strategies applied to a generic corpus of labelled Twitter datasets, which improves the accuracy of other existing models. The main objective of this study is to develop a deep learning model for sentiment analysis on Twitter using a labelled corpus from the Spanish TASS dataset [6,9], which is part of the Spanish Workshop on Sentiment Analysis at Spanish Society for Natural Language Processing (SEPLN) [10]. The aim is to improve the efficiency of detecting the polarity (positive, negative, or neutral) of opinions expressed on Twitter, with high levels of accuracy. This is crucial for obtaining real-time assessments from thousands of individuals, which is valuable for companies, organisations, governments, and consumers. By avoiding irrelevant data, this approach enables effective decision-making and strategies in real-time. Furthermore, polarity detection facilitates the management and monitoring of online reputation and social media discourses. This research also contributes to the study and analysis of various hybrid deep neural network models that combine different types of layers, with the ultimate goal of optimising natural language processing in social networks.

This paper is organised as follows: Section 2 describes the state of the art regarding sentiment analysis in Spanish; Section 3 describes the proposed model and its parameters; Section 4 shows the dataset and the results of the designed model in Spanish tweets; Section 5 deals with the discussion of the results obtained; and, finally, Section 6 summarises the main findings and conclusions of the conducted research.

2. Related Works

As mentioned in the previous section, we will consider two categories of approaches to sentiment analysis in texts: the first one is based on lexical resources and the second one is based on supervised learning.

2.1. Sentiment Analysis Based on Lexical Resources

As for techniques based on Spanish lexical resources, there are several opinion lexicons that encompass words associated with a sentiment or an opinion value (polarity). However, the amount of Spanish lexicons is quite limited. As far as polarity lexicons are concerned, we can consider the opinion lexicon developed by [9] or the iSOL [11], based on one of the most important English lexicons for the classification of polarity [12]. In addition, the Spanish adaptation of the Affective Norms for English Words (ANEW) is presented in [13]. In another work, the authors of [14] developed a lemma-level sentiment lexicon for several languages, including Spanish, generated from an improved version of SentiWordNet, a popular lexicon with positive and negative words. The authors of [15] have developed Sentitext, a sentiment analysis software for Spanish, based on knowledge and supported by several word databases. In contrast, other lexicons are based on emotions rather than polarity, such as the one developed by [16], which considers six emotions.

However, using only opinion lexicons is not entirely effective for all domains, as each domain usually has a specific way of expressing a positive or negative opinion. Moreover, these lexical resources alone are not able to cope with the detection of negation, sarcasm, or ambiguity in opinions. Consequently, many authors focus on the processing of negation in Spanish. Thus, the authors in [17] adapted the SO-CAL sentiment analysis tool from English to Spanish [18]. This proposal consists of a lexical dictionary with positive and negative words together with the integration of rules and intensifiers to detect the degree of negation. Furthermore, the authors in [19] incorporated dependency-based techniques for negation detection to establish the scope of intensifiers and negation cues. This system achieves an accuracy of around 78% in detecting only two polarities (negative or positive), but it is applied in the context of product and service reviews, not on Twitter, where the language characteristics are different. In this sense, the authors of [20] have proposed rules based on dependency trees to identify the scope of the most important negation cues, defined by the Royal Spanish Academy, together with a lexicon-based sentiment analysis system for polarity classification. This system was applied to study the scope of negation in Twitter [21], reaching accuracies of around 62% when detecting three levels of polarity (positive, negative, or neutral). In the same line of work, Miranda et al. [22] developed an opinion mining system based on the ANEW Spanish lexicon applied to hotel reviews using the negation cues studied in [20]. Although the accuracy increases above 90%, it only focuses on detecting two polarities (negative and positive) and does not integrate the more difficult-to-identify neutral polarity. Furthermore, the system was developed in the context of hotel reviews, not in the more complex context of social networks such as Twitter. Amores et al. [23] combined different methods to deal with negation by applying effects of negation, modifiers, jargon, abbreviations, and emoticons in sentiment analysis, reaching an accuracy on a Twitter corpus between 83 and 87% for the detection of two polarity classes: positive and negative.

Other approaches take into account not only sentiment but also subject matter. Thus, Anta et al. [24] have proposed a classification system for Spanish tweets by evaluating the use of stemmers and lemmatisers, n-grams, word types, negations, valence shifters, link processing, search engines, special Twitter semantics (hashtags), and different classification methods, where the highest accuracies are around 58% for topics and 42% for Twitter polarity detection. Furthermore, the authors in [25] implemented a naïve-Bayes classifier to detect the polarity of Spanish tweets, identifying different levels of polarity along with unigrams of lemmas and multiwords based on PoS tag patterns to detect the scope of negation. The accuracy of the system was 66% for four polarity levels and 55% for six polarity levels. Finally, the authors of [26] describe a Transformer-based approach to detect negation in a corpus of Spanish product reviews, achieving accuracies between 80% and 90%, although this system has not been applied in social network contexts or on Twitter. However, other negation strategies are based on annotated corpora with negation in contrast to negation cues [20]. Thus, we can find several corpora related to clinical records, such as the IxaMed-GS corpus, the UHU-HUVR, and the IULA Spanish Clinical Record. Furthermore, the UAM Spanish Treebank corpus was extracted from syntactically annotated newspaper articles.

Finally, the most recent Spanish negation corpus, the SFU ReviewSP-NEG18, also considers discontinuous negation markers and is related to products and services reviews. Therefore, it can be concluded that lexical approaches require the use of sentiment word dictionaries combined with strong negation detection strategies. However, in addition, much of these works have not been applied in social network contexts such as Twitter, where language features are more complex (sparsity, lack of context, little concern for grammar usage, informal linguistic style, etc.). Moreover, these approaches also need to integrate strategies to detect other language features, such as sarcasm or irony, which are very typical in social networks.

2.2. Sentiment Analysis based on Supervised Machine Learning

However, techniques based on supervised machine learning do not require lexical resources but a set of previously tagged examples of opinions. However, obtaining labelled examples is a costly task, as these examples are usually labelled manually; in addition, it is important to have examples for the classification domain in which it is applied. Thus, in Spanish, we can find in the literature some research studies that address this problem.

In [27], Myska et al. focus on sentiment analysis in text documents by applying a support vector machine (SVM) classifier. For training and testing, a dataset with positive and negative valence texts was used, based on the analysis from web pages with product ratings. Based on the rating of the text, it was resolved whether the text is positive or negative. The texts were divided into three groups: positive texts (P), negative texts (N), and texts with no defined valence NEU (neutral, not used). The recognition system was validated in four different languages, including Spanish. Regarding the results in the Spanish language, the classification accuracy was 93.23% when predicting P and N texts, not without considering the neutral class. Although this work shows interesting results, it was carried out with posts on web pages that have a different nature than Twitter posts as, for example, the size of the latter are limited to 280 characters. Thus, the use of language on Twitter has its own characteristics that differentiate it from other uses of language [3]. In addition, the samples database was categorised as P, N, and NEU based on starts and some thresholds. In contrast, our database is manually labelled via human effort. All these differences make it impossible to generalise their results to the SA of tweets in Spanish and justify the need to explore other alternatives for this purpose by applying Deep Learning.

In a similar context, the authors of [27] proposed a linguistically independent text classifier based on convolutional recurrent neural networks. The classifier works at the character level instead of higher structures (words and sentences). The models were tested on the Yelp dataset (reviews and user data) and on a privately collected multilingual dataset, and the classification accuracy for Spanish texts was only 67.33%. Focusing on sentiment analysis of Spanish tweets, some research can be found in the literature. In [28], SA was carried out in the context of the Colombian presidential election of 2014. In this case, the corpus consisted of 1030 tweets tagged via humans with positive, negative, or neutral polarity towards each of the candidates. This process included a feature extraction preceded by a normalisation phase, after which a logistic regression classifier was used to assign a label class to each tweet. The results showed some difficulties in inferring the vote based on sentiment analysis of the tweets. In the conclusions, the authors argued that the obtained results showed that inference methods based on Twitter data are not consistent and that more work is needed to deal with the characteristics of the language. In this sense, our proposal tries to avoid the effects of feature extraction with the use of deep learning techniques. Deep learning makes problem solving much easier because it completely automates what used to be the most crucial step in a machine learning workflow, which is feature engineering [29], as they rely on the use of unsupervised pre-training features, the most commonly found of which are word embedding vectors [30]. More recently, the authors of [30] built the first Spanish corpus of sexist expressions on Twitter and applied different techniques, such as SVM, random forest (RF), long short-term memory (LSTM) networks and Transformer Bidirectional Encoder Representations from Transformers (BERTs), to construct a novel Transformer architecture that aims to provide very promising results in NLP. The proposed methods achieved an accuracy between 61 and 74% in the automatic detection of sexist behaviours. However, this corpus is not oriented towards polarity detection as it focuses on the detection of sexist behaviour on Twitter. In a similar context, research can be found in [31], where the authors developed a proposal to detect hate speech in Spanish tweets, using BERT. The results allow distinguishing between non-aggressive and aggressive tweets (two classes) with an accuracy between 79 and 86% on different datasets. However, the authors of [32] have proposed an approach for annotators to reach a consensus in the process of annotating comments on Spanish social networks. They built a corpus with 3259 Spanish comments (P, N, and NEU) and applied several classifiers for sentiment analysis detection, achieving 70% as the average F1-Score with multilayer perceptron. In the context of the Workshop on Sentiment Analysis at SEPLN (TASS), studies focusing on SA in Spanish tweets can be found [6,9] using the TASS corpus. For example, in [33], a set of classifiers based on SVM, convolutional neural networks (CNNs), and LSTM were used over three variants of Spanish. The achieved results show the best accuracies around 59%, 59.2%, and 54.9% for the SVM, CNN, and LSTM algorithms, respectively. Similar results were reported in [34,35]. Another approach using SVM was implemented in [36] for SA on Spanish tweets, achieving an accuracy of 62.88% when predicting five levels of polarity (P+, P, N, N, N+, and NEU) and 70.25% for three levels (P, N, and NEU). Furthermore, the authors of [6] have proposed two deep neural network models (CNNs and dense neural networks) integrating a Gaussian noise layer for tweet polarity classification in a Spanish Twitter corpus, achieving an accuracy of only 57%. Unlike previous works that have focused on testing the effectiveness of different neural network models separately, our work aims to address SA by using a composition of different types of deep neural network layers in what we call a hybrid deep learning model.

The current state of sentiment analysis systems on Twitter continues to advance with the integration of hybrid deep learning architectures, a trend supported by recent research. In [37], Shazly et al. propose a hybrid architecture that combines the power of bidirectional recurrent neural networks (Bi-RNNs) and other techniques to enhance the efficiency and accuracy of sentiment analysis on Twitter data in an Arabic benchmark dataset. Bi-RNNs are employed to capture contextual information and sequential dependencies within tweets, which is crucial for understanding the nuances of sentiment in short, text-based social media content. In a 2022 study by Li and Shujuanl [38], a hybrid model combining a bidirectional long short-term memory (BiLSTM) neural network and a convolutional neural network (CNN) showed remarkable improvements in accurately analysing text emotion compared with a single CNN model. Furthermore, the impact of Transformer-based models in sentiment analysis, as highlighted in a study by Mewada et al., in 2023 [39], remains significant. BERT (Bidirectional Encoder Representations from Transformers) and its variants continue to be influential in leveraging contextual embeddings. The incorporation of ensemble strategies, such as those discussed by Shah et al. [40], play an essential role in these contemporary hybrid frameworks by combining predictions from multiple models to improve sentiment classification accuracy. These recent articles collectively highlight the dynamic and innovative landscape of hybrid deep learning approaches in tweets sentiment analysis.

2.3. *Ethical Considerations*

In the field of sentiment analysis, ethical considerations play a key role in ensuring the responsible and fair application of this technology. One of the main concerns revolves around the potential biases present in the datasets used for training sentiment analysis models. These biases can stem from the data collection process, where certain demographics or viewpoints may be over-represented or under-represented, leading to biased results. Addressing these biases is crucial to prevent reinforcing existing stereotypes or perpetuating discrimination.

In addition, the application of sentiment analysis in decision-making processes raises important ethical questions. Depending on the context, relying solely on sentiment analysis can have profound implications. Decisions based on sentiment analysis might inadvertently prioritise popular opinions over minority voices or fail to take into account the nuances and complexities of human emotions. Striking a balance between the perspectives offered by sentiment analysis and ethical considerations of fairness, transparency, and inclusivity is essential for ensuring that this technology is used responsibly and effectively. Engaging in these discussions is essential for a more comprehensive analysis of the ethical landscape surrounding sentiment analysis. However, a more in-depth discussion on this topic is beyond the scope of this study.

3. The Proposed Model

Our deep learning model combines the capabilities provided via convolutional and recurrent layers. While the convolutional neural networks (convnets) have proven their ability to tackle perception-related tasks, such as computer vision, they have also demonstrated their ability to deal with other problems, such as natural language processing, by replacing other classic machine learning techniques [29]. An important feature of convnets is that they have no memory. Each input shown to them is processed independently, without maintaining any state between inputs. In contrast, to read a tweet, one has to process all its words one by one while memorising what has already been read; this gives a smooth representation of the meaning conveyed by this tweet. Moreover, a recurrent neural network (RNN) adopts the same operation principle: it processes sequences by iterating through the elements of the sequence and maintaining a state containing information about what it has seen so far. Thus, long short-term memory (LSTM) layers and gated recurrent unit (GRU) layers are designed to cope with this problem [29]. Therefore, the proposed hybrid model, as depicted in Figure 1, is composed of the following five layers: (1) embeddings, (2) a one-dimensional convolutional layer (Conv1D), (3) a MaxPooling layer, (4) an RNN layer (LSTM or GRU), and (5) a dense layer on top. This hybrid model will be used to classify tweets into the three (P, N, or NONE) or two (P or N) categories.

In the following, some descriptive considerations will be made for each of the layers, although the details of the hyperparameters will be shown in the following sections as they are the subject of the experiments. The first layer—the embedding layer—is responsible for transforming tweets into numbers so that the network can understand them and process them properly in the form of tensors. The developed model uses embeddings calculated during the training of the network with Twitter's own dataset. For this purpose, random word vectors are initialised and adjusted using the backpropagation algorithm, which is similar to the process of adjusting the weights of a neural network. The output embeddings have a dimension size and an input tensor size, i.e., in our model, the input layer consists of a matrix of x rows and y columns, the values of which will be optimised in different experiments to be described in later sections. The rows represent each of the tokenised words that make up the tweets, and the columns represent the weights assigned to each of the words. The second layer consists of a one-dimensional CONVnet layer and is widely used in our field for text classification. This element will be parametrised via its kernel size and will use the *relu* activation function as this is usually the best activation function in these types of layers. The kernel size specifies the size of the patterns that the convolutional network can recognise. The purpose of the convolutional layer is to simplify the work of the next RNN layer, i.e., it reduces the processing of the RNN layer by suppressing certain intermediate steps by detecting text patterns. The convolutional layer will have a given number of neurons, and this number will be optimised during an experimental process that will be shown in the following sections. The next layer, called the MaxPooling1D layer, serves to reduce the dimensionality needed for the next RNN stage. The width and height dimensions tend to shrink the number of feature-map coefficients to be processed for the next layer. The next stage involves a recurrent neural network (RNN), and as mentioned before, there are two possibilities in this stage: an LSTM layer or a GRU layer. For this reason, two versions of the hybrid model will be considered in our experiments to compare the performance of both hybrid architectures. The first hybrid model will be built on top of an LSTM layer and the second one on a GRU layer. In the LSTM case, the layer is parametrised via the number of neurons, a dropout rate, and a recurrent dropout rate, the optimal values of which will be analysed and set in Sections 4 and 5. The last two parameters aim to reduce overfitting from the deactivation of neurons due to normal connections (dropout) and the deactivation of neurons due to recurrent connections (recurrent dropout). In the case of a GRU layer, there is a parametrisation similar to that detailed for the LSTM layer. The last layer consists of a dense layer with as many neurons as there are classes we want to recognise in our problem. In this case, we are initially faced with a three-category (P, N, and NONE) classification problem, although we will also analyse a two-category (P and N) scenario. For this reason, we will take into account a layer with two or three elements depending on the classification problem posed. For this layer, the Softmax activation function will be used to transform the outputs into a representation in the form of probabilities so that the sum of all the outputs equals 1. This type of stage is usually used as an output layer in multiclass classification problems, such as the one presented in this study.

Regarding network convergence , binary and categorical cross-entropy loss functions were selected due to the binary data available for this experiment, and we compared the performance of both functions to select the best one. These loss functions are the most frequently used in classification problems. The former (binary cross-entropy) is used in problems where the input can be classified into two labels, and the latter (categorical cross-entropy) is used when the input is classified into three or more labels. Furthermore, in our model, the optimisers RMSprop and ADAM were chosen to test their performance and select the best one.

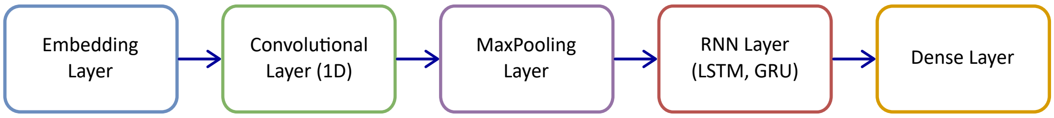

Figure 1. Block diagram of our proposed hybrid model showing the different layers.

4. Experiments and Results

The following sections show the research methodology applied in this work. It begins with an initial comparison of different combinations of recurrent layers (LSTM and GRU), choosing the best of all. From this optimal configuration, the rest of the hyperparameters are analysed and configured to optimise different accuracy metrics.

A fundamental issue when evaluating models is to have two different sets: the training set (to train our model) and the validation set (to evaluate our model). In our case, this was achieved by splitting the data into a typical ratio of 70–30% [41], and the samples of the training dataset were randomly selected from the three classes. However, to solve the problem of whether the datasets that have been selected are suitable for evaluating the model, the cross-validation technique is used, which attempts to find the best sets for the model we train. In our case, we use a k-fold cross-validation technique (k = 10), which consists of performing k iterations, so that the model is being trained and evaluated k times, or 10 times in our case. Finally, we apply the EarlyStopping technique to regulate overfitting and prevent models from losing generalisation [41]. This technique consists of stopping the training of the model at a point where the validation loss is minimised, that is, when the training loss metric stops improving, the training stops automatically.

Our deep-learning-based hybrid model was implemented in Python using the Keras [42] and TensorFlow [43] libraries. All models were run on the Google Co-laboratory (Colab) platform, which allows running Jupyter notebooks on shared hardware provided by Google, giving access to computing infrastructure similar to NVIDIA TESLA K80 and NVIDIA TESLA K4 GPUs with 13GB of RAM. A Jupyter notebook is an environment that allows you to mix executable Python code, with text and visualisations of the results, all within the same document.

4.1. Dataset

The corpus (ready for free download) was obtained from the Sentiment Analysis Workshop (TASS corpus) of the Spanish Society of Natural Language Processing (SEPLN), where the tweets are labelled with their polarity [10,44]. Our corpus contains 60,000 tweets written in Spanish by 150 personalities and celebrities from the world of politics, economics, communication, and culture, obtained between November 2011 and March 2012.

Each tweet includes its ID (tweetid), creation date (date), and user ID (user), and all user information was removed. Tweets are classified into four categories: positive (P), neutral (NEU), negative (N), and no sentiment (NONE), where tweets classified as

NONE do not express any idea. Then, the distribution of the corpus in classes is as follows: 22,021 positive tweets (P), 15,748 negative tweets (N), 21,094 tweets without polarity (NONE), and 1090 neutral tweets (NEU). The annotation of comments in these classes was performed semi-automatically via the corpus creators: a basic machine learning model was first run and then all tags were verified by human experts. In the case of entity-level polarity, due to the large volume of data to be verified, human annotation was only performed for the training set. However, this particular case is a multiclass classification problem that shows the following possible sentiments: positive (P), negative (N), neutral (NEU) and none (NONE). Furthermore, since each tweet can only belong to one sentiment, it is known as a single-label multiclass classification problem. One issue to take into account in labelled corpora is the balance between classes, since if we are solving a classification problem and have more data from one class than another, a model will be more likely to predict a tweet from the class with the highest number of tweets. Therefore, if, in our corpus, we want to test the model with four classes, we would have to somehow balance the number of tweets from the NEU class, the most minority class. A typical solution would be to generate artificial samples to balance the data since there are no real tweets from this class. This method was discarded because it is not known how these samples are generated, and in this particular case, the number of artificial samples should be very high, which can cause an over-fitting problem. Consequently, we initially tested our model only with the three most representative classes: positive (P), negative (N), and none (NONE). Thereafter, the model was analysed with two classes (P and N).

4.2. Data Preprocessing and Encoding

Once the corpus has been selected and the problem classified, the tweets should be cleaned and processed so that the network can understand the texts under the same conditions. The following considerations were taken into account in the preprocessing stage to normalise the data and avoid some grammatical errors:

- To convert to lowercase to avoid the duplication of words due to the inability to distinguish between upper case and lower case letters.
- To remove URLs as they do not provide information on opinion.
- To remove mentions (@) as they refer to other users and do not provide useful information.
- To remove hashtags (#) and retweets (RTs).
- To remove accents and diacritical vowels to eliminate spelling errors and standardise all words.
- To remove punctuation marks to reduce spelling errors in the text.
- To reduce the repetition of characters, e.g., change “Holaaaaaa” to “hola” (“hello” in English).
- To normalise laughter.
- To standardise slang/jargon to reduce spelling mistakes, e.g., change “tb” to “también” (“too” in English), “tq/tk” to “te quiero” (“I love you” in English), “+” to “más” (“more” in English), and “x” to “por” (“for” in English).

Nonetheless, it is important to highlight that in the data preprocessing phase of this proposal, no automated text correction tool was used to correct grammatical errors.

The next step is the tokenisation process, which consists of considering each word of each tweet as a token, so a particular tokeniser created for Twitter called TweetTokenizer [45] will be used. The next step is token extraction, which consists of transforming text into numbers, as a neural network cannot be fed by text strings. In fact, the inputs to the neural network must be only numbers, and in our case, we must transform both the tweet (tokenising and normalising the text) and the tag (which corresponds to a letter and represents the polarity of each tweet). To transform the tweet data into numbers, a dictionary was created in which each tweet will be represented by a vector of corresponding indices in the dictionary. To transform the tags, the One Hot Encoding technique, a mechanism that encodes the different classes as a matrix, was used. In the case that we distinguish between three polarities (P, N, and NONE), the matrix will have three columns, in which a "1" is placed in the column corresponding to the class to which a tweet belongs and a "0" in the other two columns.

Finally, feature reduction is an optional step and consists of reducing the vocabulary that the neural network can handle. In our case, initially, we have considered using the following two techniques:

- To delete Stopwords: There is a set of words that, although necessary to construct meaningful sentences, lack information to determine polarity in texts and/or sentences. In Spanish, these words are prepositions, pronouns, conjunctions, and different forms of certain verbs such as "haber" ("to have") or "ser" ("to be"), among others.
- To apply stemming: This is a process of morphological normalisation whereby a word is transformed into its root by removing its suffixes and inflexions. For example, the word "guapas" ("beautiful") would be converted to its root "guap". In this case, we use the SnowballStemmer [46] in Spanish as it is the most used for this type of problem.

4.3. Configuration of the Hybrid Model with Different Combinations of Recurrent Layers

In the proposed hybrid model, different combinations were used to analyse which of them improves the results. Specifically, the hybrid model was designed using first the convolutional layer plus an LSTM layer and then the convolutional layer plus a GRU layer to compare their performance. To ensure that all models are under the same conditions, some initial fixed parameters were decided on. The initial values chosen are within the range of typical values in these types of studies, although simulations have been carried out to corroborate their good performance, which is not included in this article due to space savings. Furthermore, it should be noted that these and additional parameters will be analysed and optimised in the following sections once the optimal combination of hybrid model layers has been identified:

- The dimension of the embedding layer is 200.
- The length of the input tensors is chosen between 33 and 200.
- The fixed vocabulary size is 45,402 words (whole corpus).

Table 1 shows the accuracy results of both models with different hyperparameters related to the optimiser and the loss functions used. In the case of optimisers, Rmsprop or Adam are the most recommended for this type of problem [6,29]. In fact, the Adam optimiser is much more versatile, as it is a mixture of Rmsprop with several factors that generally make the model perform better, although the performance of Rmsprop will also be analysed. In terms of entropy loss functions, binary and categorical functions are the most commonly used in classification problems. Thus, binary cross-entropy is used in problems where the input can be classified into two labels, and categorical cross-entropy is used when the input can be classified into three or more labels [6,29]. As can be seen in the table, the combination of the Adam optimiser with the categorical cross-entropy loss function provides the best results for both hybrid models tested. However, the hybrid convolutional model with an LSTM layer is preferred as it gives slightly better accuracy results. To be sure of this choice, a further comparison was made with the hyperparameter that controls the maximum length of the input tensors, initially set to 33. In this way, the neural network models are fed via vectors of numbers known as tensors, and the size of these tensors are important, depending on the size of the texts to be analysed. Therefore, we have tried to modify this parameter by increasing its value to 200 since it is important that the models work correctly regardless of this value, that is, for any text length. The results showed that considering larger input lengths, the GRU model provides worse accuracy results (between 55% and 70%) considering the same combinations in Table 1, while the hybrid convolutional model with LSTM gives better results (between 70% and 73%) for all combinations, so the accuracy remains stable regardless of the dimension of the tensor.

Due to its lack of stability in the GRU layer results, the experiments continue considering only the LSTM as the RRN layer.

Table 1. Summary of tests performed to evaluate the hybrid model with LSTM and GRU layer.

Large Hybrid Model				
Hybrid Model			Cross-Validation Accuracy	
Combination	Optimiser	Loss Function	n = 33	n = 200
Convolutional + LSTM	Adam	Binary *	73.14%	73.18%
Convolutional + LSTM	Adam	Categorical †	73.24%	73.20%
Convolutional + LSTM	Rmsprop	Binary	72.87%	72.31%
Convolutional + LSTM	Rmsprop	Categorical	72.90%	71.68%
Convolutional + GRU	Adam	Binary	72.11%	60.03%
Convolutional + GRU	Adam	Categorical	73.12%	55.30%
Convolutional + GRU	Rmsprop	Binary	72.10%	69.79%
Convolutional + GRU	Rmsprop	Categorical	72.08%	64.63%

* Binary cross-entropy. † Categorical cross-entropy.

4.4. Optimum Hyperparameter Setting

For the choice of the hyperparameters, the neurons and kernel size of the convolutional layer and the neurons, dropout rate, and recurrent dropout rate of the LSTM layer were adjusted. According to the Keras documentation [42], LSTM layers have two different types of dropout rates, which are represented as a floating point number between 0 and 1. To evaluate the results, different tests were carried out, with the nine best combinations shown in Table 2 as an example of the tests performed.

Table 2. Summary of tests performed to assess hyperparameters, layer size (number of neurons) for the convolutional and LSTM cases and kernel size.

Layer Size		Dropout Parameter		Kernel	Cross-Validation
Convolutional	LSTM	Convolutional	LSTM	Size	Accuracy
192	96	0.2	0.3	8	74.13%
160	128	0.2	0.2	8	74.10%
192	128	0.2	0.4	3	74.03%
128	96	0.4	0.4	5	73.99%
160	64	0.3	0.2	8	73.90%
192	64	0.3	0.3	3	73.81%
128	32	0.4	0.2	5	73.77%
96	64	0.3	0.2	3	73.66%
64	32	0.3	0.3	8	73.61%

As shown in Table 2, the best values of the hyperparameters after the different tests to achieve the best accuracy (74.13%) are as follows:

- Neurons in the Convolutional layer: 192;
- Neurons in the LSTM layer: 96;
- Dropout rate in the convolutional layer: 0.2;
- Dropout rate in the LSTM layer: 0.3;
- Kernel size: 8.

However, there are two ways to load the embeddings in the process of creating deep neural network models for the word processing task. The first option is that the embeddings are learned as the network is trained with information from the corpus itself. The other option, which is the one employed in our model, is to use pre-trained word embeddings, that is, instead of the weights being adjusted with the corpus itself, they are matched to a defined dictionary that has been created to solve this kind of problem. Thus, the cross-validation accuracy obtained using two well-known pre-trained embedded word dictionaries in Spanish are as follows:

- GloVe dictionary: 66.55%.
- Word2vec dictionary: 67.79%.

As can be seen, the results become worse, and this is due to the fact that a generic dictionary, being created in a general way and applicable to other corpora, depends, to a large extent, on the relationship between the words in our corpus and those in that dictionary, so it will always be better to learn from something more specific and closer to our corpus directly.

However, it is also possible to create a vocabulary from a clean corpus by counting the occurrences of each word in the corpus. Therefore, for the selection of vocabulary features, the Information Gain (IG) method is applied to reduce and optimise the vocabulary and keep the terms with the highest frequency. This method is chosen instead of absolute frequency because the most frequent terms are not always the ones that best discriminate between classes, as a term can appear the same number of times in different classes (P, NONE, and N), so this term would not provide much information. Thus, the IG [41,47] of a Tweet word measures the number of bits of information obtained for the prediction of a class (C) by knowing the presence or absence of a term (t) in a Tweet. IG is, thus, a measure that summarises how common a word is in a given class compared to how common the same word is in the other classes of the corpus. To understand the meaning of Information Gain (IG), suppose we have two words: $word_1$ and $word_2$. If $word_1$ has a higher IG value than $word_2$, $word_1$ will be more useful in the training of our model because it increases information and reduces uncertainty. To calculate the IG of a word, we first need to calculate the entropy [45], the formula of which is (1):

$$H = \sum_{i=1}^{n} p_i \cdot \log_2(p_i) \tag{1}$$

where p_i is the probability, $P(w_i|c_i)$, that the word, w_i, appears in the class, c_i. Then, to calculate the Information Gain (IG), the formula is (2):

$$IG(C, X) = H(C) - H(C, X) \tag{2}$$

where C is the set of classes and X is the subset of texts in which the term w_i appears. To know $H(c)$, we first need to compute the probabilities of each class within the corpus. To know $H(C, X)$, we need to obtain the probabilities of the word appearing and not appearing in the corpus, as well as the probabilities of appearing and not appearing in each of the corresponding classes.

Thus, the concept of IG is applied in our case to retain the best vocabulary words and to check whether it improves the results of our hybrid classifier model. As can be seen in Table 3, the entropy gain improves the accuracy of the model, and the best result is obtained with a vocabulary size lower than the initial one (nearly 45,000), especially by choosing 10,000 words. Indeed, an accuracy improvement of up to 2.7 percentage points can be observed by integrating the Information Gain (IG).

Table 3. Summary of test carried out to evaluate the hybrid model using IG.

Vocabulary Size	Cross-Validation Accuracy
5000	75.70%
10,000	**75.82%**
15,000	75.70%
20,000	75.58%
45,000	73.14%

At this point, in order to validate the initial configuration of the hybrid model, tests are carried out to evaluate the values of the hyperparameters related to the size and maximum length of the embedding layer, which indicate the input size of the embedding matrix, with the former indicating the columns and the latter the rows. The stability of the model is evident from Table 4, where no significant variations between different matrix sizes are observed. The optimal outcome is achieved with a dimension value of 200 and a maximum tensor length of 33, which affirms that the initial values used in the preliminary configuration of the model were effective.

Table 4. Summary of tests carried out to evaluate the hybrid model by modifying the size and maximum length of the embedding layer.

Dimension Size	Input Tensor Size	Cross-Validation Accuracy
100	22	75.32%
100	33	75.70%
100	44	75.45%
200	22	75.19%
200	**33**	**75.82%**
200	44	75.60%
300	22	75.22%
300	33	75.67%
300	44	75.52%

Finally, the batch size parameter defines the amount of data the model has in each iteration, so it will be analysed how the modification of this value affects the model. The values of Table 5 are typical values used as standards in other models. As can be seen in this table, there are no major differences in terms of accuracy results; however, in execution times, there is a relationship between higher BS values producing shorter execution times, since in each iteration, the model has more data and, therefore, takes less time to process the information. Finally, looking at the accuracy values in Table 5, a BS value of 256 is taken.

Table 5. Summary of tests carried out to evaluate the hybrid model by modifying the batch size.

Batch Size	Time (min)	Cross-Validation Accuracy
32	14	75.38%
64	7	75.10%
128	6	75.78%
256	3	75.82%
512	2	75.62%

Once the whole model tuning process is completed, the final architecture is obtained, as shown in Figure 2. In summary, this deep learning model corresponds to an LSTM-based RNN architecture consisting of five layers with their corresponding hyperparameters. The five layers are as follows: an embedding layer, a one-dimensional convolutional layer (Conv1D), a MaxPooling layer, an LSTM layer, and a dense layer with an Adam optimiser. The hyperparameters were carefully selected using a 10-fold cross-validation technique, where the batch size was set to 256. The embedding layer takes input tensors of size 33 and produces vectors of length 200. The convolutional layer comprises 192 filters with a dropout rate of 0.2, while the LSTM layer consists of 96 neurons with a dropout rate of 0.3 and a kernel size of 8. Finally, the dense layer consists of as many neurons as classes we want to classify, that is, three neurons in the case of classifying three polarity categories (P, N, or NONE) or two neurons in the case of classifying two polarity categories (P or N). In addition, this layer shall use the SoftMax activation function to transform the outputs into a representation in the form of probabilities. This detailed description of the deep learning model, as well as its layers and hyperparameters, will help other researchers reproduce and apply the model.

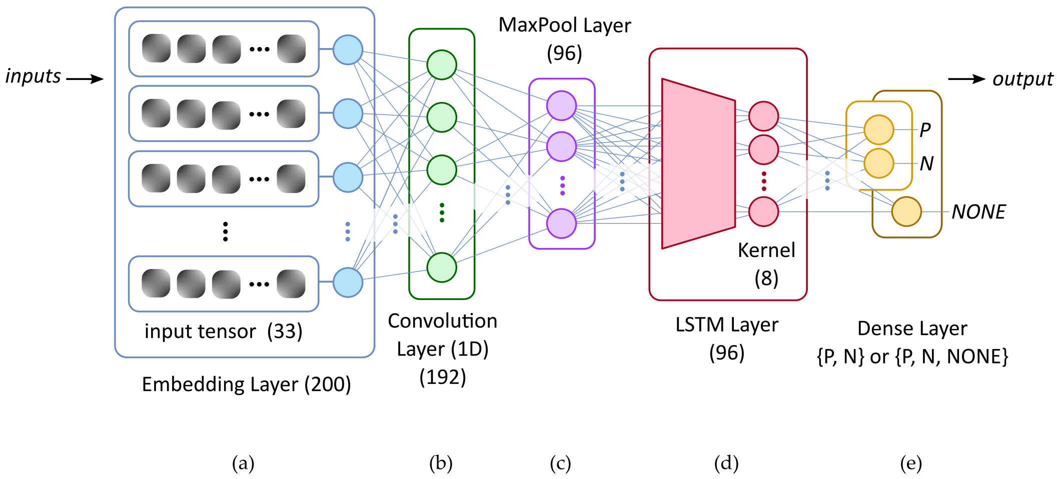

Figure 2. Final composition of the hybrid deep learning network model: (**a**) embedding layer, (**b**) one-dimensional convolutional layer (Conv1D), (**c**) MaxPooling layer, (**d**) LSTM layer, and (**e**) dense layer.

5. Discussion

Although accuracy measures the percentage of total cases where the model has predicted correctly, in our case, 75.82% for three classes (P, N, and NONE), it is worth considering some additional metrics related to the evaluation of the model on each class separately. Such metrics seem especially important when the dataset may be unbalanced. Table 6 shows the values of the precision, recall, and F1-score metrics. Precision refers to how close the outcome of a prediction is to the true value. It is, therefore, the ratio of the positive cases well classified by the model to the total number of positive predictions. Thus, precision gives the quality of the prediction, that is, the percentage of those that we have said are the positive class and actually are. At the same time, the recall measure gives us the quantity, that is, the percentage of the positive class that we are able to identify. Finally, the F1-score metric combines the two previous measures in a weighted way. To analyse the model's performance in detecting different classes, Table 6 presents the metrics corresponding to each class (P, N, and NONE). This allows us to determine which classes the model excels at in detecting and which ones require improvement. The results indicate that the negative class (N) performs the best, followed by the positive class (P), while the class representing no sentiment (NONE) exhibits the lowest performance.

Table 6. Summary of the results for precision, recall, and F1-score for three classes (P, N, and NONE).

Classes	Precision	Recall	F1-Score
P	74%	77%	75%
NONE	71%	71%	71%
N	82%	79%	80%

The confusion matrix serves as another valuable technique for evaluating these results, as it provides insights into the model's performance and helps to identify cases of class confusion. The diagonal of the matrix corresponds to the accurately classified values for each class, while the columns represent the predicted values and the rows represent the actual values. This matrix enables us to visualise and analyse the performance of the model in distinguishing between the different classes. The most striking thing that emerges from the confusion matrix of our model, as shown in Figure 3, is that the model fails most in the no sentiment class (NONE). Specifically, the worst levels of classification occur when the model predicts tweets as NONE and they are P, and in the case where the model predicts them as N but they are actually NONE.

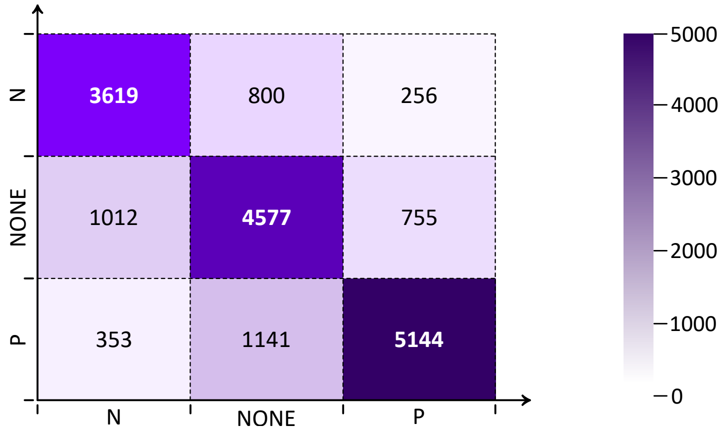

Figure 3. Confusion matrix of our hybrid model for the 3 considered classes (P, NONE, N), showing the number of cases assigned to each input class. The x-axis represents the real values, and the y-axis represents the classification or predicted values.

Finally, we will analyse the performance of our model by considering two classes (P and N), as these results are particularly significant. Tweets without polarity (NONE) do not provide significant information, while extreme polarities can play a more pivotal role in the analysis discourses or topics on social networks. Therefore, focusing on the performance of the model specifically for positive (P) and negative (N) classes will provide valuable insights for our analysis. The experiments were conducted by focusing on two classes (P and N), and our model achieved a high cross-validation accuracy of approximately 91% for the detection of these two classes. Furthermore, Table 7 shows the results of several metrics (accuracy, recall, and F1 score) for each class separately (P and N), where it can be observed that the negative class (N) performs slightly better than the positive class (P) for all metrics. In fact, our hybrid model achieves very good results, close to 90%, for both classes on all metrics. Therefore, it can be concluded that the results greatly improve the prediction of the three classes since the NONE class was the one with the most failures.

Table 7. Summary of the results for precision, recall, and F1-score for two classes (P, N).

Classes	Precision	Recall	F1-Score
P	90%	88%	89%
N	91%	93%	92%

When attempting to identify patterns in large datasets, one issue that often arises is data imbalance. The dataset is unlikely to be naturally balanced, so this difficulty occurs because the "minority classes" are under-represented compared to the "majority classes". This imbalance can negatively affect the predictive performance of machine learning models. To address this problem, we tested our deep learning model under different scenarios, including the original scenario, which featured a slightly imbalanced dataset with 22,021 positive tweets, 15,748 negative tweets (the minority class), and 21,094 neutral tweets. We also tested the model on a fully balanced dataset by randomly selecting 15,000 and 20,000 tweets from each class (positive, negative, and neutral). Table 8 shows the accuracy results of the cross-validation, which demonstrate that the performance of the algorithm is comparable across all scenarios, including the unbalanced ones. As a result, our hybrid model can efficiently handle unbalanced datasets and achieve the same level of performance regardless of class distribution.

Table 8. Accuracy results for unbalanced and balanced classes in the corpus considering 3 classes (P, N, NONE).

Number of Tweets in Each Class	Cross-Validation Accuracy
20,000	75.54%
15,000	75.62%
Unbalanced	75.82%

In summary, our hybrid deep learning algorithm demonstrates superior performance compared to other machine learning techniques when applied to Twitter data, as evidenced by previous research. Specifically, our approach significantly enhances the accuracy of machine learning models designed to identify sexist behaviour on Twitter, outperforming a method presented in [30] that relied on SVM, RF, LSTM, and BERT. While this method achieved accuracies in the range of 61–74%, our algorithm consistently achieves values around 91%, which represents a substantial improvement of up to 20 percentage points. Furthermore, our algorithm also excels when compared to the approach introduced in [31] for detecting hate speech on Twitter using BERT, achieving accuracies within the range of 79–86%. Our algorithm outperforms this method by 5–11 percentage points. Moreover, we conducted a comparison of our hybrid model with other algorithms using the same Twitter dataset (InterTASS, developed by SEPLN). Our algorithm notably outperformed all of these existing models. Specifically, it substantially enhances the accuracy of the algorithms proposed in [33,35], which relied on SVM, CNN, and LSTM, achieving only modest accuracies of between 54% and 59%. Additionally, our algorithm improves the approach described in [36], which achieved accuracies between 62% and 70%. In contrast, our algorithm consistently reaches accuracy levels of 91%, representing a substantial improvement of 20 to 36 percentage points over these prior models. Furthermore, when compared to alternative deep learning strategies, our algorithm once again stands out with its remarkable performance. For instance, the approach outlined in [34] only attains an accuracy of 57%, while our algorithm achieves an impressive 91% accuracy, representing an enhancement of 34 percentage points.

6. Conclusions

The ever-increasing amount of information and opinions available on social networks has made it imperative to develop automatic methods for effective information classification and analysis. Sentiment analysis (SA) in social networks has, therefore, become a crucial process in numerous sectors at both social and business levels. However, SA systems face unique challenges when analysing social media content, such as short message lengths, lack of context, poor grammar usage, and informal language style, as seen on platforms like Twitter. These challenges are further complicated in certain languages, such as Spanish, where the detection of negation, sarcasm, or ambiguity is crucial in sentiment analysis. Furthermore, given that Spanish is the second most used language on digital platforms and social networks, there is a critical need for further research in SA within Spanish, particularly in social networkcontexts where the complexities of the language are heightened. Consequently, we have developed a novel machine learning method based on deep learning that outperforms traditional models. Currently, there are very few proposals in Spanish for sentiment analysis in social networks using deep learning, and those that exist have room for further optimisation to improve their performance. Indeed, we propose a hybrid deep learning model that combines convolutional and LSTM layers to exploit the advantages of both, allowing us to detect the polarity of Twitter opinions through a labelled corpus. Our model improves upon previous existing models, achieving accuracy levels of approximately 76% and 91% for three and two polarities, respectively. Compared to other machine learning algorithms developed for Twitter sentiment analysis, which have accuracies ranging from 54 to 59%, our algorithm boosts its performance to 76% (an improvement of up to 20 percentage points). Furthermore, when compared to deep learning algorithms, which reach accuracies of 57%, our model performs better, improving levels by 20 percentage points.

However, some aspects remain to be studied in future work. The first may be to investigate how our classifier is able to respond to the irony or ambiguity present in Spanish tweets. At the same time, as future work, we plan to apply our model to several Spanish datasets as a way to investigate the robustness and generalizability of our approach.

Author Contributions: Conceptualisation, N.M. and J.V.; methodology, N.M. and J.V.; software, N.M. and J.V.; resources, N.M.; writing—original draft, N.M., J.V., C.L. and P.F.; writing—review and editing, N.M., J.V., C.L. and P.F.; visualisation, C.L. and P.F.; funding acquisition, N.M., J.V., C.L. and P.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the University of Valladolid.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors want to acknowledge D. Jesús Herreros Llanos for his contribution to the early development stages of the software used in this work.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kemp, S. *Digital 2022: Global Overview Report*; Techreport; Kepios Pte. Ltd.: Singapore, Singapore, 2022. Available online: https://datareportal.com/reports/digital-2022-global-overview-report (accessed on 6 October 2023).
2. Mäntylä, M.V.; Graziotin, D.; Kuutila, M. The evolution of sentiment analysis—A review of research topics, venues, and top cited papers. *Comput. Sci. Rev.* **2018**, *27*, 16–32. [CrossRef]
3. Martínez-Cámara, E.; Martín-Valdivia, M.T.; Urena-López, L.A.; Montejo-Ráez, A.R. Sentiment analysis in Twitter. *Nat. Lang. Eng.* **2014**, *20*, 1–28. [CrossRef]
4. Cervantes, I. *El español en el Mundo 2021. Anuario del Instituto Cervantes*; El español en el Mundo, Instituto Cervantes: Madrid, Spain, 2021. 2021. Available online: https://cvc.cervantes.es/lengua/anuario/anuario_21/ (accessed on 6 October 2023).
5. Perez-Rosas, V.; Banea, C.; Mihalcea, R. Learning Sentiment Lexicons in Spanish. In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, 23–25 May 2012.
6. Betancourt, E.R.; Chacón, P.S.; Murillo, E.C. Deep Neural Network Comparison for Spanish Tweets Polarity Classification. In Proceedings of the 2019 XLV Latin American Computing Conference (CLEI), Panama, Panama, 30 September–4 October 2019; pp. 1–6. [CrossRef]
7. Srivastava, R.; Bharti, P.; Verma, P. Comparative Analysis of Lexicon and Machine Learning Approach for Sentiment Analysis. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*, 71–77. [CrossRef]
8. Köhn, A. What is in an Embedding? Analyzing Word Embeddings through Multilingual Evaluation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 19–21 September 2015; pp. 2067–2073. . [CrossRef]
9. García-Cumbreras, M. Á.; García-Vega, M.; Gutiérrez, Y.; Martínez Cámara, E.; Piad-Morffis, A.; Villena-Román, J. TASS 2018: The Strength of Deep Learning in Language Understanding Tasks. *Proces. Leng. Nat.* **2019**, *62*, 77–84.
10. SEPLN. TASS: Workshop on Semantic Analysis at SEPLN. Sociedad Española para el Procesamiento del Lenguaje Natural, 2020. Available online: http://tass.sepln.org/2020/ (accessed on 6 October 2023).
11. Molina-González, M.D.; Martínez-Cámara, E.; Martín-Valdivia, M.T.; Perea-Ortega, J.M. Semantic orientation for polarity classification in Spanish reviews. *Expert Syst. Appl.* **2013**, *40*, 7250–7257. [CrossRef]
12. Hu, M.; Liu, B. Mining and Summarizing Customer Reviews. In Proceedings of the KDD '04: Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattlem WA, USA, 22–25 August 2004; pp. 168–177. [CrossRef]
13. Redondo, J.; Fraga, I.; Padrón, I.; Comesaña, M. The Spanish adaptation of ANEW (Affective Norms for English Words). *Behav. Res. Methods* **2007**, *39*, 600–605. [CrossRef] [PubMed]
14. Cruz, F.L.; Troyano, J.A.; Pontes, B.; Ortega, F.J. Building layered, multilingual sentiment lexicons at synset and lemma levels. *Expert Syst. Appl.* **2014**, *41*, 5984–5994. [CrossRef]
15. Moreno-Ortiz, A.; Hernández, C.P. Lexicon-based sentiment analysis of Twitter messages in Spanish. *Proces. Leng. Nat.* **2013**, *50*, 93–100.

16. Sidorov, G.; Miranda-Jiménez, S.; Viveros-Jiménez, F.; Gelbukh, A.; Castro-Sánchez, N.; Velásquez, F.; Díaz-Rangel, I.; Suárez-Guerra, S.; Treviño, A.; Gordon, J. Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets. In *Proceedings of the Advances in Artificial Intelligence*; Batyrshin, I., González Mendoza, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 1–14.
17. Brooke, J.; Tofiloski, M.; Taboada, M. Cross-linguistic sentiment analysis: From English to Spanish. In Proceedings of the International Conference RANLP-2009, Borovets, Bulgaria, 14–16 September 2009; pp. 50–54.
18. Taboada, M.; Brooke, J.; Tofiloski, M.; Voll, K.; Stede, M. Lexicon-Based Methods for Sentiment Analysis. *Comput. Linguist.* **2011**, *37*, 267–307. [CrossRef]
19. Vilares, D.; Alonso, M.A.; Gómez-Rodríguez, C. On the usefulness of lexical and syntactic processing in polarity classification of Twitter messages. *J. Assoc. Inf. Sci. Technol.* **2015**, *66*, 1799–1816. [CrossRef]
20. Jiménez-Zafra, S.M.; Martín-Valdivia, M.T.; Martínez-Cámara, E.; Ureña-López, L.A. Combining resources to improve unsupervised sentiment analysis at aspect-level. *J. Inf. Sci.* **2016**, *42*, 213–229. [CrossRef]
21. Jimenez Zafra, S.M.; Martin Valdivia, M.T.; Martinez Camara, E.; Urena Lopez, L.A. Studying the Scope of Negation for Spanish Sentiment Analysis on Twitter. *IEEE Trans. Affect. Comput.* **2019**, *10*, 129–141. [CrossRef]
22. Henríquez, C.; Guzmán, J.; Salcedo, D. Opinion mining based on the Spanish adaptation of ANEW on opinions about hotels. *Proces. Leng. Nat.* **2016**, *56*, 25–32.
23. Amores, M.; Arco, L.; Barrera, A. Efectos de la negación, modificadores, jergas, abreviaturas y emoticonos en el análisis de sentimiento. In Proceedings of the IWSW, Montréal, QB, Canada, 11–15 April 2016; pp. 43–53.
24. Anta, A.F.; Chiroque, L.N.; Morere, P.; Santos, A. Sentiment analysis and topic detection of Spanish tweets: A comparative study of of NLP techniques. *Proces. Leng. Nat.* **2013**, *50*, 45–52.
25. Gamallo, P.; Garcia, M.; Fernández-Lanza, S. TASS: A Naive-Bayes strategy for sentiment analysis on Spanish tweets. In Proceedings of the Workshop on Sentiment Analysis at Sepln (TASS2013), Madrid, Spain, 20 September 2013; pp. 126–132.
26. Montenegro, O.; Pabón, O.S.; De Piñerez R., R.E.G. A Deep Learning Approach for Negation Detection from Product Reviews written in Spanish. In Proceedings of the 2021 XLVII Latin American Computing Conference (CLEI), Cartago, Costa Rica, 25–29 October 2021; pp. 1–6. .. [CrossRef]
27. Myska, V.; Burget, R.; Povoda, L.; Dutta, M.K. Linguistically independent sentiment analysis using convolutional-recurrent neural networks model. In Proceedings of the 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 1–3 July 2019; pp. 212–215. .. [CrossRef]
28. Cerón-Guzmán, J.A.; León-Guzmán, E. A sentiment analysis system of Spanish tweets and its application in Colombia 2014 presidential election. In Proceedings of the 2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (Socialcom), Sustainable Computing and Communications (Sustaincom) (BDCloud-Socialcom-Sustaincom), Atlanta, GA, USA, 8–10 October 2016; pp. 250–257. .. [CrossRef]
29. Chollet, F. *Deep Learning with Python*; Manning Publications Co.: Shelter Island, NY, USA, 2021.
30. Rodríguez-Sánchez, F.; Carrillo-de Albornoz, J.; Plaza, L. Automatic classification of sexism in social networks: An empirical study on twitter data. *IEEE Access* **2020**, *8*, 219563–219576. [CrossRef]
31. Plaza-Del-Arco, F.M.; Molina-González, M.D.; Ureña-López, L.A.; Martín-Valdivia, M.T. A multi-task learning approach to hate speech detection leveraging sentiment analysis. *IEEE Access* **2021**, *9*, 112478–112489. [CrossRef]
32. Urpay-Camasi, J.; Garcia-Calderon, J.; Shiguihara, P. A Method to Construct Guidelines for Spanish Comments Annotation for Sentiment Analysis. In Proceedings of the 2021 IEEE Sciences and Humanities International Research Conference (SHIRCON), Lima, Peru, 17–19 November 2021; pp. 1–4. .. [CrossRef]
33. Chiruzzo, L.; Rosá, A. RETUYT-InCo at TASS 2018: Sentiment Analysis in Spanish Variants using Neural Networks and SVM. In Proceedings of the TASS@ SEPLN, Sevilla, Spain, 18 September 2018; pp. 57–63.
34. Pastorini, M.; Pereira, M.; Zeballos, N.; Chiruzzo, L.; Rosá, A.; Etcheverry, M. RETUYT-InCo at TASS 2019: Sentiment Analysis in Spanish Tweets. In Proceedings of the IberLEF@ SEPLN, Bilbao, Spain, 24 September 2019; pp. 605–610.
35. González, J.Á.; Lluís-F. Hurtado, F.P. ELiRF-UPV at TASS 2018: Sentiment Analysis in Twitterbased on Deep Learning. In Proceedings of the TASS 2018: Workshop on Semantic Analysis at SEPLN. Sociedad Española para el Procesamiento del Lenguaje Natural, Sevilla, Spain, 18 September 2018.
36. Pla, F.; Hurtado, L.F. Sentiment analysis in Twitter for Spanish. In Proceedings of the Natural Language Processing and Information Systems: 19th International Conference on Applications of Natural Language to Information Systems, NLDB 2014, Montpellier, France, 18–20 June 2014; pp. 208–213.
37. Shazly, K.; Eid, M.; Salem, H. An Efficient Hybrid Approach for Twitter Sentiment Analysis based on Bidirectional Recurrent Neural Networks. *Int. J. Comput. Appl.* **2020**, *175*, 32–36. [CrossRef]
38. Li, A.; Yi, S. Emotion analysis model of microblog comment text based on CNN-BiLSTM. *Comput. Intell. Neurosci.* **2022**, *2022*, 1669569. [CrossRef] [PubMed]
39. Mewada, A.; Dewang, R.K. SA-ASBA: A hybrid model for aspect-based sentiment analysis using synthetic attention in pre-trained language BERT model with extreme gradient boosting. *J. Supercomput.* **2023**, *79*, 5516–5551. [CrossRef]
40. Shah, S.; Ghomeshi, H.; Vakaj, E.; Cooper, E.; Mohammad, R. An Ensemble-Learning-Based Technique for Bimodal Sentiment Analysis. *Big Data Cogn. Comput.* **2023**, *7*, 85. [CrossRef]
41. Witten, I.H.; Frank, E.; Hell, M.A. *Data Mining*, 3rd ed.; Morgan Kaufmann: Burlington MA, USA, 2011.

42. Chollet, F.; Keras Team. Keras. 2022. Available online: https://github.com/fchollet/keras-resources (accessed on 20 April 2022).
43. Team, G.B. TensorFlow Library. 2022. Available online: https://keras.io/ (accessed on 6 October 2023).
44. SEPLN 2022. Sociedad Española para el Procesamiento del Lenguaje Natural, A Coruña, Spain, 20–23 September 2022. Available online: https://sepln2022.grupolys.org/ (accessed on 6 October 2023).
45. Bird, S.; Klein, E.; Looper, E. NKLT-Natural Language Processing Library. Natural Language Toolkit Project. 2022. Available online: https://www.nltk.org (accessed on 6 October 2023).
46. Developers, S. Snowball algorithms for stemming. Python Package Index. 2022. Available online: https://snowballstem.org/ (accessed on 6 October 2023).
47. Larose, D.T.; Larose, C.D. *Discovering Knowledge in Data: An Introduction to Data Mining*; John Wiley & Sons: Hoboken, NJ, USA, 2014; Volume 4, pp. 174–179.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.