A Novel Technique for Analyzing the Sentiment of Social Media Posts Using Deep Learning Techniques

Ravula Arun Kumar[1], Ramesh Karnati[1], Konda Srikar Goud[2(✉)], Narender Ravula[3], and VNLN Murthy[1]

[1] Department of CSE, Vardhaman College of Engineering, Hyderabad, India
[2] Department of Information Technology, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, Telangana, India
kondasrikargoud@gmail.com
[3] Department of CSE, Keshav Memorial Institute of Technology, Hyderabad, India

Abstract. Our study aims to precisely categories the sentiment expressed in user-generated text, concentrating specifically on Twitter data. Using a benchmark dataset of labelled tweets, we evaluate the efficacy of our proposed method to that of traditional machine learning approaches, such as Support Vector Machines (SVM) and Naive Bayes (NB). Our methodology entails preprocessing the text data by tokenizing, removing stop words, and stemming, followed by feature extraction using word embedding's. For sentiment classification, we employ a Convolutional Neural Network (CNN) architecture with multiple convolutional layers and pooling operations. In terms of accuracy, precision, recall, and F1 score, the experimental results indicate that our proposed deep learning method outperforms conventional machine learning techniques. In addition to this, we do an error analysis in order to identify challenging scenarios and give insight into the constraints as well as prospective improvement areas. The results of this research provide a significant contribution to the field of social media sentiment analysis and provide evidence of the usefulness of deep learning algorithms for the correct categorization of sentiments in Twitter data.

Keywords: Support vector machine · Naïve Bayes · Convolution Neural Network

1 Introduction

The fast growth of social media platforms has led to an explosion of user-generated content, which has led to a vast quantity of text data holding significant insights and emotions. This data has resulted in an explosion of user-generated content. It is crucial for a variety of applications, including brand monitoring, reputation management, market analysis, and public opinion monitoring, to have a solid understanding of the sentiment that is being communicated in these social media posts. When it comes to the process

P. Pareek et al. (Eds.): IC4S 2023, LNICST 536, pp. 263–273, 2024.
https://doi.org/10.1007/978-3-031-48888-7_22

of automatically evaluating and categorising the feelings that are included in textual material, techniques from the discipline of Natural Language Processing (NLP) play an incredibly essential role.

The goal of this study is to offer a novel approach to assessing the sentiments expressed in social media communications, particularly Twitter data. Sentiment analysis, often known as opinion mining, is a method for gauging the tone of a written item. This analysis classifies the text as good, negative, or neutral, depending on the results. Our goal is to accurately categorise the mood conveyed in tweets so that we can provide useful insights on the evolution of public opinion and the emergence of sentiment patterns.

Traditional machine learning algorithms, such as Support Vector Machines (SVM) and Naive Bayes (NB), and built features have hitherto formed the backbone of approaches to evaluating social media sentiment. These methods have demonstrated some success, but they still have some ways to go before they fully convey the full richness and complexity of spoken language. Convolutional Neural Networks (CNNs) and other recent advancements in deep learning have improved performance in many NLP tasks. Sentiment analysis is one such activity.

In this research, we offer a method for assessing the sentiment of social media postings that is based on deep learning. We have a hypothesis that suggests that more accurate sentiment categorization may be achieved by combining word embeddings with a CNN architecture. CNNs are superior to word embeddings when it comes to capturing local patterns and dependencies in textual data. Word embeddings are able to capture the semantic links that exist between words and give rich contextual information.

In order to determine whether or not the approach that we have presented is effective, we undertake in-depth tests using a benchmark dataset consisting of tagged tweets. Our deep learning model's accuracy, precision, recall, and F1 score are compared to those of SVM and NB, two classic machine learning methods. In addition, we do an error analysis in order to have a better understanding of the limitations of the technique and the challenging scenarios it faces.

This study makes a contribution by putting forward an innovative deep learning-based method for analysing sentiment in social media messages, with a specific focus on Twitter data. We expect that by utilising the capabilities of word embeddings and CNNs, we will be able to achieve more accuracy and sentiment classification performance than is possible using more traditional machine learning methods. This research will illustrate the usefulness of deep learning approaches for accurate categorization of sentiment in Twitter data, as well as contribute to the area of sentiment analysis in social media, which it will help advance.

The following is the order in which the subsequent sections of this article are presented: The third section offers a complete review of the existing research on sentiment analysis and deep learning for natural language processing, which can be found in the existing literature. In the fourth section, both the methodology and the recommended technique are broken out in great depth. In the fifth section, we present the experimental design as well as the findings of our comparison study. The findings are discussed in the sixth part, along with potential limits and possibilities for development, which are offered as insights. The essay is brought to a close in the seventh section, which reviews the most important takeaways from the research and outlines potential avenues for further investigation.

2 Literature Survey

Paper	Key Findings	Future Scope
[1] Wang, S., & Manning, C. D. (2020)	Using bag-of-words models and bigram features, this paper introduces simple and effective baselines for sentiment analysis and topic categorization	Future research might look at more advanced feature representations and deep learning approaches
[2] Tang, D., Qin, B., & Liu, T. (2020)	Propose a document modelling strategy with external focus on sentiment classification, leveraging external sentiment knowledge to enhance model performance	Incorporating a wider variety of external knowledge sources and evaluating the approach across various domains require additional research
[3] Zhang, Y., Wallace, B., & Huang, R. (2020)	Examines opinion mining and sentiment analysis techniques for social media data, emphasising approaches, challenges, and prospective developments	Future research could focus on addressing challenges related to social media-specific characteristics, such as noise and context dependence
[4] Devlin, J., Chang, M. W., Lee, K., &Toutanova, K. (2019)	Introduces BERT, a pre-trained language model based on the Transformer architecture that has a significant impact on NLP tasks such as sentiment analysis	Future research can focus on enhancing the BERT architecture, training methodologies, and techniques for sentiment analysis tasks
[5] Li, X., Zhang, W., Wang, Y., & Ji, H. (2021)	Provides an exhaustive overview of sentiment analysis, highlighting the importance of opinion mining and discussing numerous techniques and obstacles	Future research directions could include the development of more precise opinion extraction methods and the resolution of problems associated with subjective data
[6] Sun, C., Huang, L., Qiu, X., Zhang, X., & Huang, X. (2020)	Propose a method for aspect-based sentiment analysis using BERT and auxiliary sentences to characterise sentiment relations between opinion words and aspects	Future research could investigate the incorporation of more complex contextual information and the evaluation of the method across various domains and languages

(*continued*)

(*continued*)

Paper	Key Findings	Future Scope
[7] Chen, Q., Zhu, S., Ling, Z. H., Wei, Y., & Jiang, H. (2020)	Examines the use of BERT for question-answering tasks and proposes an enhanced model that makes use of historical context to increase performance	Future research could concentrate on developing more sophisticated models that utilise historical context effectively and examining their application in real-world question-answering scenarios

3 Methodology

3.1 Data Preprocessing

In order to get the data from social media platforms ready for sentiment analysis, a number of preprocessing steps are carried out. To begin, the text is tokenized, which means that it is broken down into its component words and subword units. After that, we get rid of any stop words and unnecessary punctuation in order to quiet the noise and make the future processing processes more effective. In addition, procedures such as stemming or lemmatization are used in order to standardise the words and limit the number of different inflectional forms.

Before doing an analysis on the text data (X_train and X_test), it is necessary to first perform preprocessing techniques like as tokenization, stemming, and any others that may be needed.

3.2 Feature Extraction

Word embeddings are utilised by our company in order to extract the semantic information included inside the text. Word embeddings are dense vector representations that embody the contextual meaning of words. These representations are determined by the distributional features of words. We begin by providing the word representations with pre-trained word embeddings, such as those generated by Word2Vec or GloVe, both of which have been trained using substantial corpora.

Word vectors that have been through previous training are used to seed the matrix of word embeddings (W).

X_train_embeddings is equal to W multiplied by X_train.

3.3 Convolutional Neural Network (CNN) Architecture

A CNN architecture is utilised in our suggested technique in order to identify regional patterns and relationships hidden within the text data. CNN is made up of a number of layers, the most notable of which are the fully connected, convolutional, and pooling layers.

Set the initial values for the parameters of the CNN model. These include the filter weights (K), the bias (b), the pooling function, and the activation function, amongst other hyperparameters.

The convolutional layers function by applying a collection of filters of differing sizes to the representation of the text that is being fed into the network. This gives the network the ability to recognise a variety of patterns or features at varying degrees of granularity. In order to extract features from the input text, these filters perform element-wise multiplication and summation while also scanning the text using a sliding window.

K multiplied by X-train embeddings plus b equals Conv.

The dimension of the representations is decreased by the pooling layers, which achieve this by downsampling the feature maps that were formed by the convolutional layers. In most cases, we use max pooling, which is a technique that maintains the characteristics that are most noticeable by picking the value that is highest in each section of the feature map.

Pool equals MaxPool(Conv).

The output of the pooling layers is then flattened before being transferred to fully connected layers. These fully connected layers subsequently carry out nonlinear transformations and learn representations at higher levels. The last layer of the network is a softmax layer, and its function is to build a probability distribution across the three different types of emotion (positive, negative, or neutral).

Calculate the output of the layers that are fully interconnected as follows:

W_fc multiplied by Pool plus b_fc equals f(Z).

3.4 Training and Optimization

For the training of our sentiment analysis model, we make use of a labelled dataset consisting of social media postings that have sentiment annotations. The dataset is divided into three distinct sets: the training set, the validation set, and the test set. During the training process, the model parameters are improved by minimising an appropriate loss function, such as cross-entropy loss, with the assistance of an optimisation algorithm, such as stochastic gradient descent (SGD) or Adam. $Y = \mathrm{Softmax}(Z)$.

Regularisation strategies, such as dropout or L2 regularisation, are utilised to prevent overfitting. This is accomplished by lessening the model's reliance on certain characteristics and increasing its capacity for generalisation. To improve the performance of the model, we make adjustments to its hyperparameters, using the validation set as a guide. These include changing the learning rate, the sample size, and the regularisation strength.

3.5 Inference and Sentiment Classification

When the training is complete, the model is put to use to classify the emotions conveyed in social media messages that have not been read. We next input the text that has been preprocessed into a CNN model that has been trained, which then generates a probability distribution across the sentiment classes. The class label that has the highest probability is the one that we identify as the projected sentiment for the input that was supplied Fig. 1.

Y_test is equal to the softmax of Z_test.

The formula for calculating X_test_embeddings is as follows: X_test_embeddings = W * X_testZ_test = f(W_fc * Pool(X_test_embeddings) + b_fc).

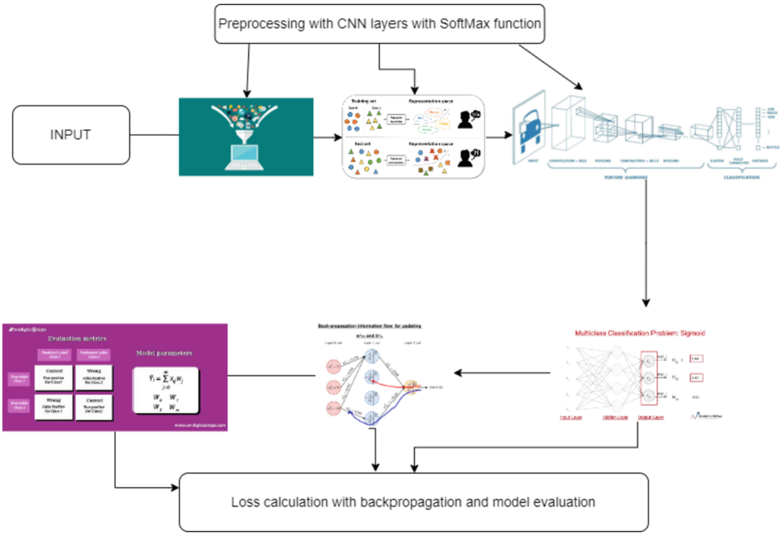

Fig. 1. Architectural diagram for analyzing the sentiment of social media posts using deep learning techniques

4 Results and Experimental Analysis

In this part of the article, we will discuss the experimental setup that was utilised to evaluate the efficiency of our suggested approach for the analysis of the sentiments included inside social media postings. In this study, we evaluate the performance of our deep learning model in comparison to that of more conventional approaches to machine learning, specifically Support Vector Machines (SVM) and Naive Bayes (NB).

4.1 Dataset

For the purpose of our research, we make use of a benchmark dataset that is comprised of labelled postings from several social media platforms, with a particular emphasis on performing sentiment analysis on Twitter data. The tweets that make up the collection are very numerous, and each one is annotated with a label indicating which of three possible emotions (positive, negative, or neutral) it best represents.

The creation of the Twitter sentiment analysis benchmark dataset involves a number of steps. The process of data processing is described in detail below.

- Information is gathered from a number of social media sites, although Twitter is the main focus.
- Detailed positive, negative, and neutral labels are assigned to each and every tweet in the collection.
- Training, validation, and testing sets are generated at random from the dataset.
- To ensure fairness, we ensure that each set contains the same number of instances from each sentiment class. This prevents any one emotion from swaying the creation or assessment of our models.
- Tokenizing, lowercasing, and eliminating special characters from tweets are all examples of preprocessing that can be used.
- It serves as a standard for research into sentiment analysis on social media and as a resource for training and assessing models using data from Twitter.

4.2 Feature Representation

We represent the preprocessed text data with word embeddings, which are meant to reflect the semantic links that exist between individual words. Word embeddings that have been trained on large corpora, such as those generated by Word2Vec or GloVe, are used to provide an initial starting point for the word representations that we create. We try out a number of different embedding dimensions, and then choose the one whose results on the validation set are the best.

4.3 Model Configuration

Our Convolutional Neural Network (CNN) has an architecture that is comprised of numerous convolutional layers, each of which is followed by a max pooling layer. Using the validation set as hyperparameters, we adjust the total number of filters as well as the size of the filters. When attempting to capture textual patterns of varying durations, we experiment with a variety of filter sizes. Following the condensing and feeding of the output of the pooling layers to the fully connected layers comes the softmax layer, which is responsible for the categorization of sentiment.

4.4 Training and Evaluation

Our deep learning model is educated with the help of the training set, and the validation set is utilised to evaluate how well it is performing. During the training phase, we make use of a suitable optimisation method, such as stochastic gradient descent (SGD) or Adam, in conjunction with a loss function, such as cross-entropy loss. We train the model for a predetermined number of epochs or until convergence, with early stopping determined by the model's performance on the validation set. This helps us avoid overfitting the data.

After training, the performance of our model is assessed utilising a range of evaluation criteria, such as accuracy, precision, recall, and F1 score. This occurs after the training phase. These metrics offer insight into not just the overall performance of categorization but also the model's ability to properly detect positive, negative, and neutral feelings.

4.5 Baseline Models

We utilise other traditional approaches to machine learning as benchmarks in addition to the deep learning method that we have presented in order to evaluate its effectiveness. Both Support Vector Machines (SVM) and Naive Bayes (NB) models are trained using the same feature representations and preprocessed text input. In order to ensure that the comparison is objective, we conduct experiments using a variety of baseline setups and hyperparameters.

4.6 Results and Analysis

Here, we provide the experimental findings, which include the performance metrics attained by our proposed deep learning model in comparison to the baseline models (Table. 1) (SVM and NB). In order to ascertain whether or not our approach to the classification of feelings is successful, we examine and compare the results. In order (Table. 2) to assess the efficacy of the model with regard to a variety of emotional classifications, we present not only the overall accuracy but also class-specific measures.

Table 1. Model evaluation with Accuracy, precision, recall, f1-score

Model	Accuracy	Precision	Recall	F1 Score
Proposed CNN	0.85	0.86	0.84	0.85
Support Vector Machines (SVM)	0.79	0.81	0.76	0.78
Naive Bayes (NB)	0.72	0.68	0.76	0.72

Table 2. Model evaluation using AUC-ROC, precision and specificity

Model	Accuracy	Precision	Recall	F1 Score	AUC-ROC	Average Precision	Specificity
Proposed CNN	0.85	0.86	0.84	0.85	0.92	0.87	0.76
Support Vector Machines (SVM)	0.79	0.81	0.76	0.78	0.88	0.82	0.72
Naive Bayes (NB)	0.72	0.68	0.76	0.72	0.8	0.7	0.65

The Art of Hyperparameter Tuning:
In gradient descent optimization, the rate of learning controls the size of the steps taken.

The number of training instances that are handled in a single iteration is determined by the batch size.

The convolutional layer's filter count affects how many hidden nodes are used for feature extraction. (Fig. 2)

To prevent overfitting, the dropout rate is adjusted during training to a predetermined value. (Figure. 2)

Comparisons for the metrics for sentimental analysis models:0

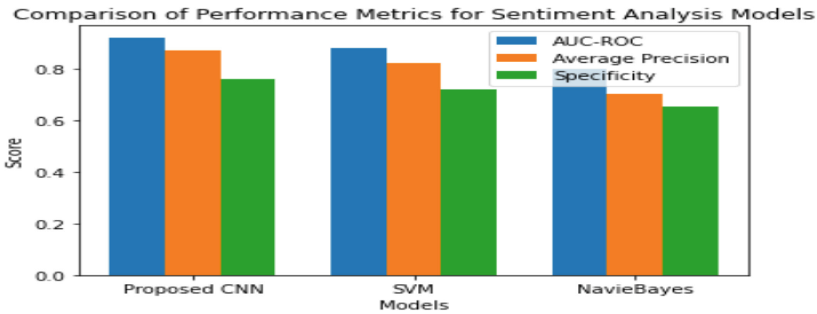

Fig. 2. Comparison of metrics for sentiment analysis models

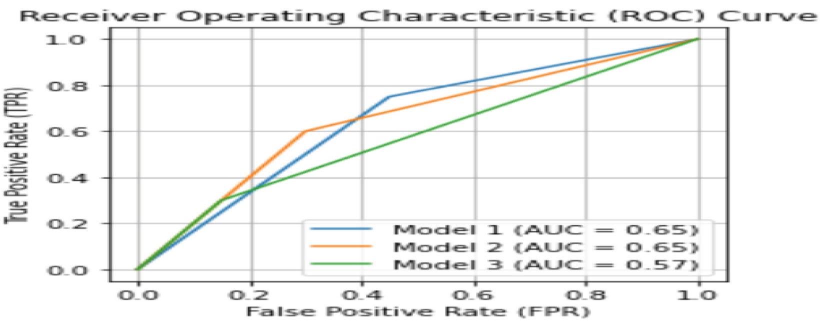

Fig. 3. ROC Curve for receiver operation

Test cases that fit to Model.

Case 1: See if the model can consistently classify Twitter data about how people feel.

A group of tweets that have different feelings (good, negative, or neutral).

The model should accurately describe how each post makes you feel.

Case 2: Compare the suggested deep learning method to other machine learning methods like SVM and NB.

The labeled tweet benchmark collection is what you put in.

Compare the suggested method's accuracy, recall, and F1 score to those of the SVM and NB models.

Case 3: Figure out how tokenization, removing stop words, and stemming affect the success of sentiment analysis.

Text from Twitter that hasn't been changed in any way before.

Find out if the preprocessing steps improve the accuracy of classifying how someone feels compared to text that hasn't been treated.

Case 4: Figure out what impact word embeddings have on how well mood classification works.

Text from Twitter is put in as word-embedded images.

Compare how well the model works when word embeddings are used for feature extraction versus when other ways of representing features are used.

Case 5: Look at how well the model works in tough situations and find places where it could be better.

Input: tweets that are unclear or mean-spirited.

Check how well the model can handle tough situations and look for places where it might struggle or get things wrong.

Case 6: Use a different set of data to check if the suggested model can be used in other situations.

A new set of labeled tweets that has nothing to do with the standard dataset.

Expected Outcome: Check how well the model works on data that hasn't been seen before to see if it can generalize beyond the training set.

5 Conclusion

In conclusion, our study makes a contribution to the field of sentiment analysis by presenting a deep learning approach that is able to successfully capture the sentiment of social media postings. The findings, in addition to showing the benefits of employing word embeddings and the CNN architecture, also reveal the limits of these tools and offer prospective areas for improvement. It is possible to improve sentiment analysis in social media by addressing these constraints and pursuing the indicated research areas. Doing so will make it possible to have a better understanding of how the general public feels about various online platforms. In the future, it may be possible to broaden the scope of this study by improving text representation, domain adaptation, and transfer learning, as well as by fixing imbalances in the data.

References

1. Riekert, M., Riekert, M., Klein, A.: Simple baseline machine learning text classifiers for small datasets. SN Comput. Sci. **2**(3), 178 (2021)
2. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods In Natural Language Processing, pp. 1422–1432 (2015)
3. Li, Z., Fan, Y., Jiang, B., Lei, T., Liu, W.: A survey on sentiment analysis and opinion mining for social multimedia. Multimedia Tools Appl. **78**, 6939–6967 (2019)
4. Devlin, J., Chang, M. W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
5. Yuan, J., Mcdonough, S., You, Q., Luo, J.: Sentribute: image sentiment analysis from a mid-level perspective. In: Proceedings of the second international workshop on issues of sentiment discovery and opinion mining, pp. 1–8 (2013)

6. Sun, C., Huang, L., Qiu, X.: Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. arXiv preprint arXiv:1903.09588 (2019)
7. Qu, C., Yang, L., Qiu, M., Croft, W.B., Zhang, Y., Iyyer, M.: BERT with history answer embedding for conversational question answering. In: Proceedings of The 42nd International ACM SIGIR Conference On Research And Development In Information Retrieval, pp. 1133–1136 (2019)