

# An Ensemble Machine Learning Approach for Twitter Sentiment Analysis

Pavlo Radiuk<sup>1</sup>, Olga Pavlova<sup>1</sup>, and Nadiia Hrypynska<sup>1</sup>

<sup>1</sup> Khmelnytskyi National University, 11, Instytut's'ka str., Khmelnytskyi, 29016, Ukraine

## Abstract

The presented study addresses the issue of classifying emotional expressions based on small texts (tweets) extracted from the social network Twitter. In this paper, we propose a novel approach to preprocessing tweets to fit them more effectively into the classification model. Moreover, we suggest utilizing two types of features, namely unigrams and bigrams, to expand the feature vector. The classification task of emotional expressions was performed according to several machine learning algorithms: raw random forest, gradient boosting random forest, support vector machine, multilayer perceptron, recurrent neural network, and convolutional neural network. The feature vector elements are presented as sparse and dense subvectors. As a result of computational experiments, it was found that the “appearance” in the reflection of the sparse vector provided higher performance than the “regularity.” The experiments also showed that deep learning approaches performed better than traditional machine learning techniques. Consequently, the best recurrent neural network achieved an accuracy of 83.0% on the test dataset, while the best convolutional neural network reached 83.34%. At the same time, it was discovered that the convolutional model with the support vector machine classifier showed better performance than the single convolutional neural network. Overall, the proposed ensemble method based on receiving the most votes according to the five best models’ predictions has reached an absolute accuracy of 85.71%, proving its practical usefulness.

## Keywords

Machine learning, deep learning, ensemble model, Twitter, sentiment analysis, sentiment classification

## 1. Introduction

The task of determining emotional expressions from text messages (tweets) on Twitter usually involves the use of advanced methods of sentiment text analysis in three categories: positive, negative, and neutral. This task also consists of analyzing opinions, dialogues, announcements, and news (within one thread of tweets) to establish business strategies [1], political analysis, assessments of public action [2], and so forth. Sentiment analysis has been widely used in identifying political and social trends based on micro-blogging [3]. It is an effective means of commercial and political marketing in social networks [4], as it allows for predicting user behavior on the Internet.

In recent years, the problem of natural language processing (NLP), which is a branch of deep learning (DL), and the problem of semantic text analysis have become especially valuable and widespread. One of the leading NLP approaches is to rank the importance of sentences in a text and words in a sentence [5] and then create a brief semantic review of the text, supported by critical figures. Information systems based on such approaches do not usually depend on manually predefined rules but instead on machine learning (ML) techniques that solve classification problems. At the same time, the

---

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland  
EMAIL: radiukpavlo@gmail.com (P. Radiuk); olya1607pavlova@gmail.com (O. Pavlova), grypynska@gmail.com (N. Hrypynska)  
ORCID: 0000-0003-3609-112X (P. Radiuk); 0000-0001-7019-0354 (O. Pavlova); 0000-0003-0103-976X (N. Hrypynska)



© 2022 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

problem of semantic text analysis is solved by an automatic system that returns one of the predefined categories based on separate samples of text.

The semantic features of the text are extracted based on sentiment analysis of the regularity distribution of speech parts for a particular category of marked tweets. It should be noted that the semantic features of Twitter are more informal than other types of texts. They relate to emotional expressions and tonality on online social platforms within a limited space of 280 characters. Twitter attributes include hashtags, retweets, capitalization, word extensions, question and exclamation marks, URLs, online emoticons, and online slang, all of which can be used for semantic analysis.

In recent years, dozens of businesses have conducted numerous sentiment analyses on Twitter to determine the attitudes of their users to a product or analyze the market overall. Many challenges occur while preprocessing textual data from short messages. For instance, a tweet containing a complaint text on Twitter can quickly escalate into a public relations crisis. An unsuccessful short joke can rapidly transform into controversy, causing a lot of negative emotions among a targeted audience. It might be difficult for responsible staff to manually notice possible issues or even the crisis before it commences. Therefore, this study aims to investigate modern NLP approaches that may facilitate sentiment analysis based on textual data from Twitter to efficiently assess and predict possible reputational failures of a business or social entity in real-time.

## 2. Related work

Over the past decades, sentiment analysis has been successfully applied to different sources of textual data, such as user reviews [6], medical data [7], web blogs [8], and highlighting key phrases [9]. However, data on Twitter is different due to the limit of 280 characters per tweet, which forces users to express narrowed opinions compressed into concise texts. The most prominent results in sentiment classification have been achieved with supervised learning techniques [10], i.e., gradient boosting random forest (XGBoost) and support vector machine (SVM); yet the manual labeling used for the supervised approach requires much time and may cause technical mistakes in labels.

The scientific community usually examines new classification features and techniques, comparing them with the baseline performance. As such, classification techniques make formal comparisons between these results to select the most effective classification techniques for specific applications. Utilizing unigrams and bigrams as features [11] for vectorization requires representing words in these  $n$ -grams by a particularly established polarity and then taking the average general polarity of the text.

Sentiment analysis of tweets has been comprehensively applied to recent challenges in all areas. For example, in work [12], the authors studied the public opinion on the vaccination of early virus pneumonia [13] on tweets posted between December 2021 to July 2021. The predictive model's performance was tested using several DL methods: recurrent neural network (RNN), long short-term memory (LSTM), and bidirectional LSTM. The highest 90.59% and 90.83% were obtained with LSTM and Bi-LSTM. Aspect-based sentiment analysis was used by [14] with six different emotional expressions on Twitter and four distinct BERT models [15]. The highest 87% was obtained by the proposed method. COVID-19 Arabic tweets are examined in [16] with 54,065 Twitter posts and four classifiers: random forest (RF), gradient boosting,  $k$ -nearest neighbor ( $k$ -NN), and SVM. Implementing an ensemble of all four classifiers provided the utmost accuracy of 89.12%. In [17], the authors examined the evolution of vaccine resistance by evaluating the Twitter discussion of the COVID-19 vaccine in the United States.

Much attention has been paid to the semantic analysis of socially significant problems. In study [18], the authors analyzed public concern in troll and bullying detection using Weibo posts on social media. The emotions are separated based Baidu emotions analysis tool. A lexicon-based technique was employed in study [19] to identify consumer attitudes towards recent sporting events. The Latent Dirichlet Allocation (LDA) extracts latent semantics patterns from Twitter posts. The most pessimistic and optimistic feelings are expressed by -1 and 1, respectively. Two lexicons, SentiwordNet AFINN with SVM classifier, were applied [20] for Twitter post classification. In [21], 20,325,929 pandemic-related Twitter posts were used to gauge public emotions using a lexicon technique for sentiment analysis. The CrystalFeel algorithm was employed to classify four feelings: fear, anger, sorrow, and joy. The authors in [22] utilized a hybrid technique to analyze 1,499,227 vaccine-related tweets from

March 18, 2019, to April 15, 2019, with an accuracy rate of more than 85%. In [23], a text-blob lexicon and Latent Dirichlet Allocation were used to study Indians' attitudes regarding COVID-19 immunization [24]. Study [25] suggested another conventional Naive Bayes-based ML technique for analyzing the general sentiments of Twitter data with 81.77%.

As it is seen from the overview above, intelligent analysis of micro-blogging using the ML and DL methods and means has become highly relevant. At the same time, since the type of textual data and the conditions for short text messages on Twitter constantly evolve, there is an urgent need to develop new techniques for semantic analysis of textual data on this platform. Therefore, to achieve this goal, the following tasks are to be resolved:

1. To investigate various machine learning and deep learning techniques for semantic analysis based on textual data.
2. To propose an approach to determine the semantics of short messages for micro-blogging.
3. To conduct computational experiments with the proposed approach and its analogs to categorize the polarity of tweets based on their semantics into positive and negative classes.
4. To validate the considered techniques according to the statistical measurements.

Thus, in this work, we investigate several classification models and propose a new ensemble ML approach to categorize the polarity of tweets based on their semantics into positive and negative classes.

### 3. Methods and materials

In this work, we utilized the manually crafted dataset of small texts from Twitter, which were labeled with two classes based on their semantics: positive or negative. This dataset consists of text messages, emoticons, usernames, and hashtags. These elements were first preprocessed and then converted into a vector form for further analysis.

#### 3.1. Data preparation

The targeted data is presented in files with two columns: text messages and corresponding labels indicating the messages' semantics. The training subset comprised tweet\_id, semantics, and emoticons that facilitated predicting polarity. It should be noted that URLs and user mentions were ignored and dropped. Here, the words within messages are meant as a mixture of words and phrases with errors, extra punctuation, and words with lots of repetitive letters. Therefore, tweets were preprocessed before semantic analysis to unify all objects in the dataset.

Raw text messages extracted from Twitter mostly contain huge data noise due to people's use of various lexemes and semantics to express their opinions on social networks. Tweets have unique characteristics, such as retweets, and emoticons, which must be suitably extracted. That is why the raw tweets should be normalized to construct a robust dataset. Several preprocessing steps were applied to the initial dataset to unify it and reduce its size. The first stage of the preprocessing was implied in the following steps: (a) converting a tweet to lower case; (b) replacing two or more dots with a single space; (c) removing spaces and quotes from texts; (d) substituting two or more spaces with a single space.

**URL.** Users often share hyperlinks to other web pages in their tweets. URLs were not essential for the text classification as they would lead to very sparse features. As a result, all URLs within tweets were replaced with the word "URL."

**Hashtag.** As a rule, hashtags (words with the hash prefix, #) do not reflect emotional semantics in short text messages [8]. Therefore, all words with the # symbol were replaced with the corresponding words without this symbol. For instance, #finance was superseded by finance.

**Emoticon.** Using a variety of smileys and emoticons in tweets to express emotions is an integral culture of communication among Twitter users. Due to the ever-increasing number of smileys and emoticons [11], it does not seem easy to compare and normalize them comprehensively. Therefore, only the most commonly used standard emoticons were used in this work for semantic analysis. As a result, all relevant smileys were divided into positive and negative ones and replaced by EMO\_POS or EMO\_NEG tokens.

After the initial preprocessing, the individual words were also processed as follows

- All punctuations like [?!,.:;] were extracted from the words.
  - Symbols -, -, \_, “, ” ‘, and ’ were eliminated within the whole text.
  - Two or more letter repetitions were converted exactly to two letters.
  - If the words began with the letters of the alphabet, followed by letters, numbers, dots, or underscores, such words remained in the text; any other words that did not fit these requirements were removed.
- Thus, all preprocessing techniques resulted in the statics presented in Table 1.

**Table 1**

Distribution of words and tokens between training and test datasets after preprocessing of all tweets

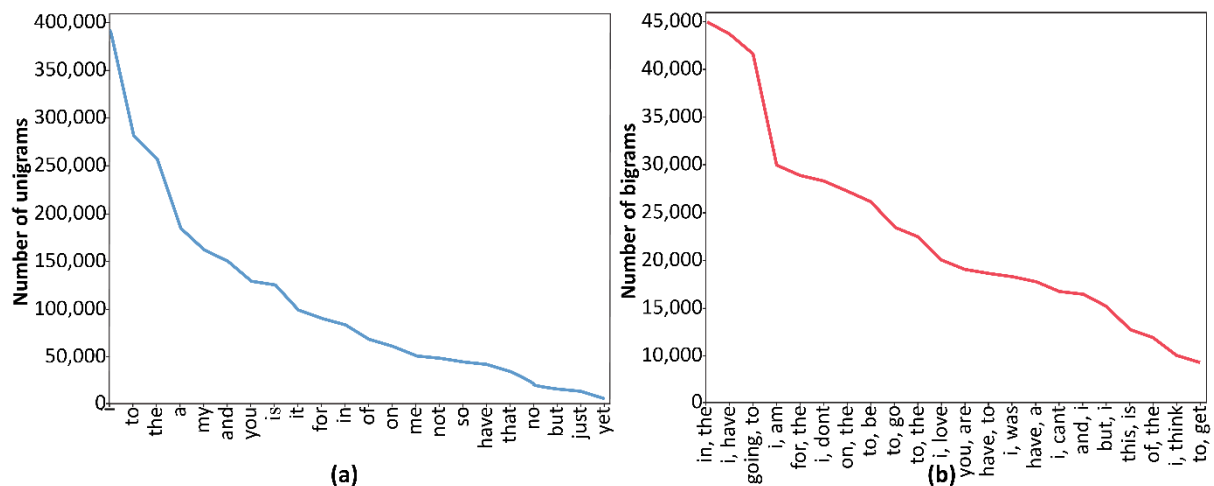
Type of text	Dataset	Unique	Average	Max	Positive	Negative	Overall
Tweets	Train	–	–	–	50,650	49,350	100,000
	Test	–	–	–	–	–	10,000
Unigrams	Train	50,000	9.68	35	–	–	1,224,630
	Test	15,000	9.43	29	–	–	325,671
Bigrams	Train	473,211	8.11	–	–	–	1,000,113
	Test	156,791	8.04	–	–	–	235,002

After preprocessing, the prepared training and test datasets comprised 100,000 and 10,000 text messages, respectively.

### 3.2. Feature extraction

Two types of features, namely unigrams and bigrams, were extracted from the prepared dataset.

**Unigrams** are the simplest and the most used features for text classification [26]. They can be seen as the appearance of single words or tokens within the text. Several single words from the training dataset were extracted, and then a regularity distribution of these words was created. On the whole, 50,000 unique words were extracted from the dataset. Top  $N$  words from the vocabulary were used to create the necessary vocabulary of 15,000 for sparse vector classification and 90,000 for dense vector classification. The regularity distribution of the top twenty words in the vocabulary is shown in Fig. 1a).



**Figure 1:** The distributions of appearances of the top twenty-two (a) unigrams and (b) bigrams

**Bigrams** are pairs of words in a dataset that occur sequentially in a corpus [26]. These features are intended to reflect the objection in a natural language, like in: “It is not bad.” On the whole, 473,211 unique bigrams were extracted from the dataset. Out of these, the bigrams at the end of the regularity spectrum are noisy and occur very few times to influence classification. We, therefore, used only the

top ten thousand bigrams from these to create the vocabulary. Fig. 1b) depicts the regularity distribution of the top twenty bigrams in the vocabulary.

Hence, the top twenty-two unigrams and bigrams were selected based on their distribution for the sentiment analysis. The extraction of features into unigrams and bigrams resulted in two feature vectors: sparse and dense vector representation. The choice of the vector representation depended on the type of ML and DL approaches.

### 3.3. Feature representation

The sparse vector representation of each tweet contained 15,000 elements for only unigrams or 25,000 for both unigrams and bigrams. Each unigram and bigram were assigned unique indices depending on their rank. The positive value of unigrams (and bigrams) indices depended on the feature type preassigned by the authors of this work, either appearance or regularity. Feature representation is defined as follows

- **Appearance:** if a feature appears in a tweet, the feature vector receives the value of “1” at indices of both unigrams and bigrams and the value of “0” – in other cases.
- **Regularity:** if a tweet contains a positive value in a unigram (bigram), then it represents the regularity of that unigram (bigram), and the feature vector receives the value of “1” at an index of that unigram (bigram), and the value of “0” – in other cases.

A matrix of such term-regularity vectors is constructed for the entire training dataset, and then each term regularity is scaled by the inverse-document-regularity of the term (IDF) to assign higher values to essential terms. The tweet-regularity of term  $t$  is determined as follows

$$\text{IDF}(t) = \log \left( \frac{1 + n_d}{1 + \frac{df(d,t)}{dt}} \right) + 1, \quad (1)$$

where  $n_d$  stands for the number of tweets,  $\frac{df(d,t)}{dt}$  represents the number of tweets where term  $t$  occurs.

A vocabulary of 90,000 unigrams, i.e., the top ninety thousand words in the dataset, was selected for the dense vector representation. Moreover, an integer index was appointed to each word according to its rank (beginning with 1).

### 3.4. Classification models

This section discusses the theoretical aspect of several ML and DL approaches [27] that were used for the classification task of sentiment polarity on Twitter.

**Random Forest** is a vivid example of ensemble ML techniques for classification and regression problems. A raw RF aggregates numerous decision trees, serving as a separate classifier. If there are a set of tweets  $x_1, x_2, \dots, x_n$  and their respective sentiment labels  $y_1, y_2, \dots, y_n$  then RF iteratively targets a random sample  $(X_m, Y_m)$ ,  $t = \overline{1, M}$ , where  $M$  is the number of trees in an RF model. The training of an RF model takes part through random sampling of various pairs  $(X_m, Y_m)$ .

**XGBoost** is an advanced ensemble of decision trees that serves as a separate classifier for binary and multiclassification tasks. In this study, the ensemble of  $M$  decision trees was used as follows

$$\begin{aligned} \hat{y}_i &= \sum_{m=1}^M \varphi_m(x_i), \quad \varphi_m \in \Phi; \\ L(\Phi) &= \sum_{i=1}^n l(\hat{y}_i, y_i) + \sum_{m=1}^M \Omega(\varphi_m); \\ \Omega(\varphi) &= \gamma T + \frac{1}{2} \lambda \|\mathbf{w}\|^2. \end{aligned} \quad (2)$$

where  $x_i$  stands for the input object,  $\hat{y}_i$  presents the final prediction,  $\varphi_m$  is the  $m$ -th decision tree,  $\Phi$  is the whole set of trees,  $L(\Phi)$  is the loss function of the whole forest, and  $\Omega$  represents the regularization function.

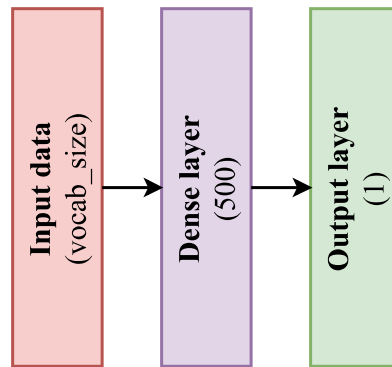
**Support Vector Machine** is a traditional and well-studied ML technique for binary classification tasks. For feature vector  $\mathbf{X} = \{x_i\}_{i=1}^n$  and label vector  $\mathbf{Y} = \{y_i\}_{i=1}^n$  there are a set of points  $(x_i, y_i)$ , for which the maximum-margin hyperplane exists and separates  $(x_i, y_i)$  with outputs  $y_i = \pm 1$ . This hyperplane is determined as follows

$$w_i \cdot x_i - b_i = 0, \quad i = \overline{1, n}. \quad (3)$$

To resolve equation (4) means to find maximum margin  $\theta$  as:

$$\begin{aligned} & \max_{w, \theta} \{\theta\}; \\ & \theta \leq y_i(w_i \cdot x_i + b_i), \quad \forall i = \overline{1, n}. \end{aligned} \quad (4)$$

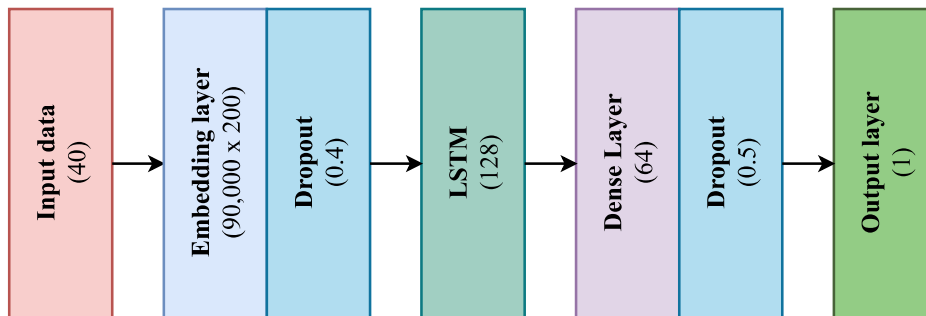
**Multilayer Perceptron (MLP)** is a type of supervised ML techniques with at least three layers of units. Every unit is imitated with a non-linear activation function (usually Sigmoid). Fig. 2 depicts the scheme of the MLP model used in this work.



**Figure 2:** The layers of the MLP model used during the semantic analysis.

A Sigmoid non-linearity function follows every unit within the scheme from Fig. 2.

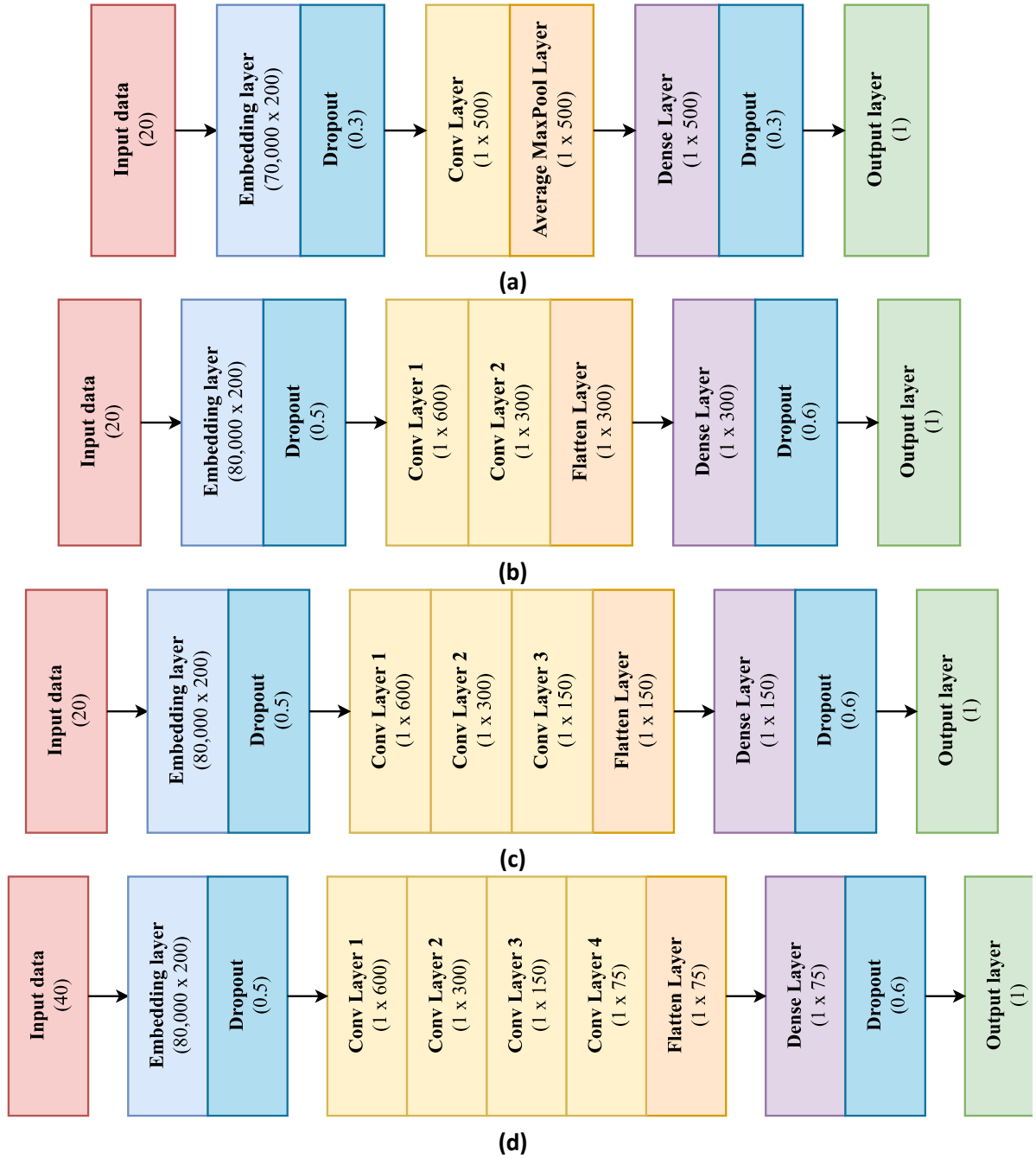
A **Recurrent Neural Network** may be considered a DL method with all neural units connected to each other. The RNN architecture consists of neurons presented in hidden layers, storing information about the consistent dependence on the previous layers. In this study, a particular type of RNN called LSTM was utilized. Fig. 3 illustrates the architecture of RNN used in this work.



**Figure 3:** The scheme of RNN architecture used in this work.

The maximum size of the input layer was set to 40, while the vocabulary size was set to 90,000 words. The two-hundred-dimensional feature vector was used in the RNN model to extract the features of appearance and regularity. The architecture comprised embeddings, LSTM, and dense (fully-connected) layers followed by ReLU activations for non-linearization and dropout for regularizing the training. The final layer with the sigmoid function outputted a single prediction.

**Convolutional Neural Network (CNN)** is a DL approach that comprises convolutional operations for processing spatial information. The temporal convolutions were applied to the CNN architecture to process sequential data (i.e., tweets). In this work, four CNN architectures with different numbers of convolutional operations were explored (Fig. 4).



**Figure 4:** The schemes of CNN architectures with a different number of convolutional layers: (a) one, (b) two, (c) three, and (d) four.

**One-Conv-NN.** The architecture from Fig. 4a) began with the embedding layer and a dropout regularizer to prevent the model from overfitting. Here, one temporal convolutional operation was embedded with a kernel of  $3 \times 3$  and a padding of  $1 \times 1$ . The convolutional layer was followed by a rectified linear unit (ReLU). After the convolution, the average max pooling (AMP) layer was inserted to reduce the data's dimensionality. A dense layer with a dropout regularizer was also applied to the scheme before the output. The final layer contained a sigmoid activation function to convert the feature vector from the fully connected neural scheme into one probability value. In this architecture, the maximum size of the input layer was set to 20 with a vocabulary of 70,000 words.

**Two-Conv-NN.** In this case, the vocabulary size was raised to 80,000 words. Moreover, the second convolution with ReLU was added, and the AMP layer was replaced with the flattened layer to reduce further the dimensionality of the feature vector processed within the network. Also, the values of

hyperparameters were changed considering the number of functions in the network. All changes are depicted in Fig. 4b).

**Three-Conv-NN.** In the architecture from Fig 4c), the general scheme remained similar to the previous one, except for the third convolutional layer and the values of hyperparameters.

**Four-Conv-NN.** The fourth architecture comprised an additional convolutional layer with 75 filters of the size of  $3 \times 3$  (Fig. 4d). Here, the maximum size of the input layer was increased to 40 due to the length of the most significant tweet in the training dataset.

The considered approaches mentioned above were evaluated by the statistical measure defined as

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (5)$$

where TP stands for true positive cases; TN is true negative cases; FP presents false positive cases; FN represents false negative cases.

The computational experiments were performed using Python v3.9 and the ML library called Scikit-learn. The hardware used in the investigation consists of an eight-core Ryzen 2700 and a single NVIDIA GeForce GTX1080 GPU with 8 GB video memory.

## 4. Results and discussion

The chosen classifiers (see subsection 3.4) were implemented to conduct computational experiments. The initial dataset of 100,000 tweets was split into training and validation subsets of 70% and 30%, respectively, i.e., 70,000 tweets were used for training and 30,000 – for validating the models. In addition, the sparse vector representation of tweets was applied to RF, XGBoost, SVM, and MLP classifiers, while the dense vector representation was applied to the RNN and CNN models. The comparison of achieved classification accuracies by ML techniques on the validation subset is shown in Table 2.

**Table 2**

Comparison of traditional classification models based on the sparse vector representation

Algorithms	Appearance, %		Regularity, %	
	Unigrams	Bigrams	Unigrams	Bigrams
RF	77.84	78.21	77.25	<b>78.91</b>
XGBoost	78.68	<b>79.90</b>	78.59	79.44
SVM	79.85	82.02	81.50	<b>82.16</b>
MLP	81.11	82.26	81.16	<b>82.47</b>

**Random Forest.** Twenty runs with features of appearance and regularity were performed during the computations. According to the experiments presented in Table 2, the targeted estimators performed slightly better (78.91%) based on the feature of regularity for bigrams.

**XGBoost.** The maximum tree depth was set to twenty-five for the classification task to handle possible overfitting. At the same time, the number of estimators (trees) was set to three hundred to balance an ensemble of weaker trees. Overall, the combination of unigrams and bigrams provided the highest accuracy of 79.90% (see Table 2).

**SVM.** The value of hyperparameter  $C$  was pointed to 0.01. The experiments were conducted based on the combination of unigrams and bigrams with the features of appearance and regularity. The highest value of 82.16% was achieved from the feature of regularity and combination of bigrams.

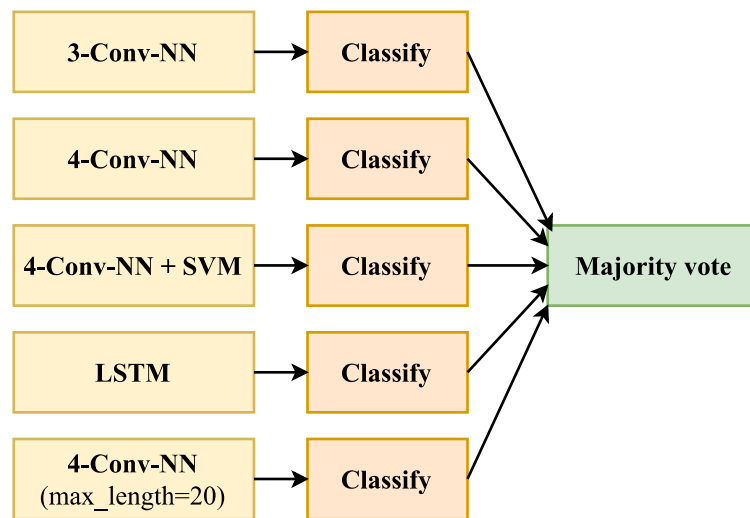
**MLP.** The used MLP model contained one hidden layer of five hundred hidden units. The sigmoid function was served for non-linearization as an output layer. A typical sigmoid function outputs the calculations as the tweet's attitude positivity probability. The probability values were round to 0 and 1 for the binary prediction of positive and negative predictions. The MLP model was trained based on the Adam optimization algorithm with the binary cross-entropy loss. Overall, the MLP model obtained the highest accuracy of 82.47%, with the features of regularity and bigrams.



**RNN.** The RNN model in this study comprised a single LSTM layer of 128 units. The top 50,000 words from the training subset were used to train the RNN model and extract the sparse feature vector. The training was conducted using the Adam optimizer with a momentum of 0.8. We also applied cross-validation for hyperparameter tuning, after which the highest accuracy stopped at 84.03%.

**CNN.** Here, the CNN model was trained using the Adam optimizer to create the dense feature vector with the whole training subset of 70,000 words. Four CNN architectures were employed in the study (see Fig. 4). It was investigated from the computational results that the CNN model with more convolutional layers performed slightly better. Models with one, two, three, and convolutional layers obtained accuracies of 83.51%, 84.18%, 84.11%, and 85.26%, respectively.

**DL ensemble model.** A straightforward ensemble model based on previous approaches was constructed to improve the obtained classification results. We extracted a six-hundred-dimensional dense feature vector from the penultimate layer of four-layer-CNN for each tweet. A SVM classifier with  $C = 0.1$  was chosen to categorize the sentiments of tweets. As such, an ensemble of five different models was prepared, and results were used in the majority vote of predictions. Fig. 5 illustrates the proposed ensemble model.



**Figure 5:** The scheme of the proposed ensemble model based on five different classifiers and majority voting.

The five-fold cross-validation test was also conducted for the combination of CNN and SVM. Table 3 presents the accuracies of five separated models and the proposed majority voting ensemble.

**Table 3**

The classification results of the deep learning models on the test dataset

Architecture	Accuracy, %
LSTM	84.18
3-CNN	84.11
4-CNN	85.26
4-CNN features + SVM	85.30
4-CNN (max_size = 20)	85.52
The proposed ensemble model	<b>85.71</b>

As seen in Table 3, the best results were obtained by the fine-tuned four-layer-CNN model with the SVM classifier (85.52%) and the proposed ensemble model with the majority voting (85.71%).

Overall, according to the computational results (Table 2-3), DL approaches, namely RNNs and CNNs, achieved better classification performance than other traditional ML techniques. The best RNN model achieved an accuracy of 84.03% on the test dataset, and the best CNN model reached 85.52%. At the same time, it was discovered that the CNN model with the SVM classifier demonstrated better

performance than a single CNN. It is also worth noting that the ensemble method based on receiving the most votes according to the five best models' predictions reached an absolute accuracy of 85.71%, surpassing the single DL models by more than 0.19%, and demonstrated its practical usefulness.

## 5. Conclusion

This study aimed to address the issue of classifying emotional expressions based on small texts (tweets) extracted from Twitter. As such, several machine learning and deep learning techniques, namely random forest, XGBoost, SVM, MLP, RNN, and CNN, were considered and implemented to categorize the polarity of tweets into positive and negative classes based on their semantics. Unigrams and bigrams were employed as features to construct the feature vector of semantics. It was investigated that bigrams contributed to improving the classification accuracy and “appearance” in the sparse vector representation recorded a better performance than “regularity.” The considered models of ML and DL were enhanced to handle different emotional expressions of semantics. Moreover, the semantic analysis showed that tweets do not always have strictly positive or negative emotional expressions; sometimes, they may not have semantics, i.e., be neutral.

It was proved according to the computational results that the considered techniques could efficiently facilitate sentiment analysis of tweets by assessing and predicting possible business outcomes on Twitter in real-time. Moreover, the proposed ensemble deep learning model managed to slightly improve (0.19% and more) the categorization of the polarity of the targeted tweets.

Further research will be aimed at expanding the number of categories of emotional expressions of semantics, for example, to classify moods from -3 to +3. In addition, a more detailed linguistic semantic study of tweets on various real-world issues will be conducted.

## 6. References

- [1] D. Zimbra, A. Abbasi, D. Zeng, H. Chen, The state-of-the-art in Twitter sentiment analysis: A review and benchmark evaluation, *ACM Trans. Manag. Inf. Syst.* 9(2) (2018) 1–29. doi:10.1145/3185045.
- [2] C. Messaoudi, Z. Guessoum, L. Ben Romdhane, Opinion mining in online social media: A survey, *Soc. Netw. Anal. Min.* 12(1) (2022) e25. doi:10.1007/s13278-021-00855-8.
- [3] P. Munjal, M. Narula, S. Kumar, H. Banati, Twitter sentiments based suggestive framework to predict trends, *J. Stat. Manag. Syst.* 21(4) (2018) 685–693. doi:10.1080/09720510.2018.1475079.
- [4] M. Bhagat, B. Bakariya, Sentiment analysis through machine learning: A review, in *Proceedings of 2nd International Conference on Artificial Intelligence: Advances and Applications 2021* (2022) 633–647. doi:10.1007/978-981-16-6332-1\_52.
- [5] R. Nagamanjula, A. Pethalakshmi, A novel framework based on bi-objective optimization and LAN2FIS for Twitter sentiment analysis, *Soc. Netw. Anal. Min.* 10(1) (2020) e34. doi:10.1007/s13278-020-00648-5.
- [6] A. Reyes-Menendez, J. R. Saura, C. Alvarez-Alonso, Understanding #WorldEnvironmentDay user opinions in Twitter: A topic-based sentiment analysis approach, *Int. J. Environ. Res. Public Health* 15(11) (2018) e2537. doi:10.3390/ijerph15112537.
- [7] P. Radiuk, Applying 3D U-Net architecture to the task of multi-organ segmentation in computed tomography, *Appl. Comput. Syst.* 25(1) (2020) 43–50. doi:10.2478/acss-2020-0005.
- [8] H. K. Sharma, T. Choudhury, H. F. Mahdi, Social and web analytics: An analytical case study on Twitter data, in *Decision Intelligence Analytics and the Implementation of Strategic Business Management*, P. M. Jeyanthi, T. Choudhury, D. Hack-Polay, T. P. Singh, S. Abujar, Eds. Cham: Springer International Publishing (2022) 135–143. doi:10.1007/978-3-030-82763-2\_12.
- [9] S. Vashishtha, S. Susan, Highlighting key phrases using senti-scoring and fuzzy entropy for unsupervised sentiment analysis, *Expert Syst. Appl.* 169 (2021) e114323. doi:10.1016/j.eswa.2020.114323.
- [10] S. Taneja, S. Bhasin, S. Kapoor, Trends and sentiment analysis of movies dataset using supervised learning, in *Proceedings of International Conference on Intelligent Cyber-Physical Systems* (2022) 331–342. doi:10.1007/978-981-16-7136-4\_25.

- [11] A. Bandhakavi, N. Wiratunga, S. Massie, D. P., Emotion-aware polarity lexicons for Twitter sentiment analysis, *Expert Syst.* 38(7) (2021) e12332. doi:10.1111/exsy.12332.
- [12] K. N. Alam et al., Deep learning-based sentiment analysis of COVID-19 vaccination responses from Twitter data, *Comput. Math. Methods Med.* 2021 (2021) e4321131. doi:10.1155/2021/4321131.
- [13] I. Krak, O. Barmak, P. Radiuk, Information technology for early diagnosis of pneumonia on individual radiographs, in 3rd International Conference on Informatics & Data-Driven Medicine (IDDM-2020) 2753 (2020) 11–21. [Online]. Available: <http://ceur-ws.org/Vol-2753/paper3.pdf>
- [14] H. Jang, E. Rempel, D. Roth, G. Carenini, N. Z. Janjua, Tracking COVID-19 discourse on Twitter in North America: Infodemiology study using topic modeling and aspect-based sentiment analysis, *J Med. Internet Res.* 23(2) (2021) e25431. doi:10.2196/25431.
- [15] G. Yenduri, B. R. Rajakumar, K. Pragmaash, D. Binu, Heuristic-assisted BERT for Twitter sentiment analysis, *Int. J. Comput. Intell. Appl.* 20(03) (2021) e2150015. doi:10.1142/S1469026821500152.
- [16] A. Addawood et al., Tracking and understanding public reaction during COVID-19: Saudi Arabia as a use case, *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020* 24(1) (2020) 1–9. doi:10.18653/v1/2020.nlpCOVID19-2.24.
- [17] N. S. Sattar, S. Arifuzzaman, COVID-19 vaccination awareness and aftermath: Public sentiment analysis on twitter data and vaccinated population prediction in the USA, *Appl. Sci.* 11(13) (2021) e6128. doi:10.3390/app11136128.
- [18] Z. Jiang, F. Di Troia, M. Stamp, Sentiment analysis for troll detection on Weibo, in *Malware Analysis Using Artificial Intelligence and Deep Learning*, M. Stamp, M. Alazab, and A. Shalaginov, Eds. Cham: Springer International Publishing (2021) 555–579. doi:10.1007/978-3-030-62582-5\_22.
- [19] F. Wunderlich, D. Memmert, Innovative approaches in sports science—Lexicon-based sentiment analysis as a tool to analyze sports-related twitter communication, *Appl. Sci.* 10(2) (2020). doi:10.3390/app10020431.
- [20] T. A. Tran, J. Duangsuwan, W. Wettayaprasit, A new approach for extracting and scoring aspect using SentiWordNet, *Indones. J. Electr. Eng. Comput. Sci.* 22(3) (2021) 1731–1738. doi:10.11591/ijeecs.v22.i3.pp1731-1738.
- [21] P. Sharma, A. K. Sharma, Experimental investigation of automated system for Twitter sentiment analysis to predict the public emotions using machine learning algorithms, *Mater. Today Proc.* (2020). doi:10.1016/j.matpr.2020.09.351.
- [22] M. Boukabous, M. Azizi, Crime prediction using a hybrid sentiment analysis approach based on the bidirectional encoder representations from transformers, *Indones. J. Electr. Eng. Comput. Sci.* 25(2) (2022) 1131–1139. doi:10.11591/ijeecs.v25.i2.pp1131-1139.
- [23] T. D. Dikiyanti, A. M. Rukmi, M. I. Irawan, Sentiment analysis and topic modeling of BPJS Kesehatan based on Twitter crawling data using Indonesian Sentiment Lexicon and Latent Dirichlet Allocation algorithm, *J. Phys. Conf. Ser.* 1821(1) (2021) e12054. doi:10.1088/1742-6596/1821/1/012054.
- [24] I. Krak, O. Barmak, P. Radiuk, Detection of early pneumonia on individual CT scans with dilated convolutions, in 2nd International Workshop on Intelligent Information Technologies & Systems of Information Security (IntelITSIS-2021) 2853 (2021) 214–227. Accessed: May 09, 2021. [Online]. Available: <http://ceur-ws.org/Vol-2853/paper20.pdf>
- [25] S. R. S. Gowda, B. R. Archana, P. Shettigar, K. K. Satyarthi, Sentiment analysis of Twitter data using Naive Bayes classifier, in *ICDSMLA 2020. Lecture Notes in Electrical Engineering* 783 (2022) 1227–1234. doi:10.1007/978-981-16-3690-5\_117.
- [26] M. Garg, UBIS: Unigram bigram importance score for feature selection from short text, *Expert Syst. Appl.* 195 (2022), e116563. doi:10.1016/j.eswa.2022.116563.
- [27] J. F. Raisa, M. Ulfat, A. Al Mueed, S. M. S. Reza, A review on twitter sentiment analysis approaches, in 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD) (2021) pp. 375–379. doi:10.1109/ICICT4SD50815.2021.9396915.