



Sentiment Analysis on Social Media Reviews Datasets with Deep Learning Approach

 Muhammet Sinan Başarslan¹,  Fatih Kayaalp²

¹Corresponding Author; Dogus University, Advanced Vocational School, Computer Programming;
mbasarslan@dogus.edu.tr;

²Duzce University; Engineering Faculty, Department of Computer Engineering; fatihkayaalp@duzce.edu.tr

Received 28 November 2020; Revision 20 December 2020; Accepted 4 February 2021; Published online 15 February 2021

Abstract

Thanks to social media, people are now able to leave guiding comments quickly about their favorite restaurants, movies, etc. This has paved the way for the field of sentiment analysis, which brings together various disciplines. In this study, Yelp restaurant reviews and IMDB movie reviews dataset were used together with the data collected from Twitter. Word2Vec (W2V), Global Vector (GloVe) and Bidirectional Encoder Representation (BERT) word embedding methods, Term Frequency-Reverse Document Frequency (TF-IDF), and the Bag-of-Words (BOW) were used on these datasets. Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), Support Vector Machine (SVM), and Naive Bayes (NB) were used in the sentiment analysis models. Accuracy, F-measure (F), Sensitivity (Sens), Precision (Pre), and Receiver Operating Characteristics (ROC) were used in the evaluation of the model performance. The Accuracy rates of the models created by the Machine Learning (ML) and Deep Learning (DL) methods using the IMDB dataset were in the range of 81%-90% and 84%-94%, respectively. These rates were in the range of 80%-86% and 81%-89% for the Yelp dataset, and in the range of 75%-79% and 85%-98% for the Twitter dataset. The models that incorporated the BERT word embedding method have the best performance, compared to the other models with ML and DL. Therefore, BERT method is recommended for this type of analysis in future studies.

Keywords: sentiment analysis, deep learning, machine learning, text representation, word embedding.

1. Introduction

In parallel with the advances in technology, visual and print communication channels have shifted towards social media. Social events such as movies, restaurants, concerts are now publicized through articles published on social media or websites, instead of recommendations on newspapers and magazines, thanks to the Internet technologies.

The fact that social media is an indispensable tool for people and that they constantly express their opinions about social issues, economy, health, products, and brands paves the way for sentiment analysis. Sentiment analysis is carried out using natural language processing, an important part of artificial intelligence. In the sentiment analysis studies, underlying sentiments in textual expressions are identified. This analysis is used to see whether the sentiment of the texts shared by people is positive, negative, or neutral. Sentiment analysis are used by companies to see whether they receive a positive feedback [1].

The purpose of the text classification is to assign single or multiple tags to a text string. Conventional approaches for text classification, and the classification in the feature extraction step of BOW, usually utilizes the TF-IDF probabilities. With the advances in natural language processing, BERT, Word2Vec, and GloVe have started to be widely used in feature extraction. However, these methods often ignore the contextual information or word order in texts and they have data flexibility issues, which affect classification accuracy. NB, support vector machines, decision trees, networks such as CNN and LSTM based RNN are used in recent ML algorithms.

In this study, architectures that increase the classification performance in ML and DL models was investigated by applying the traditional text representation method and word embedding methods, which

are widely used in sentiment analysis studies. The model with the best result was proposed as the recommended framework.

In the study, five different datasets were obtained using traditional text representation methods of TF-IDF, BOW, and the word embedding methods BERT, Word2Vec, and GloVe were used on three different datasets. After obtaining these datasets, sentiment analysis, which is one of the natural language processing tasks, was carried out by using ML algorithms of support vector machine and Naive Bayes classifier algorithms, and by using the DL methods of CNN, RNN, and LSTM. Accuracy, F, Sens, Pre, and ROC performance criteria were used in the evaluation of the models created by ML and DL.

As a contribution to the literature, hybrid classifier models of DL and ML were created by using word representation methods for meaning, context, and syntax on public data sources and datasets collected by the researchers.

As shown in the related studies section, classifier models created by ML such as SVM, ANN, and NB, CNN, RNN, LSTM DL are popular and have good performances in sentiment analysis studies. As another contribution, this study evaluates the performance of these algorithms by comparing them with traditional frequency-based text representation (TF-IDF, BOW) and prediction based text representation (W2V, GloVe, BERT) methods.

In the second section, sentiment analysis studies with ML and DL are discussed. In the third section, under the methodology subtitle, datasets used in the study, word representation and embedding methods, ML, and DL algorithms are discussed. The fourth section explains the proposed framework in the study. In the fifth section, the experiments made with the created models and their results are presented. Finally, the sixth section draws the conclusions. The flowchart of the study is shown in Figure 1.

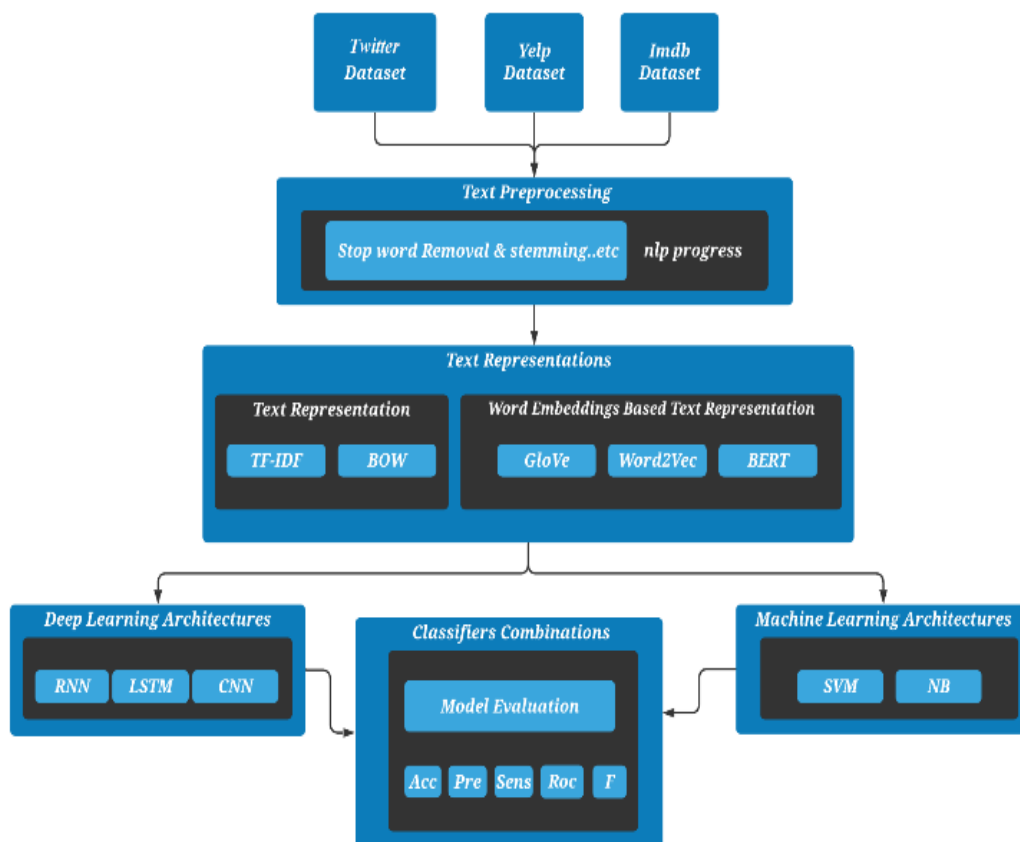


Figure 1 The flowchart of study

2. Related Works

Sentiment analysis studies with various datasets in different languages are introduced in this section. In their sentiment analysis study, Pang et al. have created a pre-classification vector space model on the movie comments present in the Internet Movie Database archive, and conducted a sentiment analysis via classifying algorithms, such as NB, Maximum Entropy (ME), and SVM. Of the classification algorithms, they achieved the best performance with SVM, by 82.9% accuracy, using unigrams on the dataset [2].

In their study on movie reviews, Kaynar et al. used NB, Multilayered Artificial Neural Network (ANN), and SVM. They also used TF-IDF for feature extraction. SVM has yielded better results in terms of accuracy, compared to other methods [3].

Hamoud et al. have used the BOW, TF, and TF-IDF for the classification of political tweets on the Twitter data. They used SVM and NB classification algorithms. According to the results, BOW-enabled SVM provides the highest accuracy and F-measure [4].

Symeonidis et al. used Linear SVC, Bernoulli NB, Logistic Regression (LR), and CNN, which are four popular ML algorithms. They achieved the best results by CNN in terms of accuracy [5].

A deep-learning-based approach using convolutional neural network (CNN) and word2vec on Twitter dataset to detect opportunities for improving the quality of their products or services through sentiment analysis has also been proposed in [6]. The study has obtained encouraging results with 88.7% precision, 88.7% recall, and 88.7% F-measure.

Zheng et al. have proposed a model based on the hybrid bidirectional RNN in their study conducted with various datasets such as Sogou, Yelp and Douban Movies. The accuracy rates of the method they proposed varies between 73.46% and 96.81% [7].

Huq et al. have used feature extraction with n-grams on Twitter data and then applied SVM and K-Nearest Neighbor algorithms on the dataset. According to their experiments, accuracy values were between 58.39% and 79.99% [9].

Amolik et al. have classified tweets correctly by using Feature-Vector, NB, and SVM classifier algorithms. Despite its lower recall and accuracy, NB had better sensitivity compared to SVM [10].

Liao et al. have created a simple CNN model with W2V on the data collected from Twitter, and have used this model for comparison against SVM and NB. As a result, CNN has shown to have higher classification performance in terms of accuracy compared to other models [11].

Li et al. have achieved a classification accuracy in the range of 52.23-55.93% in their experiments with DL architectures, such as CNN, LSTM, MemNET, AttNet, applied to three different datasets of Online debates, Restaurants, and laptop reviews [12].

Li et al. have proposed an improved version of the Sliced RNN and have compared this model against various DL models in a sentiment study. According to the results, their proposed model had the highest accuracy by 73.36% [13].

Zhao et al. obtained the highest accuracy rate of 87.9% in the models they created with CNN and LSTM DL algorithms on the Amazon product reviews dataset [14].

Al-Smadi et al. have shown better results in the models they created with the DL RNN and ML SVM algorithms, on the data of Arabic hotel reviews. They obtained an accuracy rate of 87% with RNN and 95.4% with SVM [15].

In their study, Tang, Qin et al. have achieved an accuracy of 80.95% on the restaurant views dataset with the DL algorithm, while they achieved an accuracy of 72.37% in laptop views [16].

In the study of Chen et al. on Chinese Twitter data with RNN-based models, an accuracy of 73.89% was obtained [17].

Altrabsheh et al. used NB, SVM, ME, and Random Forest (RF) algorithms in sentiment analysis with unigram, bigram, trigram-based text representations on the tweets about courses such as mathematics, database, engineering, molecular biology, chemistry, and physics. Models created with SVM and text representations had better performance compared to the other models [18].

H. Ghulam et al. have created models with LSTM, RF, NB in a sentiment analysis study on Roman Urdu tweets. Models created with LSTM had better performance compared to the other models [19].

J. Singh et al. have combined sentiment analysis and morphological assessment in Punjabi language, using DL. The accuracy rate of the model, created using DL and morphological text classification with 275 suicide cases in Punjab, was 95.45% [20].

As seen above, mostly traditional word representation methods were used in previous studies. In this study, the performances of traditional machine learning and deep learning classification algorithms were investigated also by using different text representation and word embedding techniques.

As seen above, DL algorithms such as RNN, LSTM, CNN, and ML algorithms such as NB and SVM are so popular in sentiment analysis studies. In addition, different word embedding methods such as BERT, W2V, GloVe, TF-IDF and BOW have also been used in various studies.

3. Methodology

In this section, the datasets, word embedding techniques, ML and DL algorithms, and details of the proposed system are discussed.

3.1 Datasets

Three different datasets were used in the study. These datasets include the IMDB movie review dataset, which is often used in sentiment analysis studies, Yelp hotel and restaurant comments, and Twitter API.

Yelp (restaurant reviews) dataset consists of 598,000 reviews of various restaurants. 560,000 of the reviews were reserved for training and 38,000 for testing [21]. Dataset attributes and descriptions of these features are presented in Table 1.

Table 1 Yelp Dataset

Attribute	Description
Text	Review from yelp
Sentiment class	Positive, negative

IMDB (movie reviews) dataset consists of 50,000 positive and negative movie reviews [22]. In this dataset, 50,000 reviews were split into 25,000 testing and 25,000 training data. Dataset attributes and descriptions of these features are presented in Table 2.

Table 2 IMDB Dataset

Attribute	Description
Text	Review from IMDB
Sentiment class	Positive, negative

4500 health-related Twitter data were collected using the Twitter API. The pre-processing and sentiment analysis of these data were carried out using the Python programming language. The collected tweets were labeled as 1680 neutral, 1220 positive, 1600 negative tweets. The neutral-tagged tweets were the drug ads, and their attribute information is presented in Table 3. Tweets marked as negative seem to belong to those with various diseases. On the other hand, the positive ones are the tweets indicating that diseases such as cancer have successfully treated.

Table 3 Twitter Dataset

Attribute	Description
id	Order of tweet data frame
text	tweet
created_at	Date and time the Tweet was posted
retweeted	Tweet rerun status (bool)
retweet_count	Number of retweets
user_screen_name	Username
user_followers_count	Number of followers
user_location	Followers location
hashtags	Tweet tag
sentiment_score	Sentiment score
sentiment_class	positive, negative, neutral

Since the datasets were scraped from the web, some HTML (Hyper Text Markup Language) codes were also present in the datasets. Therefore, it was necessary to clear these texts by removing HTML tags. The numbers, punctuation, and stop words were removed. Although BERT gives successful results in splitting compound names made with word representation dashes, other methods have problems. A set of NLTK (Natural Language Tool Kit) stop words was used to remove stop words. Since BERT embedding was trained on Wikipedia data, we allowed numbers and some of the punctuations like [, / () : ; '] and compound nouns with a hyphen, which may cause a more reliable embedding to remain in the text. Moreover, we saved [! ? .] to detect the end of the sentence for a later purpose (generate BERT for each sentence). Stemming and lemmatization according to POS (Part of Speech) tags of words were used for BOW and TF-IDF embedding. Finally, we replaced white spaces with only one space.

3.2 Text Representation

The representation of documents in text processing is important for successful results. In the text classification applications, texts are represented as vectors in the dataset. Such vector corresponds to the words in the document. Vector representation of documents. A document-word matrix is created. Thus, the words in the document are of importance. Vectors are calculated using various word weighting methods. TF-IDF is a weighting method widely used in text processing. In this method, the frequency of each word is represented by multiplying the inverse document frequency (IDF). This decreases the importance of highly repetitive words and increases the importance of words with fewer words.

There are also word embedding techniques used without document representation. In this study, however, the following document representation methods, BOW and TF-IDF, were used.

3.2.1 TF-IDF

TF is the method used to calculate term weights in a document. Eq. (1) is seen. The IDF tries to find out the number of words in more than one document and to determine whether the word is a term or not (Stop Words). For this, the absolute value of the logarithm of the number of documents passed by the term must be divided by the number of documents. Eq. (2) is seen [23]. In Eq. (2), t is the term and j is the document. TF-IDF score i in document j is calculated as in Eq. (3).

$$TF(i, j) = \frac{\text{Term } i \text{ frequency in document } j}{\text{Total words in document } j} \quad (1)$$

$$IDF(i) = \log \left(\frac{\text{Total documents}}{\text{documents with term } i} \right) \quad (2)$$

$$j = TF(i, j) * IDF(i) \quad (3)$$

3.2.2 BOW

BOW is the document representation model widely used in text processing. In the BOW model, the word order of text documents is not preserved, but only the word counts are taken into account [24]. The BOW model, which shows the frequency of words in documents, was used by the classifier to create a learning model with a set of features.

3.3 Word Embedding Based Text Representation

Word2Vec, GloVe, BERT word embedding methods are explained in this section.

3.3.1 W2V

W2V method is a word embedding method that learns the vector representations of words using a training set with ANN [25] - [27]. It has two models, the Continuous Bag of Words (CBOW) and Skip-gram, which matches close vectors with similar meaningful words in the vector space. While the CBOW model predicts a word in a certain context, the Skip-gram model predicts the context of a particular word.

W2V extracts vector representations of words from datasets. The skip-gram and the CBOW model are shown in Figure 2.

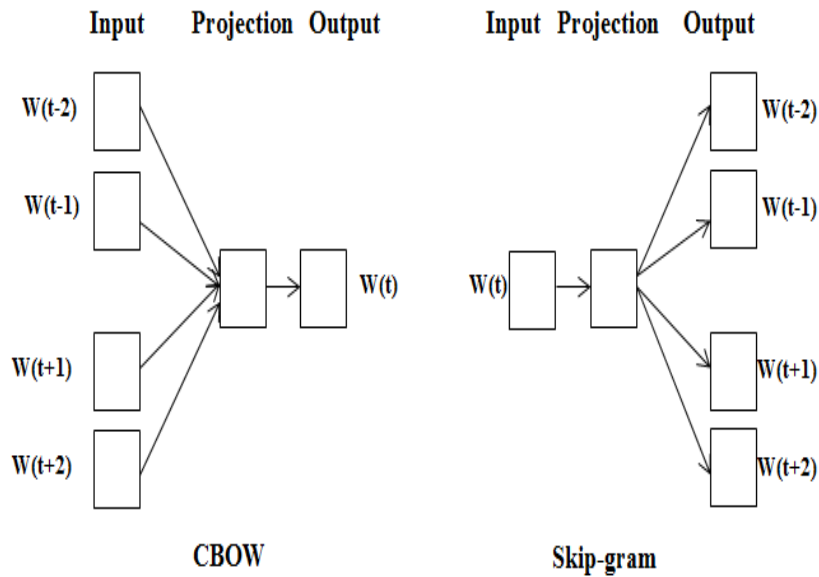


Figure 2 W2V models

3.3.2 GloVe

The GloVe is an advanced method from W2V that makes embedding words in documents more efficient. The GloVe is regression-based and the objective function is given in Equation. (4):

$$J = \sum_{i,j=1}^v f(X_{ij})(w_i^T v_j + b_i + b_j - \log X_{ij})^2 \quad (4)$$

where v denotes the vocabulary size, $w \in R^d$ represents the word vectors, V represents context word vectors, X_{ij} is the number of times the word pair (i, j) occurs together in the corpus. $f(X_{ij})$ denotes a weighting function and b_i, b_j are bias parameters [27].

3.3.3 BERT

BERT is a word embedding model that stands for bi-directional encoder representations. The BERT model is designed to condition the word in right and left contexts by pre-training the dataset in each layer and in both directions. Figure 3 shows the architecture of the BERT model.

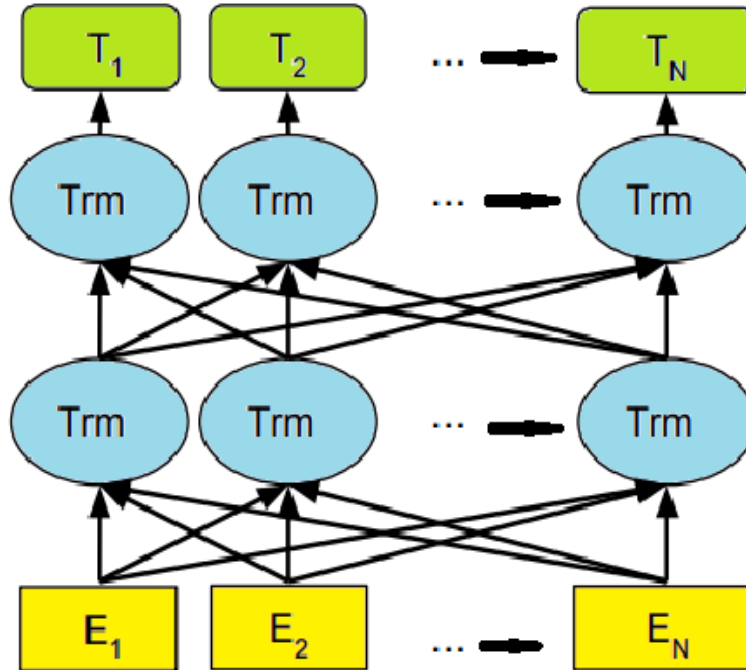


Figure 3 BERT model

3.4 Machine Learning

It has been introduced in the 1980s and has become popular in data mining. These are self-training systems that make better decisions by making simulations with the data and parameters given for learning purposes.

3.4.1 Naïve Bayes Classifier

The NB algorithm, named after Thomas Bayes, is based on Bayes' theorem.

Let $X = \{ x_1, x_2, x_3, \dots, x_n \}$ is the sample set, and $C_1, C_2, C_3, \dots, C_m$ is the class set. The sample to be classified:

$$P(X|C_i) = \frac{P(X|C_i)P(C_i)}{P(X)} \quad (5)$$

As seen in Eq. (5), the probability value is calculated according to the data of the class with the highest probability [28].

3.4.2 Support Vector Machine

SVM is a ML method that sets a boundary between any point in the training data and another furthest point [28]. One feature of SVM is the inherent risk minimization in statistical learning theory [29].

3.5 Deep Learning

Intelligent systems have been developed in various fields with ML algorithms in recent years. Various classifier algorithms are successfully used for tagging in data classification, as one of the ML methods. With the increase in the amount of data, however, the performance of the models decreases. Hence, different algorithms and methods have been developed to overcome hardware problems. One of these methods is the DL algorithms that emerged in line with the neural networks introduced in the 1940s [30].

Although there were some limited achievements before the early 2000s due to the limitations in the computing power, it was not practical to train neural networks as today [31].

DL is a structure consisting of an increasing number of ANN layers that function like neurons in the human brain. Recent years witnessed its widespread use in sentiment analysis. Of the DL algorithms, LSTM, CNN, and RNN algorithms were used in this study.

3.5.1 Recurrent Neural Network

Thanks to recent advances in technology, RNN can be used easily. RNN is a neural network model developed to learn existing patterns by taking advantage of sequential information [28]-[29]. In RNN, each output is determined by the continuous processing of the same task on each instance of the array. The output is determined according to previous calculations [32].

In RNN, the resulting output is based not only on the current input, but also on the other inputs. In addition to the input data at time t , the results of the hidden layer at the time $t-1$ are used as the input of the hidden layer at the time t . The decision regarding the input at the time $t-1$ also affects the decision to be made at the time t . In other words, the inputs of these networks generate output by combining current and previous information. Eq. (6) shows the result of the hidden layer s_t at the time t . Eq. (6), shows the input x_t at the time t , the hidden state S_t , the activation function of the f value, and the weight at U and W [33]:

$$s_t = f(Ux_t + Ws_{t-1}) \quad (6)$$

3.5.2 Long Short-Term Memory

LSTM is an RNN architecture. Unlike standard feed-forward neural networks, LSTM has feedback links. It consists of a cell, and three types of gates: an input gate, an output gate, and a forget gate. Based on the open-closed state of the gates, the cells determine the information to be preserved and the time to access the units [34].

Through these gates, the cell decides what to store, when to read, write or delete. These gates have a network structure and activation function. Just like neurons, they pass or stop the incoming information according to their weights. These weights are calculated during the learning phase of the recurrent network.

3.5.3 Convolutional Neural Networks

Although CNN is one of the deep learning algorithms used in artificial intelligence fields such as Natural Language processing, it is also often used in the field of Image processing. It consists of three main layers [35]:

The first layer is the Convolutional Layer where a filter is used to transform the input matrix. In this layer, each filter maps the input matrix to a gap, and the output size depends on the size of the filter.

The second layer is the pooling layer. It is usually placed after the convolutional layer and used to reduce the size of the mapped elements.

The third layer is the fully connected layer. It is placed after the last pooling layer. The activation functionality in each layer is determined by the network for classification.

3.6 Evaluation Metrics

The confusion matrix used in the model evaluation gives the number of correctly and incorrectly classified samples according to binary classification. (T_P) represents false positive (F_N), true positive (F_P), false negative, and (T_N) true negative numbers (Table 4) [36].

Table 4 Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	T_P	F_N
	Negative	F_P	T_N

Accuracy, Sens, Pre, F used in the study are given between Eq. (7) and Eq. (10).

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (7)$$

$$Sensitivity = \frac{T_P}{T_P + F_N} \quad (8)$$

$$Precision = \frac{T_P}{T_P + F_P} \quad (9)$$

$$F - measure = \frac{2 * Precision * Sensitivity}{Precision + Sensitivity} \quad (10)$$

In order to partition the dataset as training and testing, 10-fold cross-validation method is used in the experiments. The original dataset is randomly partitioned into 10 equal sized partitions. Each time, one of the partitions is used for testing and the others are used for training. The process is repeated ten times and the average results across all steps are calculated.

4. Proposed Framework

The image of the proposed model for sentiment analysis on the publicly available and privately collected datasets is shown in Figure 4. Text processing such as the stop-word elimination was performed in all datasets. On the collected Twitter data, hashtags and URLs were removed.

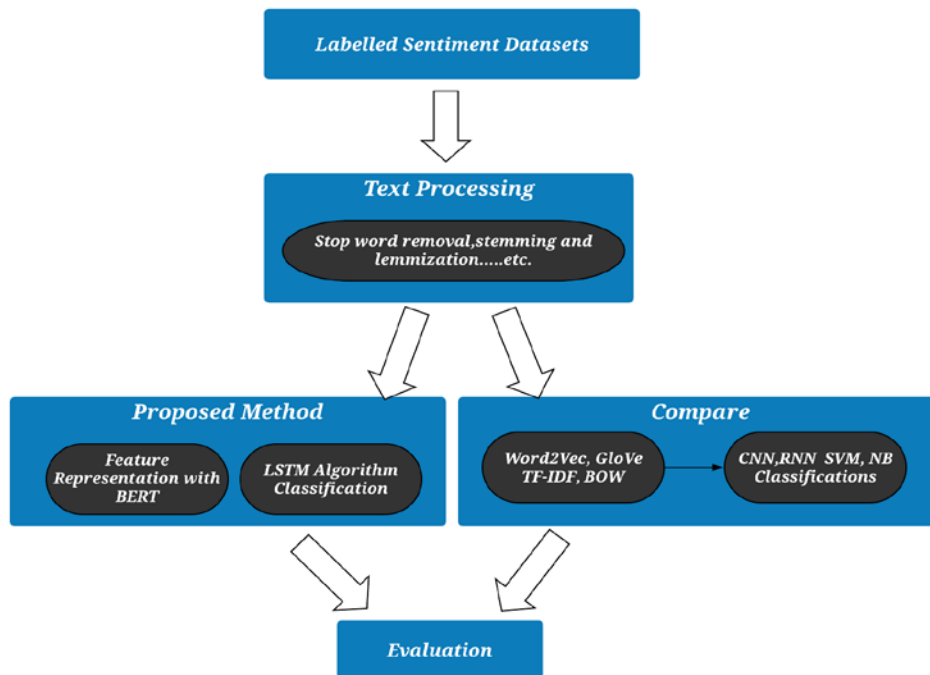


Figure 4 The proposed framework

As shown in Figure 4, the model created by the combination of the BERT word embedding representation method and the LSTM DL algorithm was compared to the models created by other word representation and learning algorithms.

The results of the proposed method are shown as bold and red in all tables. Besides, the text representation and word embedding method that gives the best results in each classification algorithm categories are shown as bold.

As shown in Table 5-7, SVM, one of the ML algorithms, gave a better performance in all performance criteria compared to NB, followed by the word representation and embedding methods. In DL algorithms, word embedding, and representation methods, the LSTM classifier model used after the

Table 5 Performance of Classification Algorithms on IMDB Review Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
SVM	BOW	81%	81%	83%	82%	88%
	TF-IDF	83%	84%	84%	84%	90%
	W2V	84%	84%	86%	85%	92%
	GloVe	89%	88%	90%	88%	91%
	Bert	90%	90%	91%	90%	91%
NB	BOW	81%	82%	81%	81%	89%
	TF-IDF	82%	82%	83%	82%	90%
	W2V	83%	84%	85%	84%	92%
	GloVe	86%	86%	86%	86%	84%
	Bert	87%	86%	87%	88%	90%
CNN	BOW	84%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	87%	84%	85%	84%	92%
	GloVe	88%	86%	86%	86%	84%
	Bert	93%	86%	87%	88%	90%
RNN	BOW	85%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	88%	84%	85%	84%	92%
	GloVe	90%	86%	86%	86%	84%
	Bert	92%	90%	88%	88%	90%
LSTM	BOW	86%	82%	81%	81%	89%
	TF-IDF	86%	82%	83%	82%	90%
	W2V	89%	84%	85%	84%	92%
	GloVe	91%	86%	86%	86%	84%
	Bert	94%	94%	93%	89%	94%

BERT word embedding method was found to perform better than other DL methods. Similarly, performances of the word representation methods with ML and DL algorithms were obtained for the classifier models created with BERT, GloVe, Word2Vec, TF-IDF, BOW, respectively. The results also confirmed that the GloVe is the improved version of W2V.

In addition, the results showed that the models with BERT word embedding method, used both with ML and DL, have better performance than the others. This reveals that the BERT is more successful than other text representation methods.

Table 6 Performance of Classification Algorithms on Yelp Review Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
SVM	BOW	81%	80%	81%	81%	81%
	TF-IDF	81%	82%	81%	81%	82%
	W2V	83%	84%	85%	84%	83%
	GloVe	84%	84%	86%	85%	86%
	Bert	86%	87%	83%	86%	90%
NB	BOW	74%	73%	73%	74%	78%
	TF-IDF	76%	77%	77%	77%	85%
	W2V	78%	78%	78%	81%	86%
	GloVe	79%	79%	78%	79%	88%
	Bert	81%	83%	81%	80%	91%
CNN	BOW	81%	82%	81%	81%	89%
	TF-IDF	82%	82%	83%	82%	90%
	W2V	84%	84%	85%	84%	92%
	GloVe	86%	86%	86%	86%	94%
	Bert	87%	86%	87%	88%	95%
RNN	BOW	82%	82%	81%	82%	86%
	TF-IDF	83%	83%	84%	83%	88%
	W2V	85%	84%	85%	85%	91%
	GloVe	87%	86%	86%	86%	92%
	Bert	88%	86%	87%	88%	94%
LSTM	BOW	83%	77%	75%	76%	82%
	TF-IDF	84%	78%	76%	75%	83%
	W2V	84%	82%	81%	81%	85%
	GloVe	85%	82%	83%	82%	87%
	Bert	89%	84%	85%	84%	91%

Table 7 Performance of Classification Algorithms on Twitter Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
SVM	BOW	80%	78%	77%	80%	89%
	TF-IDF	83%	83%	82%	81%	86%
	W2V	89%	88%	86%	87%	90%
	GloVe	89%	88%	86%	88%	90%
	Bert	89%	87%	89%	87%	93%
NB	BOW	70%	72%	73%	74%	75%
	TF-IDF	72%	73%	73%	76%	78%
	W2V	72%	76%	75%	76%	79%
	GloVe	75%	77%	75%	76%	80%
	Bert	79%	78%	76%	77%	82%
CNN	BOW	84%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	87%	84%	85%	84%	92%
	GloVe	88%	86%	86%	86%	84%
	Bert	93%	86%	87%	86%	90%
RNN	BOW	85%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	88%	84%	85%	84%	92%
	GloVe	90%	86%	86%	86%	84%
	Bert	94%	86%	87%	86%	90%
LSTM	BOW	87%	86%	87%	84%	85%
	TF-IDF	89%	89%	87%	86%	88%
	W2V	91%	94%	91%	94%	95%
	GloVe	96%	96%	96%	96%	96%
	Bert	98%	98%	99%	99%	98%

5. Conclusion And Discussion

This study was conducted on the public and privately collected data to compare the word representation and embedding methods for sentiment analysis tasks with ML and DL algorithms. The Accuracy, Pre, Sens, F, and ROC were used as performance metrics.

In the study, learning algorithms CNN, LSTM, RNN from DL; SVM, NB from ML were used for classifying the sentiments. Word embedding methods BERT, GloVe, Word2Vec, and traditional word representation methods TF-IDF, BOW were also used.

According to the results of the experiments, the model created with Bert and LSTM has shown the best performance among the model combinations created on all datasets. Besides, the models that incorporated the BERT word embedding method have the best performance, among the other text representations and word embedding method.

In future studies, methods such as EIMo that yield successful results in sentiment analysis studies and the performance of the transformers such as RoBERTa and DistilBERT in neural networks such as LSTM and RNN are planned.

References

- [1] E. Park, J. Kang, D. Choi, and J. Han, "Understanding Customers' Hotel Revisiting Behaviour: a sentiment analysis of Online Feedback Reviews," *Current Issues in Tourism*, vol. 23, pp. 605-611, 2020, doi: 10.1080/13683500.2018.1549025.
- [2] B. Pang and L. Lee, "Opinion mining and sentiment analysis", *Foundations Trends Information Retrieval*, vol. 2, no. 2, 2008, pp. 1-135.
- [3] O. Kaynar, H. Arslan, Y. Görmez and F. Demirkoparan, "Makine Öğrenmesi Yöntemleri ile Duygu Analizi," *International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1-5, Malatya, 2017.
- [4] A. Al Hamoud, A. Alwehaibi, K. Roy, and M. Bikdash, "Classifying Political Tweets Using Naïve Bayes and Support Vector Machines," *In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pp. 736-744, 2018, doi: 10.1007/978-3-319-92058-0_71.
- [5] S. Symeonidis, D. Effrosynidis, and A. Arampatzis, "A Comparative Evaluation of Pre - Processing Techniques and Their Interactions for Twitter Sentiment Analysis," *Expert System Applications*, vol. 110, pp. 298-310, 2018, doi: 10.1016/j.eswa.2018.06.022.
- [6] M. A. Paredes-Valverde, R. Colomo-Palacios, M. P. Salas-Zárate, and R. Valencia-García, "Sentiment Analysis in Spanish for Improvement of Products and Services: A Deep Learning Approach," *Scientific Programming*, vol. 2017, 2017, doi: 10.1155/2017/1329281.
- [7] J. Zheng and L. Zheng, "A Hybrid Bidirectional Recurrent Convolutional Neural Network Attention-Based Model for Text Classification," *IEEE Access*, vol. 7, 2019, pp. 106673-106685, doi: 10.1109/ACCESS.2019.2932619.
- [8] S. Liu, "Sentiment Analysis of Yelp Reviews: A Comparison of Techniques and Models", *arXiv preprint*, arXiv:2004.13851, 2020.
- [9] M. R. Huq, A. Ali, and A. Rahman, "Sentiment Analysis on Twitter Data Using KNN and SVM," *International Journal of Advanced Computer Science and Applications*, vol. 8, pp. 19-25, 2017, doi: 10.14569/IJACSA.2017.080603.
- [10] A. Amolik, N. Jivane, M. Bhandari, and M. Venkatesan "Twitter Sentiment Analysis of Movie Reviews Using Machine Learning Techniques," *International Journal of Engineering and Technology*, vol. 7, no. 6, pp. 1-7, 2016.
- [11] S. Liao J. Wang R. Yu, K. Sato, and Z., Cheng, "CNN for Situations Understanding Based on Sentiment Analysis of Twitter Data," *Procedia Computer Science*, vol. 111, 2017, pp. 376–381, 2017, doi: 10.1016/j.procs.2017.06.037
- [12] Li C, Guo X, Mei Q (2017b) Deep Memory Networks for Attitude Identification. *In: Proceedings of the tenth ACM International Conference on Web Search and Data Mining, WSDM*, Cambridge, United Kingdom, pp 671–680, 2017.
- [13] B. Li, Z. Cheng, Z. Xu, W. Ye, T. Lukasiwicz and S. Zhang, "Long Text Analysis Using Sliced Recurrent Neural Networks with Breaking Point Information Enrichment," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, pp. 7550-7554, 2019, doi: 10.1109/ICASSP.2019.8683812.
- [14] W. Zhao et al., "Weakly-Supervised Deep Embedding for Product Review Sentiment Analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, 1 Jan. pp. 185-197, 2018, doi: 10.1109/TKDE.2017.2756658.
- [15] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, "Deep Recurrent

- Neural Network vs. Support Vector Machine for Aspect-Based Sentiment Analysis of Arabic Hotels' Reviews," *Journal of Computational Science*, 2017, doi: 10.1016/j.jocs.2017.11.006.
- [16] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment Embeddings with Applications to Sentiment Analysis," *In IEEE Transactions on Knowledge and Data Engineering*: vol. 28, pp. 496–509, 2016, doi: 10.1109/TKDE.2015.2489653.
- [17] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent Attention Network on Memory for Aspect Sentiment Analysis," *Empirical Methods in Natural Language Processing*, pp. 452–461, 2017.
- [18] F. Tian et al., "Recognizing and Regulating Elearners' Emotions Based on interactive Chinese Texts in E-Learning Systems," *Knowledge Based System*, vol. 55, 148–164, 2014, doi: 10.1016/j.knosys.2013.10.019
- [19] H. Ghulam, F. Zeng, W. Li, and Y. Xiao, "Deep learning-based Sentiment Analysis for Roman Urdu Text," *Procedia Computer Science*, vol. 147, pp.131-135, 2019, doi: 10.1016/j.procs.2019.01.202
- [20] J. Singh, R. Singh, and P. Singh, "Morphological evaluation and sentiment analysis of Punjabi text using deep learning classification," *Journal King Saud University-Computer and Information Science*, 2018, doi: 10.1016/j.jksuci.2018.04.003.
- [21] Yelp Polarity Dataset, "TensorFlow Datasets Catalog homepage," 2015. [online]. Available: https://www.tensorflow.org/datasets/catalog/yelp_polarity_reviews
- [22] A. L. Maas, R.E. Daly, P.T. Pham, D. Huang, A.Y. Ng and C. Potts, "Learning Word Vectors for Sentiment Analysis", *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 142-150, 2011.
- [23] R. Sjögren, K. Stridh, T. Skotare, J. and J. Trygg, "Multivariate Patent Analysis-Using Chemometrics to Analyze Collections of Chemical and Pharmaceutical Patents," *Journal of Chemometrics*, vol. 34, pp. e3041, 2020, doi: 10.1002/cem.3041
- [24] A. Onan "Mining opinions from instructor evaluation reviews: A Deep Learning Approach, " *Computer Application in Engineering Education*, vol. 28, pp. 117–138, 2020, doi: 10.1002/cae.22179.
- [25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", *arXiv preprint*, arXiv:1301.3781, 2013.
- [26] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," *Neural Information Processing Systems Conference*, Lake Tahoe, pp. 3111–3119, 2013.
- [27] R. Ni and H. Cao, "Sentiment Analysis based on GloVe and LSTM-GRU," *39th Chinese Control Conference (CCC)*, Shenyang, China, pp. 7492-7497, 2020, doi: 10.23919/CCC50068.2020.9188578.
- [28] M. M. Saritas, A. Yasar, "Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 7, pp. 88-91, 2019, doi: 10.18201/ijisae.2019252786.
- [29] S. Qing, H. Wenjie and X. Wenfang, "Robust Support Vector Machine with Bullet Hole Image Classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 4, pp. 440-448, 2002, doi: 10.1109/TSMCC.2002.807277.
- [30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [31] S. Karita et al., "A Comparative Study on Transformer vs RNN in Speech Applications," *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, SG, Singapore, , pp. 449-456, 2019, doi: 10.1109/ASRU46091.2019.9003750.
- [32] L. M. Rojas-Barahona, "Deep Learning for Sentiment Analysis," *Language Linguistic Compass*, vol. 10, no. 12, 2016, doi: 10.1111/lnc3.12228
- [33] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015, doi: 10.1038/nature14539.
- [34] Ş. Kayıkçı, "A convolutional neural network model implementation for speech recognition," *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, vol. 7, no. 3, pp. 1892-1898, 2019, doi: 10.29130/dubited.567828.
- [35] M. S. Başarslan and F. Kayaalp, "Performance Analysis Of Fuzzy Rough Set-Based And

- Correlation-Based Attribute Selection Methods On Detection Of Chronic Kidney Disease With Various Classifiers," *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Istanbul, Turkey, 2019, pp. 1-5. doi: 10.1109/EBBT.2019.8741688.
- [36] K. Polat, and S. Güneş, "Breast cancer diagnosis using least square support vector machine," *Digital signal processing*, vol. 17, no. 4, pp. 694-701, 2007, doi: 10.1016/j.dsp.2006.10.008.