RESEARCH

MBi-GRUMCONV: A novel Multi Bi-GRU and Multi CNN-Based deep learning model for social media sentiment analysis

Muhammet Sinan Başarslan[1*] and Fatih Kayaalp[2]

Abstract

Today, internet and social media is used by many people, both for communication and for expressing opinions about various topics in many domains of life. Various artificial intelligence technologies-based approaches on analysis of these opinions have emerged natural language processing in the name of different tasks. One of these tasks is Sentiment analysis, which is a popular method aiming the task of analyzing people's opinions which provides a powerful tool in making decisions for people, companies, governments, and researchers. It is desired to investigate the effect of using multi-layered and different neural networks together on the performance of the model to be developed in the sentiment analysis task. In this study, a new, deep learning-based model was proposed for sentiment analysis on IMDB movie reviews dataset. This model performs sentiment classification on vectorized reviews using two methods of Word2Vec, namely, the Skip Gram and Continuous Bag of Words, in three different vector sizes (100, 200, 300), with the help of 6 Bidirectional Gated Recurrent Units and 2 Convolution layers (MBi-GRUMCONV). In the experiments conducted with the proposed model, the dataset was split into 80%-20% and 70%-30% training-test sets, and 10% of the training splits were used for validation purposes. Accuracy and F1 score criteria were used to evaluate the classification performance. The 95.34% accuracy of the proposed model has outperformed the studies in the literature. As a result of the experiments, it was found that Skip Gram has a better contribution to classification success.

Keywords Natural Language Processing, Sentiment Analysis, Word2Vec, Word Embedding, Deep Learning

Introduction

Especially for environmental and technological reasons, people's habits have started to change in recent years. In line with technological developments, socialization has almost entirely moved into social media applications through smartphones. Indeed, beyond the likes of photos and videos, people are now shooting and sharing videos that reflect their personal experiences. People instantly share their satisfaction and appreciation for a movie or a service in a restaurant, municipalities, and businesses, including public institutions. In addition to social media applications such as Twitter, and Facebook, they also share their ideas as videos on YouTube channels. They also share their experiences and comments with applications such as TripAdvisor and Airbnb. These shares account for a very large amount of data.

Approximately 7 TB of data on Twitter and approximately 10 TB of data on Facebook are accumulated every day [1]. The inadequacy of the processing power of traditional computer systems and the classical machine learning methods used for the analysis of these large amounts of data has led to the emergence of new technologies. While cloud computing technologies are becoming prevalent for higher processing power [2], deep learning

*Correspondence:
Muhammet Sinan Başarslan
muhammet.basarslan@medeniyet.edu.tr
[1] Department of Computer Engineering, Faculty of Engineering And Natural Sciences, Istanbul Medeniyet University, Istanbul 34680, Turkey
[2] Department of Computer Engineering, Engineering Faculty, Düzce University, Düzce 81620, Turkey

methods are also becoming prevalent for data analysis. One of the application areas of deep learning methods on texts is the sentiment analysis task of natural language processing (NLP).

Sentiment analysis is the classification of the analyzed text data for estimating sentiment polarity using NLP techniques. Tagging comments on social media applications and websites, especially Twitter, as positive and negative, and classifying them using NLP techniques through these tags is widely used in many areas for different purposes. As a recent technology, deep learning is also frequently seen in studies in this field.

Deep learning is a machine learning method based on an improved version of artificial neural network technology, which is inspired by the structure and function of the human brain. Unlike artificial neural networks, it consists of many more hidden layers. Consequently, more comprehensive and efficient learning can be achieved by creating more complex networks, similar to the human brain. Deep learning methods are used in many areas, such as image processing [3–6], text classification [7], speech recognition [8, 9], and NLP [10, 11].

In the deep learning methods used in many areas, different and multi layer neural networks are used. In one of the studies on the use of multi layer neural networks, geometric relational features based on distances between joints and selected lines were investigated using multi layer Long Short-Term Memory (LSTM) [12]. In their study, researchers created a model of energy consumption using a multi layer Gated Recurrent Unit (GRU) [13]. A new multi layer GRU neural network was also used to speed up the calculation of aerodynamic forces [14]. A multi-layer LSTM model was proposed for binary and multi class data on denial-of-service attacks [15]. Multi-layer Bidirectional LSTM (Bi-LSTM) was used in image processing. This method had better results than single-layer LSTM and GRU [16]. In a forecast study on daily power consumption controlled by the US electric company, a multi layer LSTM model was used, which performed better than the traditional single-layer LSTM and Bi-LSTM [17].

New models using multilayer neural networks have been proposed in many areas, from the energy industry to network security to image processing, and they give better results than single-layer neural network models.

In this study, a new multi layer deep learning model was proposed to investigate the effect of multi layer and different Artificial Neural Network (ANN) models on sentiment analysis performance. In the proposed model, a new deep learning model (MBi-GRUMCONV) was developed by using a combination of 6 Bi-GRU layers and 2 Convolution (Conv) layers.

The main contributions of this study are as follows:

- A new model (MBi-GRUMCONV) was proposed by using multi layer and different neural network architectures together to achieve the best performance in the sentiment analysis task.
- For the proposed method, the accuracy performance was investigated to reveal the effect of 3 different vector sizes (100, 200, 300) and 2 different Word2Vec word embedding methods (Skip Gram, Continuous Bag of Words (CBOW)) on sentiment classification performance.
- It is stated in the literature that while the Bi-GRU sequence gives good results in the analysis of data, CNN gives better results in image processing data. Bi-GRU, which found the most up-to-date solution to the vanishing gradient problem of RNN, was used as a multi-layer and then added 2 convolution layers to increase performance.

The rest of the article is organized as follows: Sect. "Related Works" presents a literature review. Under the heading of the methodology in Sect. "Methodology", the introduction of the dataset, data preprocessing, word embedding methods, and deep learning methods are discussed. The structure and details of the proposed model are given in Sect. "Proposed Model (MBi-GRUMCONV)". Sect. "Experiment Environments" presents the results and comments of the experiment carried out according to the proposed architecture and performance metric. The 6th and last section of the study discusses the overall evaluation of the study results and makes recommendations for future research.

Related Works

Existing studies with different approaches to SA task have been reviewed and detailed information about them have been given below.

Sentiment analysis is called by a few names, depending on its application areas, such as aspect-based, opinion extraction, sentiment mining, subjective analysis, and impact analysis. Idea mining and sentiment analysis are used interchangeably [18] The terms sentiment, opinion, and impact are used interchangeably. We can divide emotion classification into two as machine learning and dictionary-based approaches.

There are various levels in the studies of extracting emotions from texts. There are documented, sentence, aspect and concept levels. There are dictionary-based methods for approaches at the Concept [19–21], Document [22] level. In the sentence-level sentiment analysis studies, the document and sentence-level analysis does not affect the effect of the properties and aspects of the entities in the sentiment classification. For this, aspect based [18, 23, 24] methods are used. Aspect-based

provides more detailed and accurate information from the data by dividing it into small parts, taking into account the integrity of the emotion in the text [25].

Before emotion classification or just emotion identification studies are also done. Emotions such as happiness, disappointment, anger, and sadness can be determined in more detail, rather than standard positive, negative and neutral classes. In systems using emotion detection feature, thanks to these emotion classes, the user's attitude can be examined as positive or negative, and appropriate actions can be taken by analyzing the mood more clearly [25].

Sentiment analysis approaches, as seen in Fig. 1, can be generalized under four main headings as a dictionary-based approach, machine learning approach and their hybrid use, and other studies [20]. In our study, we followed the deep learning methods under supervised learning, one of the machine learning methods.

In the first studies with sentiment analysis (Term Frequency-Inverse Document Frequency), after frequency-based word embedding methods such as TF-IDF and Bag of Words, Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT) Multinominal Naïve Bayes (MNB), Bernoulli Naïve Bayes (BNB), Logistic Regression (LogR), and stochastic gradient descent (SGD) were used for emotion classification using machine learning methods. Afterwards, prediction-based methods such as Word2Vec, Global Vector (GloVe) started to be used. Bidirectional Encoder Representations from Transformers (BERT) has been used in other tasks of NLP, including the sentiment analysis task in pre-trained models such as CNN.

In a textual affective computing tasks such as sentiment analysis, supervised machine learning was used LogR, MNB, RF, SVM, Recurrent Neural Network (RNN), LSTM, and CNN [27–33] Studies are carried out with methods such as prompt-based analysis [34–36], graph-based [37–39]. There are also neurosymbolic AI [40–42] approaches that can be mentioned as new trends in sentiment analysis studies. Within the scope of the study, it is desired to investigate the performance of using multiple
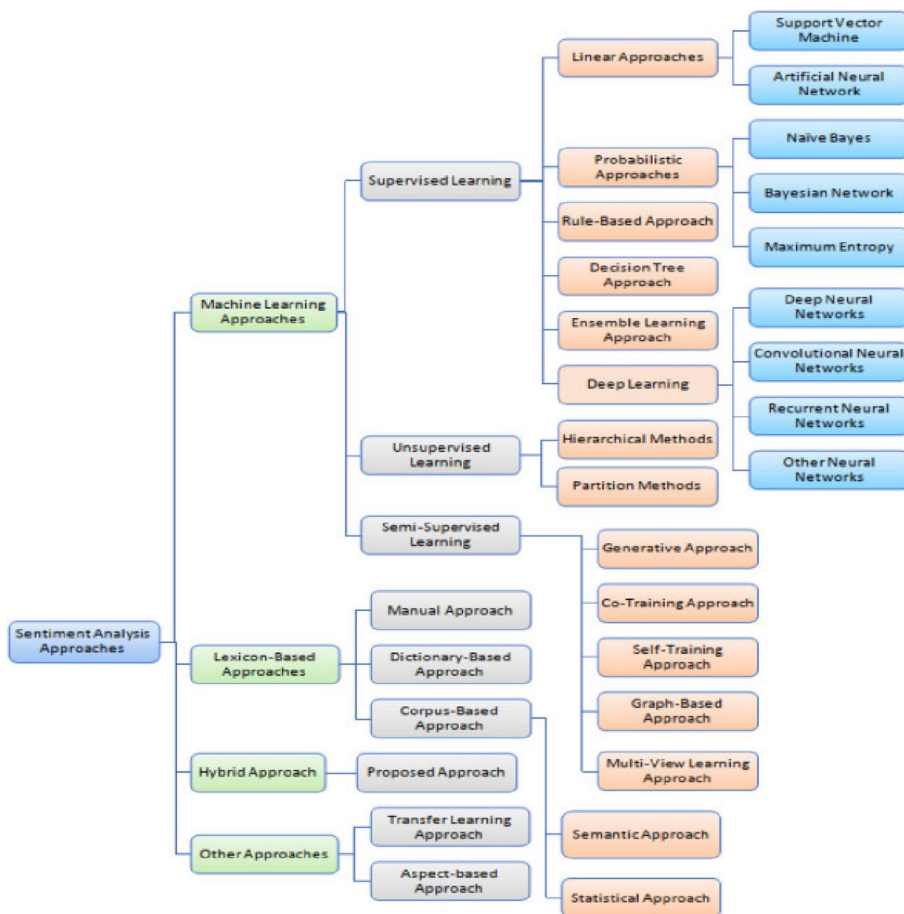

Fig. 1 Sentiment analysis approaches [26]

Table 1 Sentiment analysis studies on the IMDB dataset

Ref	Text Representation/Word Embedding	Model	Model description	Accuracy (%)
[43]	GloVe	Bi-LSTM -CNN	Multi Layered and Hybrid Deep Learning Model	92.05
[44]	GloVe	2 Bi-LSTM	Multi layer deep learning model	87.46
[45]	GLoVe	Bi-GRU, Bi-LSTM	Hybrid deep learning model	91.98
[46]	Word2Vec	CNN	Single layer neural network model	89.47
[47]	Keras embed	CNN-LSTM	Hybrid deep learning model	89.5
[48]	Keras embed	CNN-LSTM-CNN	Multi Layered Hybrid deep learning model	89.02
[49]	TF-IDF, Count Vectorizer, Keras embed	LogR, SVM, MNB, CNN, RNN, LSTM	Single layer deep learning model and traditional Machine Learning models	88 (RNN)
[50]	Word2Vec	RNN	Single layer neural network model	87
[51]	Word2Vec	CNN-LSTM	Hybrid deep learning model	88.3
[52]	GloVe	LSTM-GRU	Hybrid deep learning model	87.10
[53]	BOW	Multi Layer Perceptron	Single layer neural network model	86.67
[54]	Word2Vec	Multi Layer Perceptron, CNN, LSTM and CNN-LSTM	Hybrid deep learning model and Single layer neural network model	89.2 (CNN-LSTM)
[55]	Unigram with TF-IDF	MNB, BNB, LogR, SVM, SGD, RF	Deep learning models and traditional Machine Learning models	88.66 (RF)
[56]	D-dimensional dense vector, n inputs, feature map of d × n in size	Hierarchical Conv-Nets	Numerous NLP tasks have been studied	87.90

neural networks together in NLP tasks after the prediction-based based wor2vec method.

Among the sentiment analysis studies published in popular databases such as IEEE and Science Direct in the last 5 years, those applied to the IMDB reviews dataset was selected, and the details of these studies are presented in Table 1. The Relevant Table shows the word representation or embedding methods used in each study, the architectures used, the architecture descriptions, and the accuracy performance values in detail.

As shown in Table 1, various deep learning models were created by using the same or different multi-layer neural network architectures together. When these studies are examined, it is seen that the use of multiple layers leads to higher accuracy than single layers. We decided to use a multi-layered architecture to obtain higher performance in our study.

Methodology

In this section, the dataset used in the study is introduced, and preprocessing, word embedding and deep learning methods are discussed.

Dataset

The IMDB dataset, which contains 25,000 positive-tagged and 25,000 negative-tagged movie reviews, is a reliable, balanced, and popular dataset collected by Stanford researchers to be used in sentiment analysis studies [57]. The sentiment class distribution of the dataset is shown in Fig. 2.
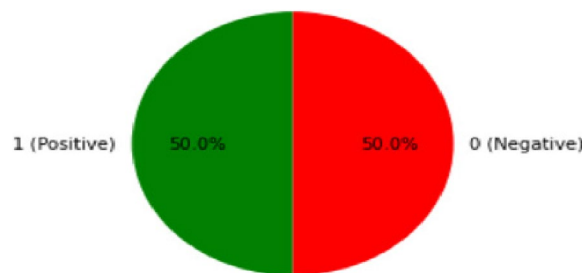

Fig. 2 IMDB dataset sentiment class distribution

Table 2 IMDB dataset

Attribute Name	Description	Attribute Type
Sentiment	Tagged sentiment classes of the texts	Integer
Review	Comments	Text

The use of the IMDB dataset was preferred to compare the results of our proposed architecture with previous studies in the literature. Table 2 shows information about the features of the IMDB dataset.

In Table 2, a sentiment attribute of 1 indicates a positive sentiment class, and 0 indicates a negative sentiment class. Table 3 shows more detailed information about the dataset.

Table 3 Dataset detailed information

Parameter	Values
Number of Classes	2
Number of Reviews-	50,000
Positive Reviews	25,000
Negative Reviews	25,000
Maximum Review Length	13,704
Minimum Review Length	6
Number of Unique Words	49,576

The word cloud of the negative and positive word distribution from the 50,000 comments in the dataset is shown in Fig. 3. As shown in Fig. 3, in the distribution of positive-tagged and negative-tagged data, words such as "good", and "great" are positive-tagged, and words such as "even", and "though" are negative-tagged. Words such as "film", "scene", and "story" are seen both in the positive and negative word clouds.

Data preprocessing

Text preprocessing is the preparation process for easier processing of texts. It includes processes such as the removal of stop words and special characters. It is used to prepare the data before the NLP classification stage.

To achieve better and more reliable classification results, only the most important words are taken or existing words are cleaned from HTML tags, and all are converted into normalized form. Preprocessing operations:

1. HTML tags were removed,
2. Special characters, URLs, and email addresses were removed.
3. Numbers were removed,
4. White spaces were removed,
5. Punctuations were removed,
6. Tokenization was performed,
7. All text data were converted to lowercase to eliminate variations in word forms.

8. Stopwords were removed,
9. Word2Vec's CBOW and Skip Gram methods were used in vector sizes of 100, 200, and 300 for vectorization operations.

Figure 4 shows the outline of the study, including preprocessing the dataset, word embedding operations, splitting the dataset into training, testing, and validation sets, feeding them into the proposed classification model, and evaluating the results. In the experiments carried
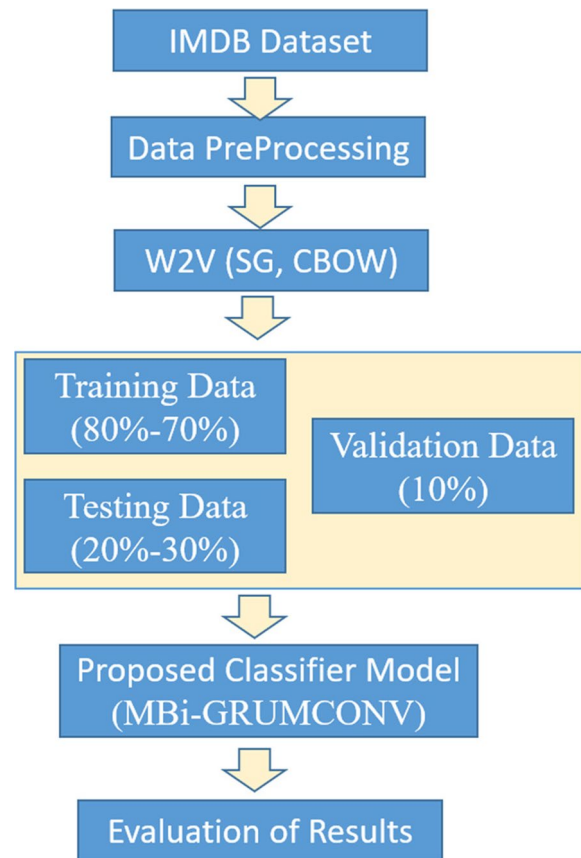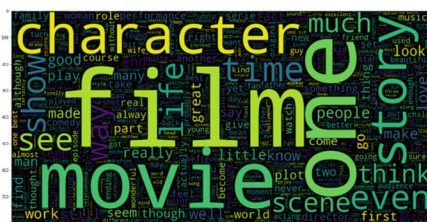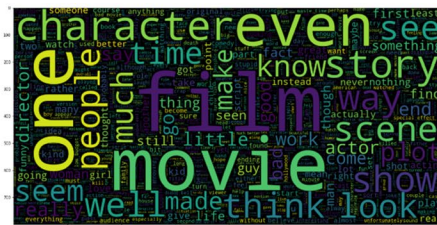
Fig. 4 Block diagram of the study

(a) Positive reviews   (b) Negative reviews

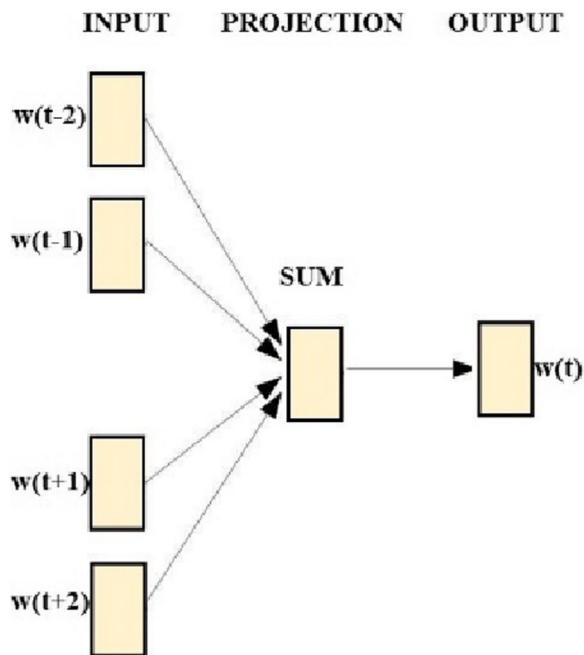**Fig. 3** Reviews dataset word clouds

Fig. 5 CBOW model structure

out to examine the performance of the proposed model in the study, the data set was divided into 80%-20% and 70%-30% training-test sets. Approximately 10% of the training data were used for validation.

Word Embedding

Word embedding is a method in which each word in a sentence is mapped to a d vector so that words with a close semantic similarity are represented closer together in a hidden space. This is a more advanced technique than the bag-of-words, which carries more context of the sentence in a low-dimensional space. The input to neural networks must be in numerical form. Therefore, the text is converted to a numeric representation called text vectorization. There are some useful methods used for text vectorization. One of the most popular techniques for word vectorization using neural networks is the technique. It was developed by Google. Word2Vec uses two methods: SkipGram and CBOW [58].

CBOW is a word embedding architecture that uses future words as well as past words to create a word embedding [56]. The objective function of the CBOW is given in Eq. (1):

$$J_\theta = \frac{1}{T} = \sum_{t=1}^{T} \log p\left(w_t | w_{t-n,\ldots\ldots}, w_{t-1}, w_{t+1},\ldots\ldots w_{t+n}\right) \qquad (1)$$

In the CBOW method, distributed representations of the context are used to predict the word in the middle

of the window [58]. A visual view of the CBOW model structure is shown in Fig. 5.

Skip Gram uses the central word to predict the surrounding words [58]. The objective function of the Skip Gram is given in Eq. (2):

$$J_\theta = \frac{1}{T} = \sum_{t=1}^{T} \sum_{-n \leq j \leq n, \neq 0} \log p\left(w_{j+1} | w_t\right) \qquad (2)$$

The log probabilities of the surrounding n words to the left and right of the w_t target word are summed for the objective function of the Skip Gram given in Eq. (2): The structure of the Skip Gram model is shown in Fig. 6.

Word2Vec parameters;

Size: It is the minimum number of occurrences of a word in the corpus to be included in the model.

Window: This is the maximum distance between the current and predicted words in a sentence.

Workers: It denotes the number of running threads for parallelization to speed up training.

min_count: The size of the feature vectors.

SG: If its value is 0, then CBOW is performed; otherwise, Skip Gram is performed.

Table 4 shows the Word2Vec parameters used in the study.

Deep learning

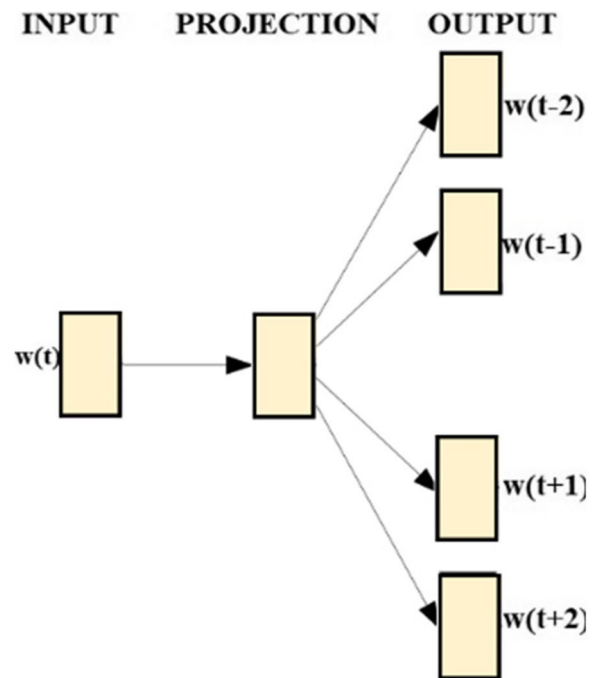Deep learning is a machine learning method based on artificial neural networks. It is based on at least one

Fig. 6 Skip gram model structure

Table 4 Word2Vec parameters used in the study

Parameters	Values
Vector size	100, 200, 300
Workers	8
min_count	5
Skip Gram (SG)	0, 1

artificial neural network and many algorithms to obtain new data from the available data. Deep learning can be carried out in a supervised, semi-supervised or unsupervised manner. It has been successfully applied to areas such as computer vision, machine vision, speech recognition, natural language processing, voice recognition, social network filtering, and machine translation [59].

RNN

It is a type of neural network that is often used in NLP problems. Successful results were obtained since it can also recall past information. The RNN uses the output of the previous step as the input to the current step. The RNN ensures that each output proceeds based on the previous step. It tries to keep the results calculated in the previous steps in memory [60].

The RNN architecture has one input layer, two hidden layers, and one output layer. All of these layers work independently. The structures in each layer have weights, and the specific threshold values are determined for each layer. In this way, the system will give more realistic results. As a result of these recurrent steps, the previous input state is stored and combined with the newly

obtained input value to associate the newly obtained input with the previous input. Visualization of the RNN architecture is shown in Fig. 6 [60].

Due to the problems of RNN, various networks of later variants, such as GRU and Bi-GRU, have been proposed. Information about GRU and Bi-GRU are given in subsections.

Gated Recurrent Unit Network

GRU is a type of Recurrent Neural Network. GRU, which consist of only two gates, a reset gate, and an update gate, use the hidden state to transfer information [60].

Update Gate: It decides the information to discard and the new information to include.

Reset Gate: This gate is used to decide how much of the past information is to be forgotten.

GRU are slightly faster than other types of RNN since they have fewer vector operations.

The model structure of the GRU is shown in Fig. 7. For a given time range t, $x_t \in R^{n*d}$ is the mini-batch input (n: sample count, d: number of inputs), and $h_{t-1} \in R^{n*h}$ is the hidden state of the last step (h: number of hidden states) [61].

GRU has Update and Reset Gates that control the flow of information within the unit. The Update Gate decides the information to discard and the information to include [61].

The calculation of the Update Gate is given in Eq. (3). This gate is used to decide how much of the past to forget. It also helps to capture short-term dependencies on time series.
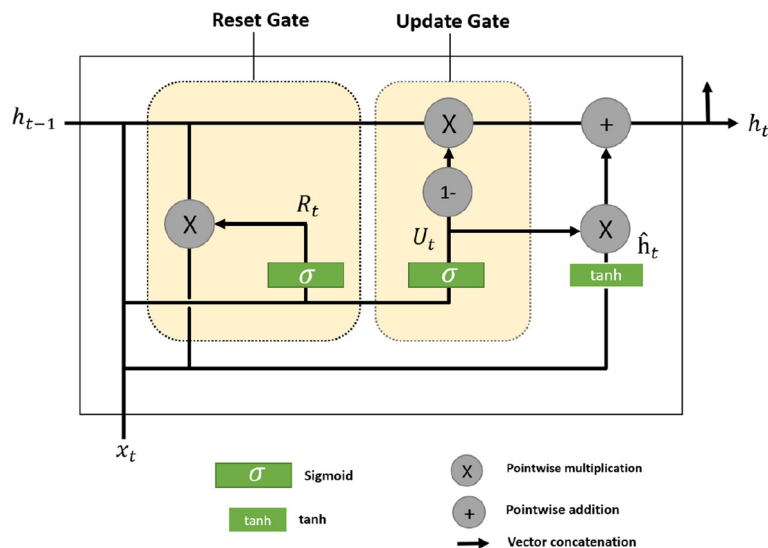

Fig. 7 GRU architecture

$$U_t = \sigma(W_U h_{t-1} + W_U x_t + b_U) \qquad (3)$$

The calculation of the reset gate is given in Eq. (4). A value close to 0 indicates that previous information is forgotten in the current memory content. A value close to 1 indicates that it will be retained in the current memory [62].

$$R_t = \sigma(W_R h_{t-1} + W_R x_t + b_R) \qquad (4)$$

After determining how much of the memory will be forgotten and how much will be kept through the reset and update gates at the moment t in Eq. (5), the information in these gates is scaled using the activation function [63].

$$h_t^{'} = \tanh \qquad (5)$$

The information stored by the hidden layer at time t is determined by Eq. (6).

$$h_t = (1 - U_t) * h_{t-1} + U_t * h_t^{'} \qquad (6)$$

Bi-GRU

Bidirectional recurrent neural networks bring together only two independent GRUs. This structure ensures that networks always have both backward and forward information about the array in each step.

The bidirectional GRU processes inputs in two ways, one from the past to the future and the other from the future to the past, and what distinguishes this approach from the unidirectional one is that the GRU working backward preserves information from the future and combines two hidden states. The Bi-GRU can preserve information from both the past and the future at any point in time. Figure 8 shows the schematic structure of the GRU and Bi-GRU side by side [64].

CNN

CNN is a class of deep, forward-propagation artificial neural networks most commonly used for image recognition and computer vision. It is a regular version of the multilayer perceptron. They were developed for image processing and image classification, but they were also used in text classification tasks. CNN uses convolution and pooling as the two main processes for feature extraction in the model. The output from convolution and pooling is connected to a fully connected multilayer perceptron [65].

A convolutional neural network consists of an input layer, hidden layers, and an output layer. In any forward-propagation neural network, any middle layer is called the hidden layer since its inputs and outputs are masked by the activation function and the final convolution. In a convolutional neural network, the hidden layers contain the layers that perform the convolutions. Typically, this involves a layer that performs a multiplication or another pointwise multiplication and consists of an activation function. This is followed by other layers, such as pooling layers, fully connected layers, and normalization layers [66].

Convolution is a process in which we select a filter or kernel of a predefined size and move that filter along the text values and multiply the corresponding values. The text in sentences is converted to numbers using the Word2Vec embedding technique. Then, all the values in the filter are added for the first feature in the resulting array. The filter moves along text values based on the stride size (which determines how many steps the filter can take each time it moves). The resulting array is called the convoluted feature [67].

Pooling is a technique used to downsample convoluted features. The global maximum pooling technique was chosen in the present study. This technique is used to reduce the size of the convoluted feature. After the convolution process, "same padding" was used to have the same size for both the array and the input.
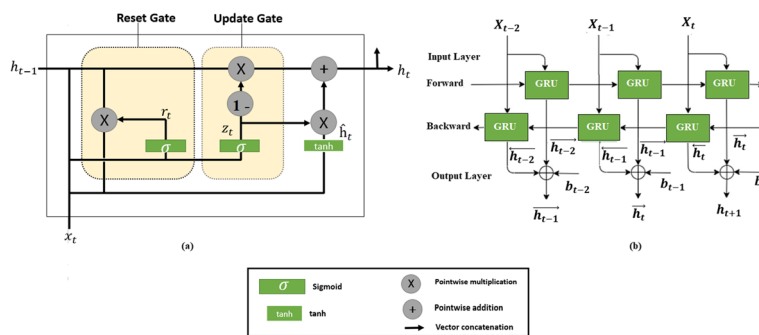

Fig. 8 **a**) GRU, **b**) Bi-GRU

Proposed Model (MBi-GRUMCONV)

In this section, a multilayered deep learning model is proposed for sentiment classification after the Word2Vec embedding process (Skip Gram and CBOW) in 3 different vector sizes (100, 200, and 300) on the public IMDB reviews dataset. The details of this proposed model, shown in Fig. 9, are discussed in this section.

Input Layer

The input layer is considered the first stage of the network. The input layer is the layer where the input data to be used in the model are imported.

Embedding layer

It is the second layer of the model and converts each list corresponding to a specific word in the information array into an element vector. These real-value vectors together form a structure. The embedding layer is designed so that each row represents a unique array corresponding to a specific word in the dictionary. The added network element is "d*w" in size. The dataset size is denoted by "d", and word weights are denoted by "w". Word2Vec (Skip Gram, CBOW) word vectors with 100, 200, and 300 vector sizes were used in the study.

Convolution, dropout, and dense layer

The convolution layer convolves its input obtained by calculating the pointwise multiplication between all input channels and filters based on the structure of the feature map. The dropout layer nullifies the connection of certain neurons in the next layer while leaving others unchanged. The dense layer is the layer of basic artificial neural network neurons.

Output layer

The output layer is the last layer in the model that yields negative or positive prediction results.

Experiment environments

Experiments on the established model were conducted on Google Colaboratory (Colab Pro) [68] using the TensorFlow 2.9.0 [69] and Keras 2.9 [70] libraries and Python version 3.9.13 [71]. The Colab Pro version was used to make the experiments faster without any interruption.

Hyperparameters settings

The ReducePlatue [72] and various callbacks of Keras [70] for early stopping [73] were used to create a model with a good fit.

If there is no improvement in the validation accuracy during the training of the model, the training process is stopped before the specified number of steps. This process is called early stopping. It is performed with the EarlyStopping function in the Keras library. The training is stopped if there is no improvement in the validation accuracy value for 5 steps.

If there is no improvement in the validation loss during training, the learning speed is reduced by multiplying by a specific coefficient. In the study, in the case of no improvement in validation loss performance in 5 steps, the learning speed was reduced by multiplying by 0.1.

TensorBoard is a tool of TensorFlow that enables tracking of experimental metrics such as loss and accuracy, model graph visualization, and fast and multiparameter model setup in machine learning studies [69]. Using the TensorBoard library, the parameters in the first column of Table 5 were run by assigning the values in the second column in nested loops to find the optimum value for each parameter. These optimum values are shown in bold in Table 5.

Summary information of the entire model determined by TensorBoard and used in the study is presented in Table 6.

Table 5 Parameters used in the model

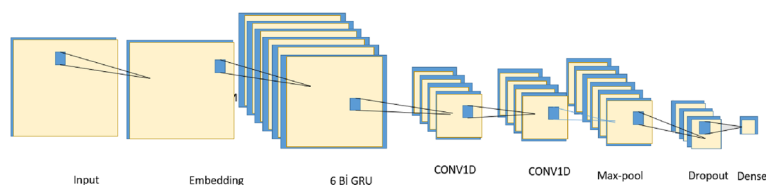| Parameters	Values
Batch Size	32,**64**,128,256,512,1024
Epoch	5,10,**20**,30,40
Activation Function	**ReLu**, sigmoid, tanh
Optimizer	**Adam,** RMSprop, SGD
Number of units and Number of Conv filters	64,100,**128**,200,300
Kernel	2,3,4,**5**,6,7
Dropout	**0.3**, 0.4, 0.5, 0.6, 0.7, 0.8

Fig. 9 The Proposed Mbi-GRUMCONV model for sentiment analysis

Table 6 Model parameter summary information

Hyperparameter	Values-Settings
Train size	70%, 80%
Test size	30%, 20%
Validation split	10%
Embedding Dimension	Word2Vec 100, 200, 300
Filter	128
Number of Units	128
Learning Rate	0.001
Optimizer	Adam
Batch Size	64
Number of Epochs	20
Recurrent Dropout	0.3
Max Length	200
Output Size	1
Metrics	Accuracy
Kernel Regularization	L2
Activation Function	ReLu
Loss Function	Binary Cross Entropy
Output Function	Sigmoid

Experiments

This section gives the results of the experiments of the proposed deep learning model conducted in 3 different (100–200-300) word embedding vector sizes with Word2Vec on 70%-30% and 80%-20% train-test split and 10% validation split for each set. Moreover, the performance metrics used in the evaluation of these experimental results are also addressed in this section.

Performance metrics

The performance of the models created for sentiment analysis was evaluated by accuracy criteria. This criterion is obtained by the ratio of the True Negatives (TN) and True Positives (TP) to the total number of samples in Eq. (7) [74].

$$\text{ACC} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{7}$$

F1 Score is used when there is a need for a measurement metric that will include not only False Negative F_N or False Positive F_P But also all error costs. F1 score is given in Eq. (8) [74].

$$F1 = 2 * \left(\frac{\frac{T_P}{T_P+F_P} * \frac{T_P}{T_P+F_N}}{\frac{T_P}{T_P+F_P} + \frac{T_P}{T_P+F_N}} \right) \tag{8}$$

Experimental results

The results of the experiments are given in Tables 7 and 8.

As shown in Table 7, in all dataset splits (train, test, and validation), the performance was found to increase in line with the increase in the vector size in both the Skip Gram and CBOW methods.

For the vector size of 100 in the CBOW and Skip Gram, the Skip Gram has better accuracy performance in the training and validation sets, while the CBOW has better results in the test data set. In the vector size of 200, however, CBOW was better in the training and test datasets, while Skip Gram was better in validation. At a vector size of 300, Skip Gram performed better than CBOW in all three sets (test, train,

Table 7 Accuracy values of the proposed model (80%-20%)

Performance Metric	Dataset Parts	CBOW			Skip Gram		
		100	200	300	100	200	300
Accuracy (%)	**Train**	92.80	93.20	94.38	92.42	91.08	**95.32**
	Validation	88.79	87.63	89.75	89.15	89.22	**94.67**
	Test	92.37	93.84	93.92	91.95	92.72	**95.34**

Table 8 Accuracy values of the proposed model (70%-30%)

Performance Metric	Dataset Parts	CBOW			Skip Gram		
		100	200	300	100	200	300
Accuracy (%)	**Train**	92.58	93.93	94.23	90.40	90.93	**94.68**
	Validation	89.03	89.17	89.49	89.29	89.43	**90.37**
	Test	90.47	93.40	93.13	91.83	91.69	**93.65**

Table 9 IMDB Dataset 80%-20% train and validation accuracy/loss graphs
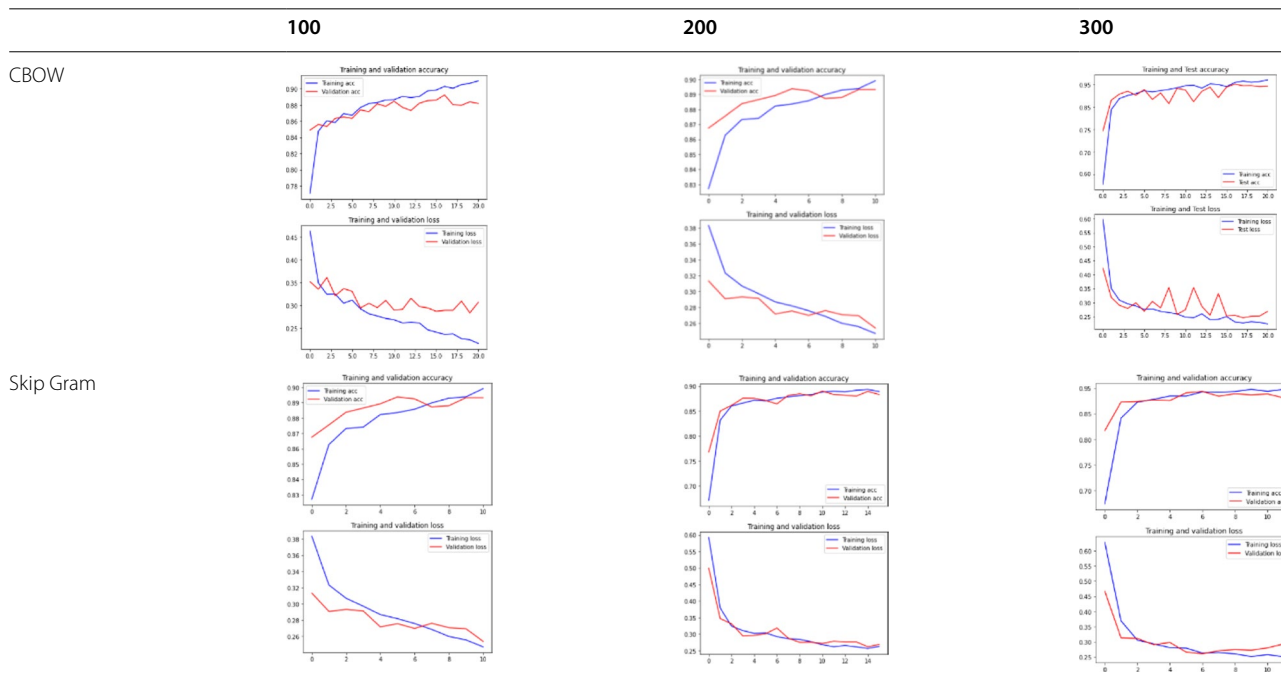
	100	200	300
CBOW			
Skip Gram			

and validation). The best result of the proposed model was obtained with a vector size of 300 in the Skip Gram embedding in all three data sets.

As shown in Table 8, in all dataset splits (training, testing, and validation), the performance was found to increase with the increase in the vector size in both the Skip Gram and CBOW methods.

For the vector size of 100 in the CBOW and Skip Gram, the Skip Gram has better accuracy performance in the test and validation sets, while the CBOW has better results in the training data set. In the vector

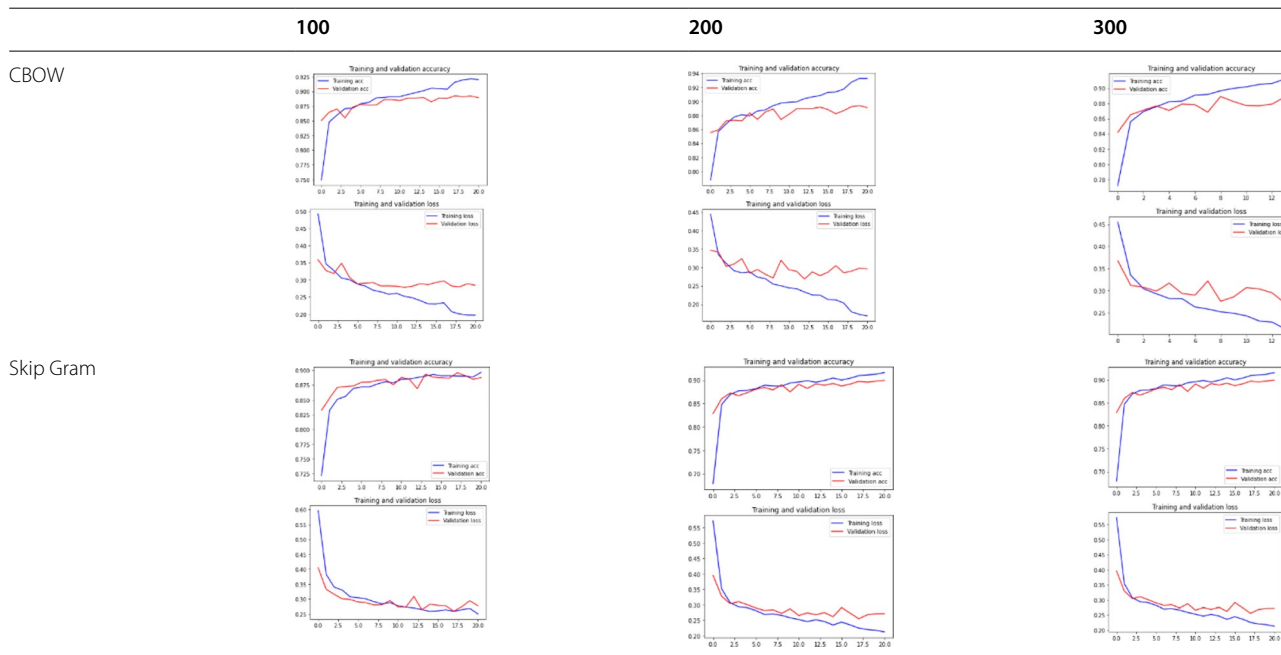**Table 10** IMDB Dataset 70%-30% train and validation accuracy/loss graphs

	100	200	300
CBOW			
Skip Gram			

Table 11 F1 score values of the proposed model

F1 Score	CBOW			Skip Gram		
Training–Testing and validation splits	100	200	300	100	200	300
80–20	0.89	0.87	0.91	0.89	0.88	0.92
70–30	0.90	0.87	0.92	0.91	0.89	0.93

size of 200, however, CBOW was better in the training and test datasets, while Skip Gram was better in validation. At a vector size of 300, Skip Gram performed better than CBOW in all three sets (test, train, and validation). The best result of the proposed model was obtained with a vector size of 300 in the Skip Gram embedding in all three data sets.

Considering Table 7 and Table 8 together, 80%-20% (Table 9) gave better results than the results obtained in the 70%-30% train-test separation. During the two different 80%-20% and 70%-30% (Table 9, Table 10) train-test splits, the performance was found to increase as the vector size increased in both the Skip Gram and CBOW methods. Although the best results in both separations were obtained by the Skip Gram method and at a vector size of 300, the 80%-20% train-test separation gave the best result in the proposed model.

Table 12 Comparison of the test accuracy results of the proposed model with those of previous studies

Author	Model	Accuracy(%)
Derra [44]	Bi-LSTM-Bi-LSTM	87.46
Pushpakumara et al. [50]	RNN	87
Ni and Cao [52]	LTM-GRU	87.10
Le and Mikolov [58]	Paragraph Vector	87.8
Wenpeng et al. [56]	Hieachical Con-vNets	87.90
Nguyen et al. [65]	SVM-NB	87.95
Amulya et al.[49]	RNN	88
Hassan and Mahmood [52]	Conv-Lstm	88.3
Mohaiminul [54]	CNN-LSTM	89.2
Narayanan et al. [76]	NB	88,80
Camacho-Collados and Pilehvar [77]	CNN-LSTM	88.90
Ghorbani, et al. [48]	CNN-LSTM-CNN	89.02
Zhou and Feng [78]	DT	89.16
Wang and Manning [79]	SVM	89.16
Nahal et al. [55]	CNN-LSTM	89.20
Ma and Fan [80]	Bi-LSTM with Attention mechanism	89.29
Sabba et al. [46]	Sequential CNN	89.47
Mesnil et al. [81]	NB-SVM	91.87
Shaukat et al. [53]	Multi Layer Perceptron	91.9
Islam [45]	Bi-GRU-Bi-LSTM	91.98
Amit [43]	MBi-LSTMGLoVe	92.05
Brychcín and Habernal [82]	ME	92.24
Xu et al. [83]	RNN	92.77
Dai and Le [14]	LSTM	92.80
Radford et al. [84]	LSTM	92.88
Johnson and Zhang [86]	CNN	93.49
Dieng et al. [87]	RNN	93.72
Johnson and Zhang [88]	LSTM	94.06
Gray et al. [89]	LSTM	94.99
Proposed Model	**MBi-GRUMCONV**	**95.34**

Table 13 Comparison of the validation accuracy results of the proposed model with those of previous studies

Author	Model	Accuracy (%)
Chen et al. [90]	CNN-Bi-LSTM	82.30
Shaukat et al. [53]	Multi Layer Perceptron	86.67
Pang and Lee [7]	NB-SVM	87.20
Derra [44]	2 Bi-LSTM	88.36
Ghorbani, et al. [48]	CNN-LSTM-CNN	89.02
Rathor and Agrawal [91]	Temporal CNN	89.27
Yenter and Verma [47]	CNN-LSTM	89.50
Ghosh [86]	Linear SVM	91.10
Amit [43]	MBi-LSTMGLoVe	93.55
Proposed Model	**MBi-GRUMCONV**	**94.67**

Accuracy and loss graphs for all embedding and vectors of the proposed model are presented in Table 9 and Table 10.

Table 9 shows that Skip Gram is more robust than CBOW in all vector sizes (100, 200, 300).

Table 10 shows that the Skip Gram yields a good fit/robust model, better than CBOW in all vector sizes (100, 200, and 300).

F1 Score value is given in Table 11.

As seen in Table 11, Skip Gram method gave better results than CBOW embedding method, similar to the Accuracy values.

Comparison of the proposed model (MBi-GRUMCONV) with related works

The experimental results of the proposed model, state-of-the-art models proposed in other studies in the literature, author information, and the comparison of the accuracy results of the test sets are given in Table 12.

The validation accuracy value of the model we proposed is specifically compared to the proposed multilayered models in Table 13. In these comparisons, it is seen that it surpasses other studies.

Conclusion and Discussion

A new model was proposed based on deep learning with word embedding using Word2Vec on the IMDB reviews dataset. Multilayered Bi-GRU and Conv were used as deep learning methods. The proposed model uses 6 Bi-GRUs followed by 2 Conv. During the experiments, the data set was used as 80%-20%-10% and 70%-30%-10% training-test-validation splits, and the results are presented in two separate tables (Table 7, Table 8). As expected, it was seen that results with higher accuracy were obtained in the 80%-20%-10% split, which has larger training data.

The proposed model was found to have a 95.32% training accuracy, 94.67% validation accuracy, and 95.34% test accuracy with a vector size of 300 in the Word2Vec Skip Gram method. These values yielded a higher performance compared to state-of-the-art studies.

One of the reasons for this performance increase is that Word2Vec works more efficiently with Skip Gram, which uses the sum of probabilities of the surrounding words to the left and right of the target word, unlike CBOW, which uses distributed representations of context to predict the word in the middle of the window. Another reason is that the increase in the vector dimension has a positive effect on the classification.

According to the results of both the literature studies and the proposed model, it was seen that multilayered and hybrid models gave better results compared to single-layer neural network models. It is recommended that those who will conduct research in this field use neural network architectures in multilayer and hybrid structures.

Future studies should investigate the effects of different deep learning models on sentiment analysis performance with different word representation methods and different classifiers In future work, we will be looking at hybrid approaches to investigate the affects of different learning techniques such as supervised and semi-supervised learning methods on enhancing the sentiment analysis accuracy.

Authors' contributions
Fatih Kayaalp: Defining the methodology, evaluations of the results and draft editing, the authors read and approved the final manuscript. Muhammet Sinan Başarslan: Preprocessing the dataset, data analysis, experiments and evaluations, manuscript draft preparation. The authors read and approved the final manuscript.

Code availability
Custom Code.

Availability of data and materials
Public datasets are used. The dataset is shared as open source, collected and labeled by Stanford researchers https://ai.stanford.edu/~amaas/data/sentiment/.

Declarations

Ethics approval and consent to participate
No study was carried out that required ethical approval. No information from human beings was used in the study.

Consent for publication
No information from human beings is used in the study.

References

1. "Global social media statistics research summary 2022." https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/. Accessed 20 Sept 2022
2. "Ground radar precipitation estimation with deep learning approaches in meteorological private," doi: https://doi.org/10.1186/s13677-020-00167-w.
3. Kabakus AT, Erdogmus P (2022) An experimental comparison of the widely used pre-trained deep neural networks for image classification tasks towards revealing the promise of transfer-learning. Concurr Comput Pract Exp. https://doi.org/10.1002/cpe.7216
4. A. Sevik, P. Erdogmus, and E. Yalein, "Font and Turkish Letter Recognition in Images with Deep Learning," in 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), 2018, pp. 61–64, doi: https://doi.org/10.1109/IBIGDELFT.2018.8625333.
5. Alhudhaif A, Polat K, Karaman O (2021) Determination of COVID-19 pneumonia based on generalized convolutional neural network model from chest X-ray images. Expert Syst Appl 180:115141. https://doi.org/10.1016/j.eswa.2021.115141
6. N. Calik and B. U. Toreyin, "Image Analysis," 2022, pp. 201–238.
7. B. Pang and L. Lee, "A sentimental education," in Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics - ACL '04, 2004, pp. 271-es, doi: https://doi.org/10.3115/1218955.1218990.
8. Y. Zhang et al., "BigSSL: Exploring the Frontier of Large-Scale Semi-Supervised Learning for Automatic Speech Recognition," IEEE J. Sel. Top. Signal Process., pp. 1–14, 2022, doi: https://doi.org/10.1109/JSTSP.2022.3182537.
9. Mridha MF, Ohi AQ, Hamid MA, Monowar MM (2022) A study on the challenges and opportunities of speech recognition for Bengali language. Artif Intell Rev 55(4):3431–3455. https://doi.org/10.1007/s10462-021-10083-3
10. S. Meera and S. Geerthik, "Natural Language Processing," in Artificial Intelligent Techniques for Wireless Communication and Networking, Wiley, 2022, pp. 139–153.
11. K. R. Chowdhary, "Natural Language Processing," in Fundamentals of Artificial Intelligence, New Delhi: Springer India, 2020, pp. 603–649.
12. S. Zhang, X. Liu, and J. Xiao, "On Geometric Features for Skeleton-Based Action Recognition Using Multilayer LSTM Networks," in 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Mar. 2017, pp. 148–157, doi: https://doi.org/10.1109/WACV.2017.24.
13. Mahjoub S, Chrifi-Alaoui L, Marhic B, Delahoche L (2022) Predicting Energy Consumption Using LSTM, Multi-Layer GRU and Drop-GRU Neural Networks. Sensors 22(11):4062. https://doi.org/10.3390/s22114062
14. Dai Y, Rong H, Wu Y, Yang C, Xu Y (2022) Stall flutter prediction based on multi-layer GRU neural network. Chinese J Aeronaut. https://doi.org/10.1016/j.cja.2022.07.011
15. V. Gaur and R. Kumar, "DDoSLSTM: Detection of Distributed Denial of Service Attacks on IoT Devices using LSTM Model," in 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), Mar. 2022, pp. 01–07, doi: https://doi.org/10.1109/IC3IOT53935.2022.9767889.
16. S. Malik, P. Bansal, P. Sharma, R. Jain, and A. Vashisht, "Image Retrieval Using Multilayer Bi-LSTM," 2022, pp. 745–755.
17. Chi D (2022) Research on electricity consumption forecasting model based on wavelet transform and multi-layer LSTM model. Energy Rep 8:220–228. https://doi.org/10.1016/j.egyr.2022.01.169
18. Rana TA, Cheah YN (2016) Aspect extraction in sentiment analysis: comparative analysis and survey. Artif Intell Rev 46:459–483. https://doi.org/10.1007/s10462-016-9472-z]
19. D'Aniello G, Gaeta M, La Rocca I (2022) KnowMIS-ABSA: an overview and a reference model for applications of sentiment analysis and aspect-based sentiment analysis. Artif Intell Rev 55:5543–5574. https://doi.org/10.1007/s10462-021-10134-9
20. Hemmatian F, Sohrabi MK (2019) A survey on classification techniques for opinion mining and sentiment analysis. Artif Intell Rev 52(3):1495–1545. https://doi.org/10.1007/s10462-017-9599-6
21. S. Rosenthal, N. Farra, P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter. In: Proceedings of the 11th international workshop on semantic evaluation," Association for Computational Linguistics, Vancouver, Canada, 2017, pp 502–518. https://doi.org/10.18653/v1/S17-2088
22. S. Behdenna, F. Barigou, G. Belalem, "Document level sentiment analysis: a survey," EAI Endorsed Trans Context Aware Syst Appl. https://doi.org/10.4108/eai.14-3-2018, 154339
23. E. Cambria, D. Das, S. Bandyopadhyay, A. Feraco, A. "Affective Computing and Sentiment Analysis", In: Cambria, E., Das, D., Bandyopadhyay, S., Feraco, A. (eds) A Practical Guide to Sentiment Analysis. Socio-Affective Computing,2016 vol 5. Springer, Cham. https://doi.org/10.1007/978-3-319-55394-8_1
24. S. Poria, A. Cambria, and A. Gelbukh, A. "Aspect extraction for opinion mining with a deep convolutional neural network,". Knowledge-Based Systems, 2017, 108, 42–49.
25. B. Liang, H. Su, ., Gui, E. Cambria, and R. Xu, "Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks," Knowledge-Based Systems, 2022, 235, 107643.
26. Birjali M, Kasri M, M., & Beni-Hssane, A. (2021) A comprehensive survey on sentiment analysis: Approaches, challenges and trends. Knowl-Based Syst 226:107134
27. Li W, Guo K, Shi Y, Zhu L, Zheng Y (2018) DWWP: Domainspecific new words detection and word propagation system for sentiment analysis in the tourism domain. Knowl-Based Syst 146:203–214
28. Ye Q, Zhang Z, Law R (2009) Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Syst Appl 36(3):6527–6535
29. D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in Proceedings of Annual Meeting of the Association for Computational Linguistics, 2014, pp. 1555–1565.
30. Jain VK, Kumar S, Fernandes SL (2017) Extraction of emotions from multilingual text using intelligent text processing and computational linguistics. Journal of computational science 21:316–326
31. Liu Q, Lu J, Zhang G, Shen T, Zhang Z, Huang H (2021) Domain-specific meta-embedding with latent semantic structures. Inf Sci 555:410–423
32. Q. Liu, H.-Y. Huang, Y. Gao, X. Wei, Y. Tian, and L. Liu, "Taskoriented word embedding for text classification," in Proceedings of the 27th international conference on computational linguistics, 2018, pp. 2023–2032.
33. W. Li, W. Shao, S. Ji, and E. Cambria, "BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis," Neurocomputing, vol. 467, pp. 73–82, 2022.]
34. Li, C., Gao, F., Bu, J., Xu, L., Chen, X., Gu, Y., ... & Yu, Z. (2021). Sentiprompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. arXiv preprint arXiv:2109.08306, Dang,
35. E., Hu, Z., & Li, T. (2022). Enhancing Collaborative Filtering Recommender with Prompt-Based Sentiment Analysis. arXiv preprint arXiv:2207.12883.,
36. Mao, R., Liu, Q., He, K., Li, W., & Cambria, E. (2022). The biases of pre-trained language models: An empirical study on prompt-based sentiment analysis and emotion detection. IEEE Transactions on Affective Computing
37. Sonawane, S. S., & Kulkarni, P. A. (2014). Graph based representation and analysis of text document: A survey of techniques. International Journal of Computer Applications, 96(19).]
38. Yu, B., & Zhang, S. (2022). A novel weight-oriented graph convolutional network for aspect-based sentiment analysis. The Journal of Supercomputing, 1–26.
39. P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," 2021, arXiv preprint arXiv:2107.13586.
40. E. Cambria, "Affective Computing and Sentiment Analysis," in IEEE Intelligent Systems, vol. 31, pp. 102–107, Mar.-Apr. 2016, doi: https://doi.org/10.1109/MIS.2016.31.
41. M. Gaur, K. Gunaratna, S. Bhatt and A. Sheth, "Knowledge-Infused Learning: A Sweet Spot in Neuro-Symbolic AI," in IEEE Internet Computing, vol. 26, no. 4, pp. 5–11, 1 July-Aug. 2022, doi: https://doi.org/10.1109/MIC.2022.3179759.

42. Cambria, E., Liu, Q., Decherchi, S., Xing, F., & Kwok, K. (2022). SenticNet 7: a commonsense-based neurosymbolic AI framework for explainable sentiment analysis. Proceedings of LREC 2022.

43. A. Pimpalkar and J. R. Raj R, "MBiLSTMGloVe: Embedding GloVe knowledge into the corpus using multi-layer BiLSTM deep learning model for social media sentiment analysis," Expert Syst. Appl., vol. 203, p. 117581, Oct. 2022, doi: https://doi.org/10.1016/j.eswa.2022.117581.

44. N. D. Derra and D. Baier, "Working in Detail: How LSTM Hyperparameter Selection Influences Sentiment Analysis Results," Arch. Data Sci. Ser. A, 6(1), 10., vol. 6, no. 1, pp. 1–22, 2020, doi: https://doi.org/10.5445/KSP/1000098011/10.

45. Islam MS, Ghani NA (2022) A Novel BiGRUBiLSTM Model for Multilevel Sentiment Analysis Using Deep Neural Network with BiGRU-BiLSTM. Lect Notes Electr Eng 730(July):403–414. https://doi.org/10.1007/978-981-33-4597-3_37

46. S. Sabba, N. Chekired, H. Katab, N. Chekkai, and M. Chalbi, "Sentiment Analysis for IMDb Reviews Using Deep Learning Classifier," in 2022 7th International Conference on Image and Signal Processing and their Applications (ISPA), 2022, pp. 1–6, doi: https://doi.org/10.1109/ISPA54004.2022.9786284.

47. A. Yenter and A. Verma, "Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis," in 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), Oct. 2017, pp. 540–546, doi: https://doi.org/10.1109/UEMCON.2017.8249013.

48. Ghorbani M, Bahaghighat M, Xin Q, Özen F (2020) ConvLSTMConv network: a deep learning approach for sentiment analysis in cloud computing. J Cloud Comput 9(1):16. https://doi.org/10.1186/s13677-020-00162-1

49. K. Amulya, S. B. Swathi, P. Kamakshi, and Y. Bhavani, "Sentiment Analysis on IMDB Movie Reviews using Machine Learning and Deep Learning Algorithms," in 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), Jan. 2022, pp. 814–819, doi: https://doi.org/10.1109/ICSSIT53264.2022.9716550.

50. S. prabha.K.S and P. N. Karthikayan, "For Movie Reviews, A Sentiment Analysis using Long Short Term Memory Networks," Turkish J. Comput. Math. Educ., vol. 12, no. 9, pp. 1758–1766, 2021.

51. R. Ni and H. Cao, "Sentiment Analysis based on GloVe and LSTM-GRU," in 2020 39th Chinese Control Conference (CCC), Jul. 2020, pp. 7492–7497, doi: https://doi.org/10.23919/CCC50068.2020.9188578.

52. A. Hassan and A. Mahmood, "Deep Learning approach for sentiment analysis of short texts," in 2017 3rd International Conference on Control, Automation and Robotics (ICCAR), Apr. 2017, pp. 705–710, doi: https://doi.org/10.1109/ICCAR.2017.7942788.

53. Shaukat Z, Zulfiqar AA, Xiao C, Azeem M, Mahmood T (2020) Sentiment analysis on IMDB using lexicon and neural networks. SN Appl Sci 2(2):148. https://doi.org/10.1007/s42452-019-1926-x

54. Mohaiminul M, Sultana N (2018) Comparative Study on Machine Learning Algorithms for Sentiment Classification. Int J Comput Appl 182(21):1–7. https://doi.org/10.5120/ijca2018917961

55. N. Mohamed Ali, M. M. A. El Hamid, and A. Youssif, "Sentiment Analysis for Movies Reviews Dataset Using Deep Learning Models," Int. J. Data Min. Knowl. Manag. Process, vol. 09, no. 03, pp. 19–27, 2019, doi: https://doi.org/10.5121/ijdkp.2019.9302.

56. W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative Study of CNN and RNN for Natural Language Processing," 2017, [Online]. Available: http://arxiv.org/abs/1702.01923.

57. A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," ACL-HLT 2011 - Proc. 49th Annu. Meet. Assoc. Comput. Linguist. Hum. Lang. Technol., vol. 1, pp. 142–150, 2011.

58. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations ofwords and phrases and their compositionality," Adv. Neural Inf. Process. Syst., pp. 1–9, 2013.

59. Li D, Du L (2022) Recent advances of deep learning algorithms for aquacultural machine vision systems with emphasis on fish. Artif Intell Rev 55(5):4077–4116. https://doi.org/10.1007/s10462-021-10102-3

60. Q. Yao, D. D.-C. Lu, and G. Lei, "A Surface Temperature Estimation Method for Lithium-ion Battery Using Enhanced GRU-RNN," IEEE Trans. Transp. Electrif., pp. 1–1, 2022, doi: https://doi.org/10.1109/TTE.2022.3197927.

61. Ni Q, Cao X (2022) MBGAN: An improved generative adversarial network with multi-head self-attention and bidirectional RNN for time series

62. imputation. Eng Appl Artif Intell 115:105232. https://doi.org/10.1016/j.engappai.2022.105232

62. Canli H, Toklu S (2021) Deep Learning-Based Mobile Application Design for Smart Parking. IEEE Access 9:61171–61183. https://doi.org/10.1109/ACCESS.2021.3074887

63. Canlı H, Toklu S (2022) Design and Implementation of a Prediction Approach Using Big Data and Deep Learning Techniques for Parking Occupancy. Arab J Sci Eng 47(2):1955–1970. https://doi.org/10.1007/s13369-021-06125-1

64. Yan J, Liu J, Yu Y, Xu H (2021) Water Quality Prediction in the Luan River Based on 1-DRCNN and BiGRU Hybrid Neural Network Model. Water 13(9):1273. https://doi.org/10.3390/w13091273

65. Nguyen VQ, Anh TN, Yang H-J (2019) Real-time event detection using recurrent neural network in social sensors. Int J Distrib Sens Networks 15(6):155014771985649. https://doi.org/10.1177/1550147719856492

66. Casillo F, Deufemia V, Gravino C (2022) Detecting privacy requirements from User Stories with NLP transfer learning models. Inf Softw Technol 146:106853. https://doi.org/10.1016/j.infsof.2022.106853

67. A. Onan, "Bidirectional convolutional recurrent neural network architecture with group-wise enhancement mechanism for text sentiment classification," J. King Saud Univ. - Comput. Inf. Sci., vol. 34, no. 5, pp. 2098–2117, 2022, doi: https://doi.org/10.1016/j.jksuci.2022.02.025.

68. Google LLC, "Colab." https://colab.research.google.com/. Accessed 20 Sept 2022

69. Google, "Tensorflow," 2021. https://tensorflow.org. Accessed 20 Sept 2022

70. Keras, "Keras." https://www.keras.io. Accessed 20 Sept 2022

71. Python, "Python." https://www.python.org/downloads/. Accessed 20 Sept 2022

72. RedecuPlatue, "RedecuPlatue." https://keras.io/api/callbacks/reduce_lr_on_plateau/. Accessed 20 Sept 2022

73. E. Stopping, "Early Stopping," https://keras.io/api/callbacks/early_stopping/. Accessed 20 Sept 2022

74. Ağbulut Ü, Gürel AE, Biçen Y (2021) Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison. Renew Sustain Energy Rev 135:110114. https://doi.org/10.1016/j.rser.2020.110114

75. D. Q. Nguyen, T. Vu, and S. B. Pham, "Sentiment Classification on Polarity Reviews: An Empirical Study Using Rating-based Features," in Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2014, pp. 128–135, doi: https://doi.org/10.3115/v1/W14-2621.

76. V. Narayanan, I. Arora, and A. Bhatia, "Fast and Accurate Sentiment Classification Using an Enhanced Naive Bayes Model," 2013, pp. 194–201.

77. J. Camacho-Collados and M. T. Pilehvar, "On the Role of Text Preprocessing in Neural7Network Architectures: An Evaluation Study on Text Categorization and Sentiment Analysis," in Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 2018, pp. 40–46, doi: https://doi.org/10.18653/v1/W18-5406.

78. Zhou ZH, Feng J (2019) Deep forest. Natl Sci Rev 6(1):74–86. https://doi.org/10.1093/nsr/nwy108

79. S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," 50th Annu. Meet. Assoc. Comput. Linguist. ACL 2012 - Proc. Conf., vol. 2, no. July, pp. 90–94, 2012.

80. Ma Y, Fan H, Zhao C (2019) Feature-Based Fusion Adversarial Recurrent Neural Networks for Text Sentiment Classification. IEEE Access 7:132542–132551. https://doi.org/10.1109/ACCESS.2019.2940506

81. G. Mesnil, T. Mikolov, M. A. Ranzato, and Y. Bengio, "Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews," 3rd Int. Conf. Learn. Represent. ICLR 2015 - Work. Track Proc., pp. 1–5, 2015.

82. T. Brychcín and I. Habernal, "Unsupervised improving of sentiment analysis using global target context," Int. Conf. Recent Adv. Nat. Lang. Process. RANLP, no. September, pp. 122–128, 2013.

83. W. Xu, H. Sun, C. Deng, and Y. Tan, "Variational Autoencoder for Semi-Supervised Text Classification," Proc. AAAI Conf. Artif. Intell., vol. 31, no. 1, Feb. 2017, doi: https://doi.org/10.1609/aaai.v31i1.10966.

84. A. Radford, R. Jozefowicz, and I. Sutskever, "Learning to Generate Reviews and Discovering Sentiment," 2017, [Online]. Available: http://arxiv.org/abs/1704.01444.

85. Monnet A, Cabadi Y (1978) Automatized Hemagglutination Kinetics. Vox Sang 34(4):227–230. https://doi.org/10.1111/j.1423-0410.1978.tb02472.x

86. Johnson R, Zhang T (2015) "Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding," in Advances in Neural Information Processing Systems, vol 28. Available: https://proceedings.neurips.cc/paper/2015/file/acc3e0404646c57502b480dc052c4fe1-Paper.pdf. Accessed 1 Oct 2022

87. A. B. Dieng, J. Gao, C. Wang, and J. Paisley, "TopicRNN: A recurrent neural network with long-range semantic dependency," 5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc., pp. 1–13, 2017.

88. R. Johnson and T. Zhang, "Supervised and semi-supervised text categorization using LSTM for region embeddings," 33rd Int. Conf. Mach. Learn. ICML 2016, vol. 2, pp. 794–802, 2016.

89. Gray S, Radford A, Kingma DP (2017) "GPU Kernels for Block-Sparse Weights," OpenAI 17;1:12. Available: https://blog.openai.com/block-sparse-gpu-kernels. http://www.jneurosci.org/cgi/. https://doi.org/10.1523/JNEUROSCI.1088-12.2012. https://github.com/openai/blocksparse. Accessed 1 Oct 2022

90. Chen T, Xu R, He Y, Wang X (2017) Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. Expert Syst Appl 72:221–230. https://doi.org/10.1016/j.eswa.2016.10.065

91. Rathor S, Agrawal S (2022) Sense understanding of text conversation using temporal convolution neural network. Multimed Tools Appl 81(7):9897–9914. https://doi.org/10.1007/s11042-022-12090-0

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.