Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(5): 8766–8781.
DOI: 10.3934/mbe.2023385
Received: 13 December 2022
Revised: 01 March 2023
Accepted: 02 March 2023
Published: 07 March 2023

Research article

A novel sentiment analysis method based on multi-scale deep learning

Qiao Xiang[1], **Tianhong Huang**[2,*], **Qin Zhang**[1], **Yufeng Li**[3], **Amr Tolba**[4] **and Isack Bulugu**[5]

[1] School of Artificial Intelligence, Chongqing Technology and Business University, Chongqing 400067, China

[2] School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331-5501, USA

[3] School of Mechanical Engineering, Chongqing Technology and Business University, Chongqing 400067, China

[4] Computer Science Department, Community College, King Saud University, Riyadh 11437, Saudi Arabia

[5] Department of Electronics and Telecommunications Engineering, College of ICT, University of Dar es Salaam, Dar es Salaam, Tanzania

* **Correspondence:** Email: huantian@oregonstate.edu.

Abstract: As the college students have been a most active user group in various social media, it remains significant to make effective sentiment analysis for college public opinions. Capturing the direction of public opinion in the student community in a timely manner and guiding students to develop the right values can help in the ideological management of universities. Universally, the recurrent neural networks have been the mainstream technology in terms of sentiment analysis. Nevertheless, the existing research works more emphasized semantic characteristics in vertical direction, yet failing to capture sematic characteristics in horizonal direction. In other words, it is supposed to increase more balance into sentiment analysis models. To remedy such gap, this paper presents a novel sentiment analysis method based on multi-scale deep learning for college public opinions. To fit for bidirectional semantic characteristics, a typical sequential neural network with two propagation paths is selected as the backbone. It is then extended with more layers in horizonal direction. Such design is able to balance both model depth and model breadth. At last, some experiments on a real-world social media dataset are conducted for evaluation, well acknowledging efficiency of the proposed analysis model.

Keywords: sentiment analysis; multi-scale deep learning; public opinions; semantic characteristics

1. Introduction

The rapid development of newly emerging techniques such as computer, mobile computing and big data has facilitated the arrival of new media era [1, 2]. As the native of mobile Internet, college students have strong sense in pursuit of free voice, as well as strong willingness in discussion of public topics [3,4]. Nowadays, they have been one of the most active user groups in mobile social media [5,6]. But impacted by increasingly fruitful multiple information channels, the information monopoly of mainstream media has been challenged unprecedentedly [7, 8]. Various negative and false information on the Internet are mixed with massive information, easily bringing emotional fluctuation to college students [9, 10]. Among them, the more typical is the impact of film and television works [11]. At present, the variety of film and television works on the Internet is complex, which has a positive or negative impact on college students' life attitude to a certain extent [12]. This circumstance may even influence mental health of college students and harmony of campuses [13, 14]. Therefore, effective sentiment analysis for campus public opinions in mobile social media [15], is of great importance to ideological management in colleges and universities [16].

During the past few years, the wide application of deep neural networks have been continuously explored in terms of its powerful information processing ability [17], and have also been the mainstream technology for sentiment analysis [18]. The most popular deep sentiment analysis methods are established upon recurrent neural networks (RNN) [19]. The RNN emphasizes the dependency among word-building units of text sequences, and builds a global feature representation on this basis [20]. From the perspective of linguistics, there is often some contextual dependency among the characters of text, which is more consistent with the RNN-based methods in scenarios [21]. Typical models include long short-term memory (LSTM) and its bidirectional version [22]. However, existing methods only considered the vertical features of the text, yet ignoring the potential horizontal features. From the perspective of model structure, they concentrated more on learning depth than learning width to some extent [23, 24].

In order to make up the aforementioned gap, this paper extends the RNN-based models from a single scale to multi-dimensional scales. Keeping the depth stable, the number of concurrency to increase the learning fields is increased [25]. It is expected to get a more comprehensive semantic feature expression to handle the imbalance problem between model depth and model width. Therefore, taking English text based campus public opinion as the object, this paper proposes a novel sentiment analysis method based on multi-scale deep learning for college public opinions (named as MSLSTM-CPO for short). In order to match the bidirectional dependency of semantic sequences, Bi-LSTM is selected as the basic networking unit. Vertically, a single word in English is used as the minimum processing unit. Horizontally, more than two basic networking units are adopted to form a parallel computing structure. As a result, the model has been enhanced with respect to horizonal layer numbers, realizing the trade-off between depth and breadth. We can summarize major working points of this paper as following aspects:

- This work discusses and demonstrates semantic characteristics from both vertical direction and horizonal direction.
- This paper develops, MSLSTM-CPO, a novel sentiment analysis method via multi-scale deep learning for college public opinions.
- This work conducts some experiments to comprehensively assess performance of the proposal.

2. Related works

In recent years, sentiment analysis has received more and more attention as an important research direction in the field of natural language processing, and has gradually become a research hotspot. From the analysis of its research development history, sentiment analysis has mainly gone through three stages: lexicon based method, machine learning method and deep learning method.

In the initial stages of sentiment analysis, lexicon-based statistics on the number of sentiment subscripts was the main method. It is obvious that lexicon-based statistical methods, although easy to understand, do not develop generalisation very well. With the development of natural language processing, machine learning methods research has so far also yielded many effective results [26–28]. Rashmi et al. [29] proposed a soft voting classifier by integrating five baseline models of logistic regression, balanced random forest, eXtreme Gradient Boosting, random forest, and support vector machine with the task goal of classifying mixed Indian languages. The method is useful for classifying positive, negative, neutral, mixed emotions and unemotional states with better results. Maipraditj et al. [30] proposed a machine learning-based approach that uses three different datasets for text processing, using the N-gram IDF method for feature extraction, and then utilizes automated machine learning to classify positive, neutral emotions, and negative emotions. However, traditional machine learning methods have the drawback of not being able to combine semantic information with textual context, so deep learning is gradually becoming a research trend [31–36]. Huang et al. [37] explored changes in emotion in blended learning using LSTM-based text mining methods and epistemic network analysis. Jia et al. [38] constructed a sentiment classification model using BERT, CNN and attention mechanism methods to mine text for contextual connections and features. Harendranath et al. [39] proposed a modeling method based on recurrent neural network to classify the emotions of political comments.

To sum up, almost all the related works dealt with various semantic analysis problems from the perspective of vertical-directional semantics. Nevertheless, semantics modeling from both vertical and horizonal directions, still needs to be deeply discussed and considered. Therefore, the next section of this paper is going to display the proposed technical framework from such point.

3. Methodology

3.1. Overview

Currently, almost all valid natural language models are built on modelling the serialisation of natural utterances. The data for this experiment is English comment text data, and in most cases the semantic information of each comment keyword is closely related to the information of the previous text. Therefore, a neural network that can remember the context can better handle the sentiment analysis of the comment text for this experiment. The traditional LSTM model achieves the ability to remember the text and can handle the problem of internal information complexity and overload caused by long sequential data. In order to match the bidirectional dependency of semantic sequences, Bi-LSTM is chosen as the basic networking unit for this experiment. Hence, a novel sentiment analysis method for college public opinions via multi-scale deep learning is develped in this work. As shown in Figure 1, the whole workflow of the MSLSTM-CPO model consists of two parts: Word-level semantic encoding and Sentence-level semantic encoding.

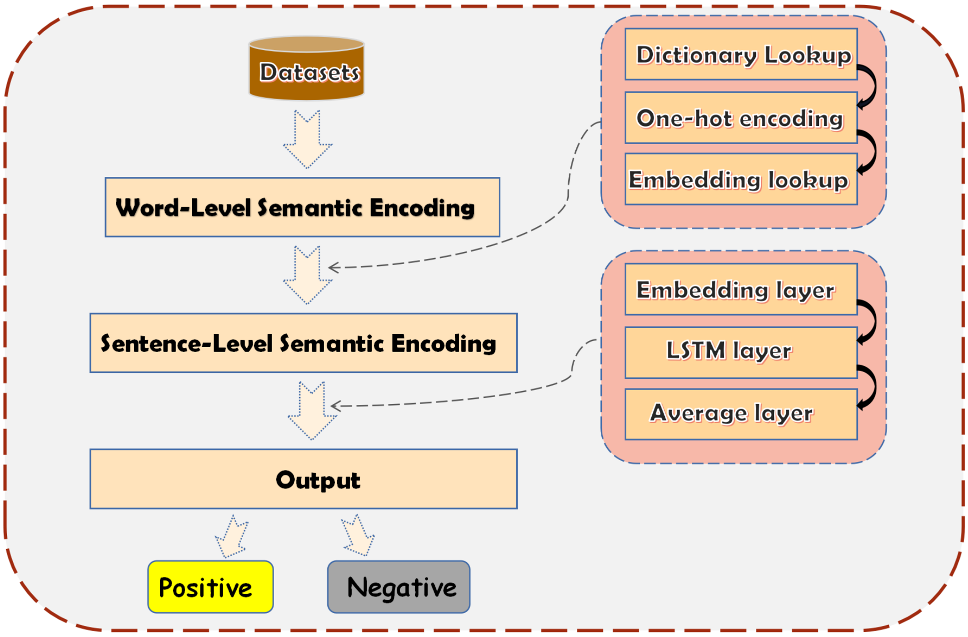


Figure 1. The major workflow of the proposed MSLSTM-CPO.

3.2. Word-level semantic encoding

The data for this experiment is English text data and the classification model cannot be trained directly on the input text. Therefore, each English comment text needs to be separated using space characters and then the words are converted into a vector representation. If there are many words in the corpus, this will result in a very high dimensionality of the vector for each word. As a result, a normal one hot encoding would make the word vectors very sparse. Word embedding is a way of representing words in natural language by representing each word as a vector in a high-dimensional space. In this way, natural language computation is converted into vector computation. The detailed steps of the word embedding are shown in Figure 2.

Step 1: Dictionary lookup. The text is first cut into words using spaces to divide long sentences into a number of words. The words in the text sentence are converted into fixed ID integers by querying the dictionary.

Step 2: One-hot encoding. If the dictionary has P words in its word list, each particular word can be represented by a P-dimensional vector, thus converting each ID into a fixed-length vector. For a word with ID x, the xth element of the vector is 1 and the rest of the $P-1$ elements are 0. This process is One-hot encoding and can be expressed as:

$$v_{x=1} = \left[1, \underbrace{0, 0, \cdots, 0}_{P-1} \right] \tag{3.1}$$

Step 3: Embedding lookup. In a real-life experimental scenario, each text is of different lengths, both long and short. In order to avoid the overall training result of the model being affected by the length of the data being too long or too short, a parameter max_seq_len is set to truncate and complement the text. After One-hot encoding, the sentence tensor is denoted as V. Then, this tensor V is

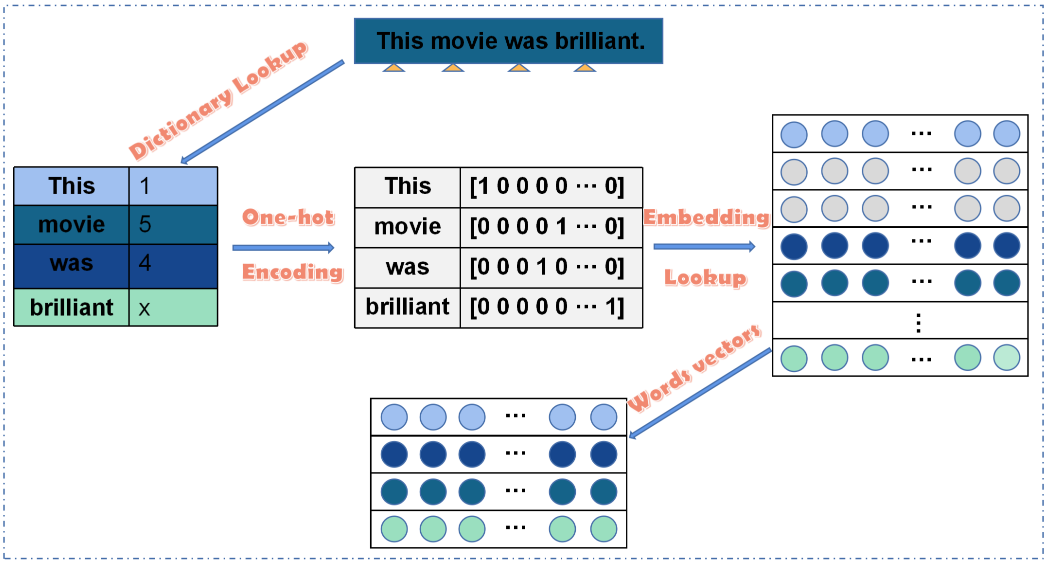


Figure 2. The workflow diagram for the computation of word embedding.

multiplied by another dense tensor W, $W \in \mathbb{R}^{P \times l}$. P denotes the word table size and l denotes the vector size of each word. After tensor multiplication, this can then be mapped to an embedding representation X, thus completing the purpose of representing words as vectors.

3.3. Sentence-level semantic encoding

Based on word-level semantic encoding, this subsection is sentence-level semantic encoding. For each input text sentence, the sentences are truncated and padded by the set global variable max_seq_len, turning them into fixed-length vectors. The processed information is then trained using the model proposed in this paper to obtain the results of sentiment classification. As shown in Figure 3, the MSLSTM-CPO model proposed in this paper has a total of four layers: embedding layer, LSTM layer, average layer and output layer.

3.3.1. Bidirectional LSTM layer

The LSTM model cannot process the raw text data directly, and requires Word-Level Semantic Encoding for Word Embedding, which is a vectorised representation of the input text words. Assuming a set of input sequences as $\mathbf{X} \in \mathbb{R}^{B \times L \times M}$, where B is the batch size, L is the length of the sequence and M is the input feature dimension, the LSTM scans the sequences sequentially from left to right and updates the internal state $\mathbf{C}_t \in \mathbb{R}^{B \times D}$ and the output state $\mathbf{H}_t \in \mathbb{R}^{B \times D}$ of the state at each moment computationally through the loop unit. D denotes the dimensionality of the hidden state vector. The computational steps of the LSTM consist mainly of computing the three gates, computing the internal state and computing the output state.

Step 1: Calculating the three ”doors”. At moment t, the loop unit of the LSTM computes a set of input gates $\mathbf{I}_t$, oblivion gates $\mathbf{F}_t$ and output gates $\mathbf{O}_t$ using the input $\mathbf{X}_t \in \mathbb{R}^{B \times M}$ at the current moment and the output state $\mathbf{H}_{t-1} \in \mathbb{R}^{B \times D}$ at the previous moment. This experiment uses the paddle framework to build the model, which differs from the conventional LSTM model implemented by itself, with two

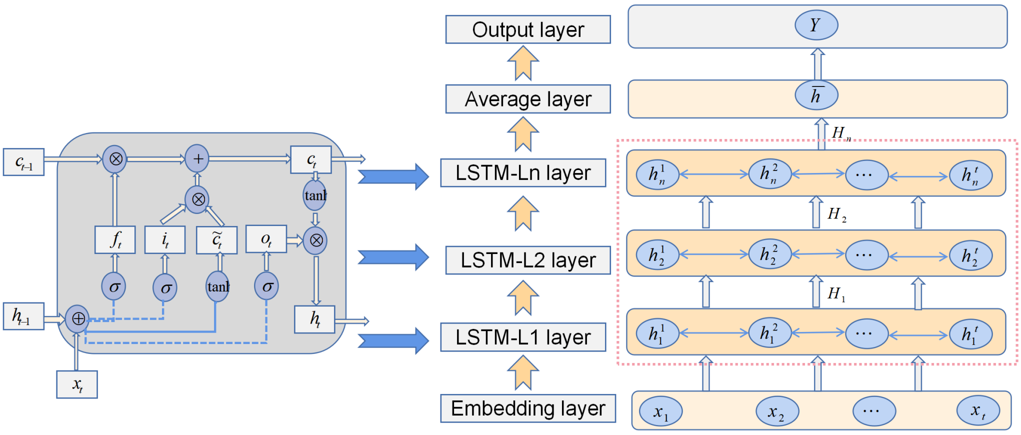


Figure 3. Schematic diagram of the MSLSTM-CPO network structure.

additional biases and the parameters in front of the input data when the matrix is multiplied. The calculation formulas are as follows:

$$\mathbf{I}_t = \sigma\left(\mathbf{W}_{ii}\mathbf{X}_t + \mathbf{b}_{ii} + \mathbf{U}_{hi}\mathbf{H}_{t-1} + \mathbf{b}_{hi}\right) \tag{3.2}$$

$$\mathbf{F}_t = \sigma\left(\mathbf{W}_{if}\mathbf{X}_t + \mathbf{b}_{if} + \mathbf{U}_{hf}\mathbf{H}_{t-1} + \mathbf{b}_{hf}\right) \tag{3.3}$$

$$\mathbf{O}_t = \sigma\left(\mathbf{W}_{io}\mathbf{X}_t + \mathbf{b}_{io} + \mathbf{U}_{ho}\mathbf{H}_{t-1} + \mathbf{b}_{ho}\right) \tag{3.4}$$

where $\mathbf{W}_* \in \mathbb{R}^{M\times D}$, $\mathbf{U}_* \in \mathbb{R}^{D\times D}$, $\mathbf{b}_{i*} \in \mathbb{R}^{1\times D}$, $\mathbf{b}_{h*} \in \mathbb{R}^{1\times D}$ are learnable parameters and σ is a logistic function that controls the values of the ”gates” in the (0,1) interval. The ”gates” here are all matrices of B samples, each row being a vector of ”gates” of one sample.

Step 2: Calculating internal states. The first step is to calculate the internal state of the candidate with the following equation:

$$\widetilde{\mathbf{C}}_t = \tanh\left(\mathbf{W}_{ic}\mathbf{X}_t + \mathbf{b}_{ic} + \mathbf{U}_{hc}\mathbf{H}_{t-1} + \mathbf{b}_{hc}\right) \tag{3.5}$$

where $\mathbf{W}_{ic} \in \mathbb{R}^{M\times D}, \mathbf{U}_{hc} \in \mathbb{R}^{D\times D}, \mathbf{b}_{ic} \in \mathbb{R}^{1\times D}, \mathbf{b}_{hc} \in \mathbb{R}^{1\times D}$ are learnable parameters. Next, the internal state at moment t is then calculated using the use of forgetting gates and input gates, using the following equation:

$$\mathbf{C}_t = \mathbf{F}_t \otimes \mathbf{C}_{t-1} + \mathbf{I}_t \otimes \widetilde{\mathbf{C}}_t \tag{3.6}$$

where $\otimes$ denotes the element-by-element product.

Step 3: Calculating the output state. The current cell state of the LSTM can be calculated according to Eq (3.6) and as follows:

$$\mathbf{H}_t = \mathbf{O}_t \otimes \tanh\left(\mathbf{C}_t\right) \tag{3.7}$$

The input of the LSTM cyclic cell structure is the internal state vector $\mathbf{C}_{t-1} \in \mathbb{R}^{B\times D}$ and the hidden state vector $\mathbf{H}_{t-1} \in \mathbb{R}^{B\times D}$ at moment $t-1$, and the output is the state vector $\mathbf{C}_t \in \mathbb{R}^{B\times D}$ and the hidden state vector $\mathbf{H}_t \in \mathbb{R}^{B\times D}$ at the current moment t. With the LSTM cyclic cell, the whole network can establish longer distance temporal dependencies, thus solving the difficulty when the sequence data carried by the RNN is too long to be handled. In addition, the LSTM can help the model capture the semantic information of long sentences more fully by selectively ignoring or reinforcing the current memory and input information.

The Paddle framework's built-in LSTM model has several parameters, including direction and num_layers. Direction indicates the direction of the network iteration and can be set to forward or bidirectional, with the default being forward. Num_layers indicates the number of layers in the network and defaults to 1. The multi-layer bidirectional LSTM needs to receive a sequence of vectors to update the cyclic units with forward and reverse respectively. Therefore, it is only necessary to set the parameter direction to bidirectional and num_layers to any desired n when defining the LSTM to use the multi-layer bidirectional LSTM directly.

3.3.2. Output layer

The average layer is calculated by averaging the hidden states at all positions of the bidirectional LSTM layer and then used as a representation of the whole sentence. In the experiments, the Average-Pooling operator is implemented for the aggregation of hidden states. First, the sequence length vector is used to generate a mask matrix, which is used to mask the vectors that fill the placeholder positions in the text sequence. The vectors of the sequence are then summed and averaged.

The final output layer, by using Linear to output the results of the classification. Sentiment analysis is essentially a classification problem and in the practice of classification problems only the logarithmic odds of classification are usually required to be output by the model. First, the Linear layer transforms the last moment of the hidden state vector $\mathbf{H}_L \in \mathbb{R}^{B\times D}$ linearly and then outputs the logits of the classification. The formula is as follows:

$$\mathbf{Y} = \mathbf{H}_L\mathbf{W}_{out} + \mathbf{b}_{out} \tag{3.8}$$

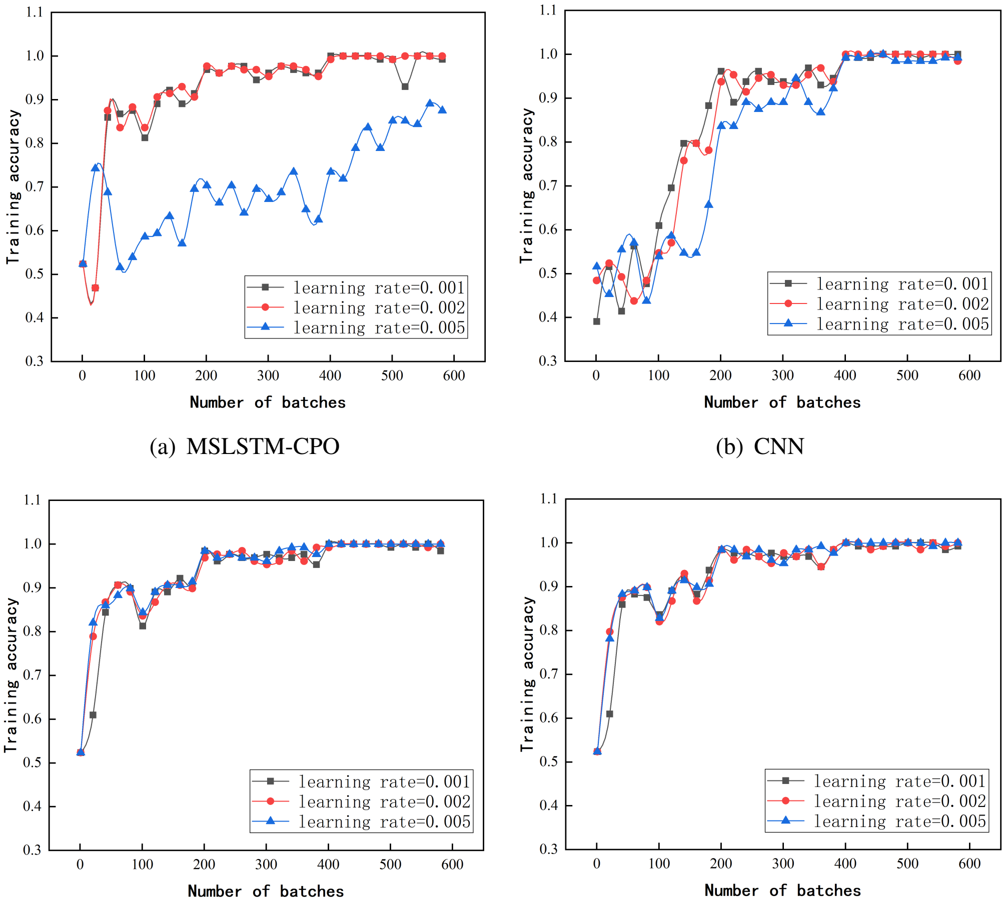
where $\mathbf{W}_{out} \in \mathbb{R}^{D\times N}$ and $\mathbf{b}_{out} \in \mathbb{R}^{N}$ are the learnable weight matrix and bias. N indicates the number of classifications.

4. Experiments and analysis

4.1. Dataset and setting

As far as we know, there is no datasets about college public opinions that are publicly available. This work selects a standard dataset IMDB [40] in area of sentiment analysis for evaluation. It was a kind of dataset that records short reviews in social media and sentiment information is also associated. The data of IMDB comes from the Internet Movie Database, including the user's comment text and rating information for a movie. In the current Internet era, university students are an inescapable part of the film industry's consumer base. It is easy to see from everyday entertainment life that the majority of the film industry's loyal audience comes from university students who have free time. Movie reviews can be used to determine students' emotions and capture the direction of public opinion in a timely manner, thus guiding students to develop the right values.

The IMDB dataset collected the review information of many films, with a total of 50,000 English review texts. The film viewer can give a score when commenting on the film, ranging from 1–10 points. In the data processing, if the rating is below 5, it is judged as a negative review and the label is 0. If the rating is above 6, it is judged as a positive review and the label is 1. The purpose of this experiment



(a) MSLSTM-CPO

(b) CNN

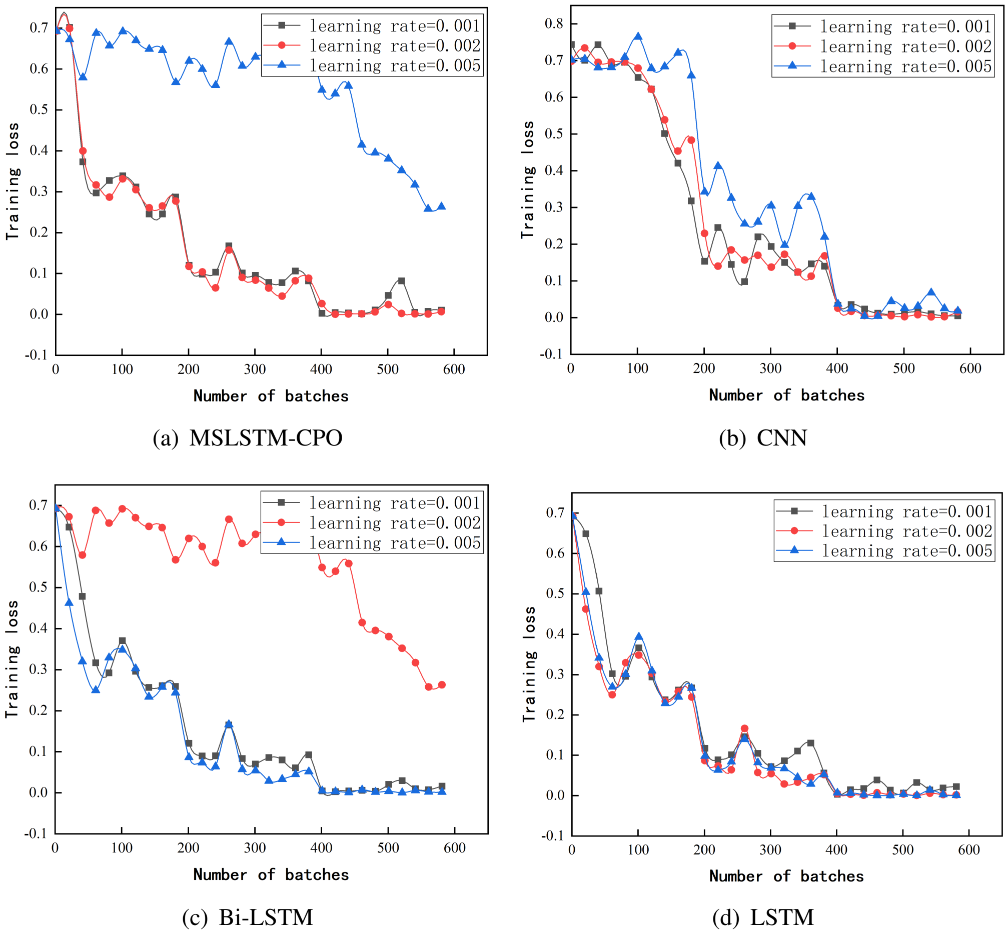
(c) Bi-LSTM

(d) LSTM

Figure 4. Tendency of four methods with respect to training accracy.

is to determine whether the emotion expressed by users is positive or negative according to the text information of the comments. After 50,000 pieces of data are processed, each sample data includes the user's comment text and 0/1 label for a movie. There are 25,000 training data and 25,000 test data. In the training set and test set, there are 12,500 positive and negative samples respectively.

In the experiments, the MSLSTM-CPO model proposed in this paper is compared with both machine learning methods and deep learning methods, respectively. The machine learning models discussed are SVC, GaussianNB, MultinomialNB, BernoulliNB. The deep learning models are mainly discussed in the MSLSTM-CPO proposed in this paper and other baseline methods: CNN model, LSTM model, and bidirectional LSTM model. The text of the comments in this experiment varied in length, so the max-seq-len parameter was set to 256 to truncate and complement the text. For the other parameters in the model training, batch-size was set to 128, hidden layer size was set to 256 and embedding size was set to 256. Considering the time consumption of model training and test accuracy, epoch was set to 3. The learning rate was set to 0.001, 0.002 and 0.005 respectively. The num_layers of the MSLSTM-CPO model was set to 3. According to the composition of the initial dataset, the split ratio between the training and test sets was 50%. In order to fully test the validity of the experiments, four common evaluation metrics of classification models were used: accuracy, precision, recall and



(a) MSLSTM-CPO

(b) CNN

(c) Bi-LSTM

(d) LSTM

Figure 5. Tendency of four methods with respect to training loss.

F1-score. Their detailed descriptions can be found from references like [2,41] and are left out here.

4.2. Results and discussion

4.2.1. Machine learning methods results and discussion

Sentiment analysis is essentially a binary classification problem, so one SVC model and three typical Naive Bayes-based methods were chosen as machine learning comparison models for this experiment. They are named as Gaussian Naive Bayes, Multinomial Naive Bayes, and Bernoulli Naive Bayes, separately. Among them, GaussianNB is Naive Bayes with Gaussian distribution a priori, MultinomialNB is Naive Bayes with polynomial distribution a priori, and BernoulliNB is Naive Bayes with Bernoulli distribution a priori. Machine learning-based sentiment classification models generally have only simple steps: data pre-processing, text vectorisation, and training the classifier.

As shown in Table 1, this experiment computes the four metric values for the five models. The learning rate of the MSLSTM-CPO model is 0.002, and the other parameters are kept constant. From the results, it can be seen that the machine learning method can generally only achieve an accuracy of 0.5 for classification, while the MSLSTM-CPO model can achieve as high as 0.86, which is much higher than the classification effectiveness of the machine learning methods. A visualisation of the

Table 1. Experimental results of machine learning models.

Model	Accuracy	F1-score	Recall	Precision
SVC	0.5206	0.4889	0.5275	0.5206
GaussianNB	0.5057	0.4262	0.5075	0.5128
MultinomialNB	0.4936	0.4935	0.4936	0.4936
BernoulliNB	0.5	0.3333	0.5	0.25
MSLSTM-CPO	**0.8678**	**0.8676**	**0.86755**	**0.86905**

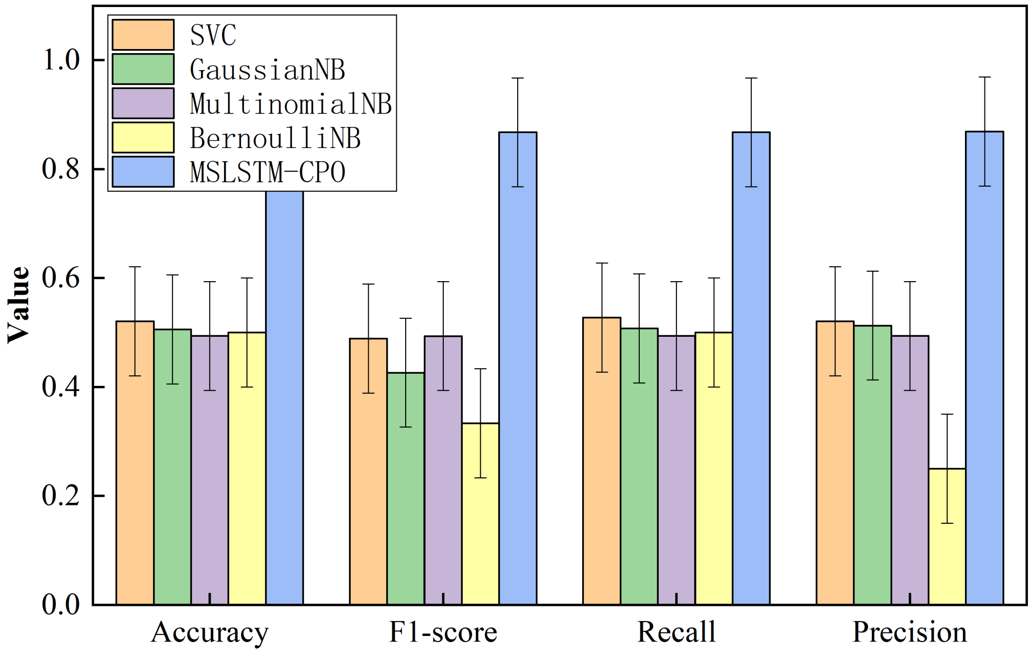


Figure 6. The evaluation results of MSLSTM-CPO with four machine learning models.

evaluation results of MSLSTM-CPO compared to four traditional machine learning models is shown in Figure 6. Its horizontal coordinates indicate the four metric types and the vertical coordinates indicate the magnitude of the values. It is clear from this that the MSLSTM-CPO model, indicated by the blue bars, performs approximately 50% better than the other machine learning models on all four evaluation metrics. Thus, it is again verified that deep learning models are generally able to perform better than traditional machine learning models in terms of sentiment analysis problems.

4.2.2. Deep learning methods results and discussion

In order to discuss the effectiveness of the MSLSTM-CPO proposed in this paper, three other typical deep learning methods are chosen as comparison models: CNN model, LSTM model, and bidirectional LSTM model.

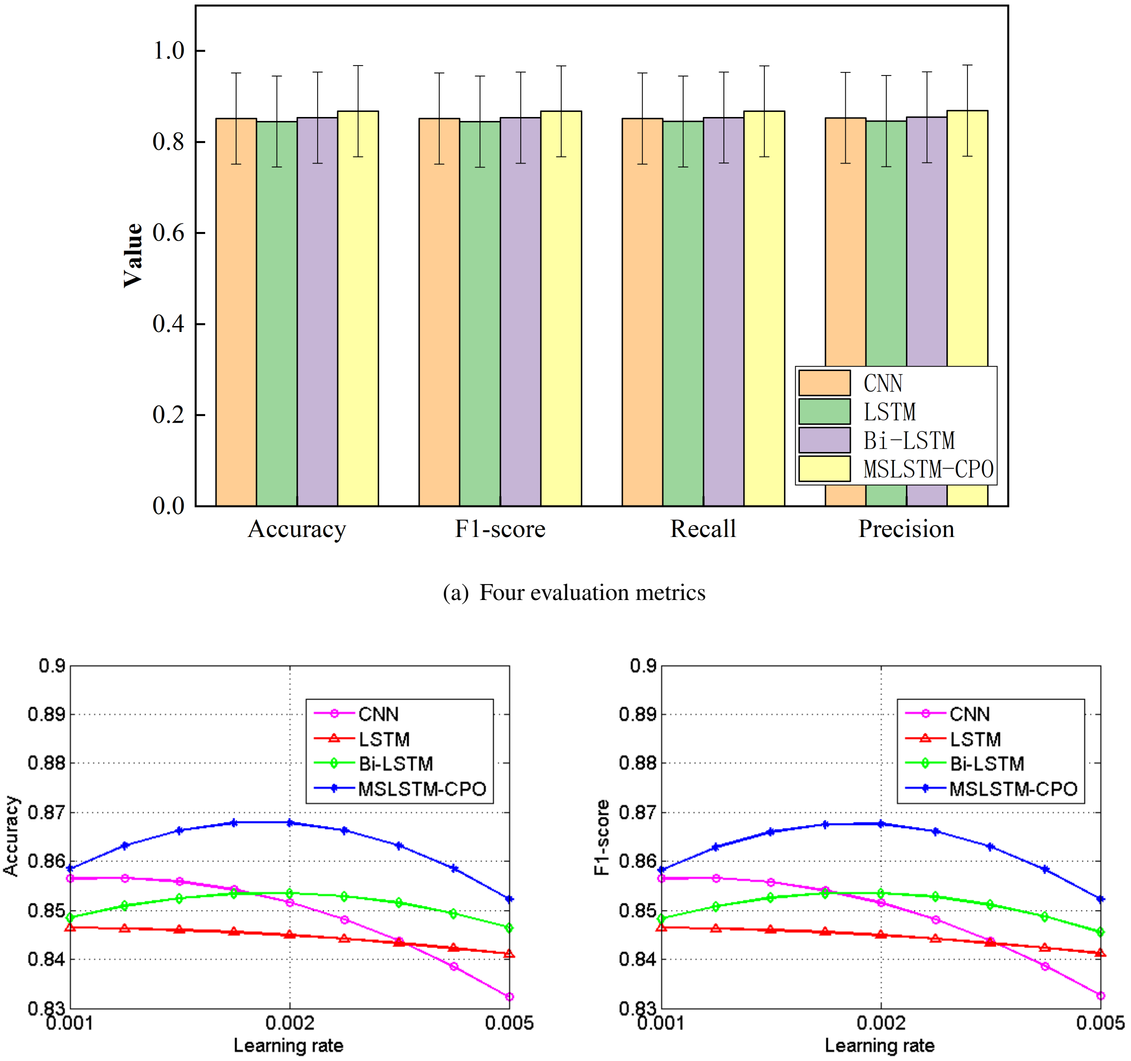
As shown in Figures 4 and 5, the trends of accuracy and loss in the training phase of the models are indicated for the four methods when the learning rates are set to 0.001, 0.002, 0.003 respectively. For each subplot, the x-axis represents the number of iterative rounds ranging from 1 to 600. The y-axis in Figure 4 represents the value of the accuracy of the model training, and the y-axis in Figure 5 represents

Table 2. Experimental results of deep learning models.

Learning rate	Model	Accuracy	F1-score	Recall	Precision
0.001	CNN	0.8565	0.8565	0.8565	0.8566
	LSTM	0.8465	0.84645	0.8465	0.84675
	Bi-LSTM	0.8485	0.8483	0.8487	0.85005
	MSLSTM-CPO	0.8585	0.85815	0.85555	0.85885
0.002	CNN	0.8516	0.8515	0.85165	0.85305
	LSTM	0.8449	0.84485	0.8452	0.84605
	Bi-LSTM	0.8535	0.85345	0.8537	0.8545
	MSLSTM-CPO	**0.8678**	**0.8676**	**0.86755**	**0.86905**
0.005	CNN	0.83225	0.83255	0.83225	0.8322
	LSTM	0.8411	0.8412	0.8412	0.8411
	Bi-LSTM	0.84645	0.84545	0.8471	0.85485
	MSLSTM-CPO	0.85225	0.8522	0.85235	0.8529

the value of the loss of the model training. Each subplot is plotted from 30 consecutive sample points of data, representing the training trend of the same model at different learning rates. It is clear from this that the red curve with the learning rate set to 0.002 shows better results in the training phase of the four models, with the MSLSTM-CPO model showing the most intuitive comparison of trends. As the number of iteration rounds increases, the training accuracy of the four models shows an increasing trend and the training loss shows a decreasing trend, eventually tending to stabilise. This demonstrates that the models are working properly and reasonably well.

The above trends only represent the normal training of the model, but only a comparative analysis with each evaluation index can reflect the comprehensive expression ability of the model. Therefore, in this experiment, the learning rate was used as the variable to calculate the evaluation index values of the four models MSLSTM-CPO, CNN, LSTM and Bi-LSTM respectively, while the other parameters were set unchanged. As shown in Table 2, the learning rates were set to 0.001, 0.002 and 0.005, respectively. Sentiment analysis is essentially a dichotomous problem, and the commonly used evaluation metrics are accuracy, precision, recall and F1-score. From the cross-sectional data of the table, the evaluation metrics of each model are stable under different parameter settings. From the analysis of the longitudinal data in the table, the MSLSTM-CPO model works best when the learning rate is 0.002, with an ACC of 0.8678, which is 2, 3 and 1.7% higher than CNN, LSTM and Bi-LSTM respectively. The evaluation metrics of the MSLSTM-CPO model were the highest and stable for different learning rate settings. As shown in Figure 7, the representation is a visualisation of the evaluation results of MSLSTM-CPO compared to the other three deep learning models. It has three subfigures that demonstrate such visualization effect from two angles. For Figure 7(a), it demonstrates comparison of four metric values between MSLSTM-CPO and other three methods. For Figure 7(b),(c), they select two typical metrics (Accuracy and F1-score) and display the evaluation results of the four models under three learning rate values (0.001, 0.002 and 0.005). This figure can clearly reflect the fact that the proposed MSLSTM-CPO has better performance results compared with baseline methods.



(a) Four evaluation metrics

(b) Accuracy

(c) F1-score

Figure 7. The evaluation results of MSLSTM-CPO with three deep learning models.

4.2.3. Discussion

Currently, sentiment analysis tasks are generally well developed in both machine learning and deep learning approaches. This experimental design compares the MSLSTM-CPO model proposed in this paper with traditional machine learning models and deep learning models respectively. Based on the data in Tables 1 and 2, it can be concluded that the comprehensive performance of the MSLSTM-CPO model has some validity and reliability compared to these models.

The reason why the MSLSTM-CPO model proposed in this paper can achieve better performance can be explained from the perspectives of both machine learning models and deep learning models respectively. First, the focus of the sentiment analysis task is on the exploitation of textual content. Traditional machine learning methods focus on feature extraction from a large amount of labelled data, which leads to classification results. This approach neglects the contextual coherence of the text content and leads to poor model training results. Secondly, RNN and LSTM models with memory capability are commonly used in sentiment analysis tasks. However, only LSTM can adapt to data with long sequences for memorisation. Therefore, the MSLSTM-CPO model proposed in this paper is based on the LSTM. While a normal LSTM model construction is unidirectional and has only one layer, MSLSTM-CPO is a bidirectional and multilayer model. As a result, the model is enhanced in terms of the number of layers, achieving a certain degree of balance between depth and breadth.

In summary, the MSLSTM-CPO model proposed in this paper can effectively implement a new type of college opinion analysis. And compared with machine learning and deep learning baseline methods, MSLSTM-CPO has better classification performance.

5. Conclusions

This paper successfully proposes a MSLSTM-CPO model based on multi-scale deep learning for sentiment analysis of university students. Firstly, the method performs a word embedding operation on the text. Secondly, Bi-LSTM is chosen as the basic networking unit for layer superposition, and the sentiment classification results are obtained by embedding layer, LSTM layer, average layer and linear output layer. The limitations of traditional methods are balanced by the combination of depth and breadth of the model. Experiments on real datasets show that the MSLSTM-CPO model exhibits better performance than traditional machine learning models and commonly used deep learning models.

In future work, it is expected that the user's image and audio information can be mined and fused with textual information in the future to better improve the efficiency and accuracy of sentiment analysis of campus opinion. In addition, this study will further explore more machine learning and deep learning methods in future work. Their performance characteristics in real-world applications will be analysed in order to investigate sentiment analysis tasks based on deep learning frameworks more effectively. This will help to promote a positive attitude towards life and the development of mental health among university students.

Acknowledgement

This work was funded by the Researchers Supporting Project number (RSPD2023R681) King Saud University, Riyadh, Saudi Arabia, and also funded by Humanities and Social Sciences Project of Chongqing Municipal Education Commission (22SKSZ047).

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. E. K. W. Leow, B. P. Nguyen, M. C. H. Chua, Robo-advisor using genetic algorithm and BERT sentiments from tweets for hybrid portfolio optimisation, *Expert Syst. Appl.*, **179** (2021), 115060. https://doi.org/10.1016/j.eswa.2021.115060
2. Z. Guo, K. Yu, A. Jolfaei, F. Ding, N. Zhang, Fuz-spam: Label smoothing-based fuzzy detection of spammers in internet of things, *IEEE Trans. Fuzzy Syst.*, **30** (2022), 4543–4554. https://doi.org/10.1109/TFUZZ.2021.3130311

3. L. Zhao, Z. Bi, A. Hawbani, K. Yu, Y. Zhang, M. Guizani, Elite: An intelligent digital twin-based hierarchical routing scheme for softwarized vehicular networks, *IEEE Trans. Mob. Comput.*, 2022. https://doi.org/10.1109/TMC.2022.3179254

4. F. Ding, B. Fan, Z. Shen, K. Yu, G. Srivastava, K. Dev, et al., Securing facial bioinformation by eliminating adversarial perturbations, *IEEE Trans. Ind. Inf.*, 2022. https://doi.org/10.1109/TII.2022.3201572

5. M. Dragoni, I. Donadello, E. Cambria, Ontosenticnet 2: Enhancing reasoning within sentiment analysis, *IEEE Intell. Syst.*, **37** (2022), 103–110. https://doi.org/10.1109/MIS.2021.3093659

6. S. Zhang, H. Gu, K. Chi, L. Huang, K. Yu, S. Mumtaz, Drl-based partial offloading for maximizing sum computation rate of wireless powered mobile edge computing network, *IEEE Trans. Wireless Commun.*, 2022. https://doi.org/10.1109/TWC.2022.3188302

7. Z. Cai, Z. He, X. Guan, Y. Li, Collective data-sanitization for preventing sensitive information inference attacks in social networks, *IEEE Trans. Dependable Secure Comput.*, **15** (2018), 577–590. https://doi.org/10.1109/TDSC.2016.2613521

8. F. Ding, Z. Shen, G. Zhu, S. Kwong, Y. Zhou, S. Lyu, Exs-gan: Synthesizing anti-forensics images via extra supervised gan, *IEEE Trans. Cybern.*, 2022. https://doi.org/10.1109/TCYB.2022.3210294

9. L. Malandri, C. Porcel, F. Xing, J. Serrano-Guerrero, E. Cambria, Soft computing for recommender systems and sentiment analysis, *Appl. Soft Comput.*, **118** (2022), 108246. https://doi.org/10.1016/j.asoc.2021.108246

10. Y. Li, H. Ma, L. Wang, S. Mao, G. Wang, Optimized content caching and user association for edge computing in densely deployed heterogeneous networks, *IEEE Trans. Mob. Comput.*, **21** (2022), 2130–2142. https://doi.org/10.1109/TMC.2020.3033563

11. F. Ding, Y. Shi, G. Zhu, Y. Q. Shi, Real-time estimation for the parameters of gaussian filtering via deep learning, *J. Real-Time Image Process.*, **17** (2020), 17–27. https://doi.org/10.1007/s11554-019-00907-5

12. Y. Zhu, W. X. Zheng, Observer-based control for cyber-physical systems with periodic dosattacks via a cyclic switching strategy, *IEEE Trans. Autom. Control*, **65** (2019), 3714–3721.

13. X. Zheng, Z. Cai, Privacy-preserved data sharing towards multiple parties in industrial iots, *IEEE J. Sel. Areas Commun.*, **38** (2020), 968–979. https://doi.org/10.1109/JSAC.2020.2980802

14. L. Chen, Y. Zhu, C. K. Ahn, Adaptive neural network-based observer design for switched systems with quantized measurements, *IEEE Trans. Neural Networks Learn. Syst.* , 2021.

15. M. Qorib, T. Oladunni, M. Denis, E. Ososanya, P. Cotae, Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination twitter dataset, *Expert Syst. Appl.*, **212** (2023), 118715. https://doi.org/10.1016/j.eswa.2022.118715

16. F. Ding, Y. Shi, G. Zhu, Y. Q. Shi, Smoothing identification for digital image forensics, *Multimedia Tools Appl.*, **78** (2019), 8225–8245. https://doi.org/10.1007/s11042-018-6807-6

17. S. Liu, S. Huang, S. Wang, K. Muhammad, P. Bellavista, J. D. Ser, Visual tracking in complex scenes: A location fusion mechanism based on the combination of multiple visual cognition flows, *Inf. Fusion*, 2023. https://doi.org/10.1016/j.inffus.2023.02.005

18. A. Keramatfar, H. Amirkhani, A. J. Bidgoly, Modeling tweet dependencies with graph convolutional networks for sentiment analysis, *Cogn. Comput.*, **14** (2022), 2234–224. https://doi.org/10.1007/s12559-021-09986-8

19. Q. Zhang, K. Yu, Z. Guo, S. Garg, J. J. P. C. Rodrigues, M. M. Hassan, et al., Graph neural network-driven traffic forecasting for the connected internet of vehicles, *IEEE Trans. Netw. Sci. Eng.*, **9** (2022), 3015–302. https://doi.org/10.1109/TNSE.2021.3126830

20. Z. Guo, K. Yu, A. K. Bashir, D. Zhang, Y. D. Al-Otaibi, M. Guizani, Deep information fusion-driven poi scheduling for mobile social networks, *IEEE Network*, **36** (2022), 210–216. https://doi.org/10.1109/MNET.102.2100394

21. S. Liu, Y. Li, W. Fu, Human-centered attention-aware networks for action recognition, *Int. J. Intell. Syst.*, **37** (2022), 10968–10987. https://doi.org/10.1002/int.23029

22. S. Liu, P. Gao, Y. Li, W. Fu, W. Ding, Multi-modal fusion network with complementarity and importance for emotion recognition, *Inf. Sci.*, **619** (2023), 679–694. https://doi.org/10.1016/j.ins.2022.11.076

23. Z. Guo, K. Yu, A. Jolfaei, G. Li, F. Ding, A. Beheshti, Mixed graph neural network-based fake news detection for sustainable vehicular social networks, *IEEE Trans. Intell. Transp. Syst.*, 2022. https://doi.org/10.1109/TITS.2022.3185013

24. D. Peng, D. He, Y. Li, Z. Wang, Integrating terrestrial and satellite multibeam systems toward 6g: Techniques and challenges for interference mitigation, *IEEE Wireless Commun.*, **29** (2022), 24–31. https://doi.org/10.1109/MWC.002.00293

25. Z. Guo, K. Yu, Z. Lv, K. K. R. Choo, P. Shi, J. J. P. C. Rodrigues, Deep federated learning enhanced secure poi microservices for cyber-physical systems, *IEEE Wireless Commun.*, **29** (2022), 22–29. https://doi.org/10.1109/MWC.002.2100272

26. B. P. Nguyen, W. L. Tay, C. K. Chui, Robust biometric recognition from palm depth images for gloved hands, *IEEE Trans. Hum.-Mach. Syst.*, **45** (2015), 799–804. https://doi.org/10.1109/THMS.2015.2453203

27. A. X. Wang, S. S. Chukova, B. P. Nguyen, Implementation and analysis of centroid displacement-based k-nearest neighbors, in *Advanced Data Mining and Applications: 18th International Conference*, (2022), 431–443. https://doi.org/10.1007/978-3-031-22064-7_31

28. A. X. Wang, S. S. Chukova, B. P. Nguyen, Ensemble k-nearest neighbors based on centroid displacement, *Inf. Sci.*, 2023.

29. K. B. Rashmi, H. S. Guruprasad, B. R. Shambhavi, Sentiment classification on bilingual code-mixed texts for dravidian languages using machine learning methods, in *Working Notes of FIRE 2021–Forum for Information Retrieval Evaluation*, (2021), 899–907. http://ceur-ws.org/Vol-3159/T6-3.pdf

30. R. Maipradit, H. Hata, K. Matsumoto, Sentiment classification using n-gram inverse document frequency and automated machine learning, *IEEE Softw.*, **36** (2019), 65–70. https://doi.org/10.1109/MS.2019.2919573

31. Y. Li, N. Li, Sentiment analysis of weibo comments based on graph neural network, *IEEE Access*, **10** (2022), 23497–23510. https://doi.org/10.1109/ACCESS.2022.3154107

32. S. O. Alhumoud, A. A. A. Wazrah, Arabic sentiment analysis using recurrent neural networks: A review, *Artif. Intell. Rev.*, **55** (2022), 707–748. https://doi.org/10.1007/s10462-021-09989-9

33. Z. Jin, M. Tao, X. Zhao, Y. Hu, Social media sentiment analysis based on dependency graph and co-occurrence graph, *Cogn. Comput.*, **14** (2022), 1039–1054. https://doi.org/10.1007/s12559-022-10004-8

34. J. He, H. Hu, Language reinforced superposition multimodal fusion for sentiment analysis, *IEEE Signal Process. Lett.*, **29** (2022), 1347–1351. https://doi.org/10.1109/LSP.2022.3180668

35. C. Q. Huang, F. Jiang, Q. H. Huang, X. Z. Wang, Z. M. Han, W. Y. Huang, Dual-graph attention convolution network for 3-d point cloud classification, *IEEE Trans. Neural Networks Learn. Syst.*, 2022.

36. Z. Lin, H. Wang, S. Li, Pavement anomaly detection based on transformer and self-supervised learning, *Autom. Constr.*, **143** (2022), 104544.

37. C. Huang, Z. Han, M. Li, X. Wang, W. Zhao, Sentiment evolution with interaction levels in blended learning environments: Using learning analytics and epistemic network analysis, *Australas. J. Educ. Technol.*, **37** (2021), 81–95.

38. K. Jia, Sentiment classification of microblog: A framework based on BERT and CNN with attention mechanism, *Comput. Electr. Eng.*, **101** (2022), 108032. https://doi.org/10.1016/j.compeleceng.2022.108032

39. V. Harendranath, S. Rodda, Penguin rider optimization algorithm-based deep recurrent neural network for sentiment classification of political twitter data, *Int. J. Web Serv. Res.*, **19** (2022), 1–25. https://doi.org/10.4018/IJWSR.299019

40. A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, C. Potts, Learning word vectors for sentiment analysis, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (2011), 142–150.

41. R. Das, T. D. Singh, A multi-stage multimodal framework for sentiment analysis of assamese in low resource setting, *Expert Syst. Appl.*, **204** (2022), 117575. https://doi.org/10.1016/j.eswa.2022.117575