IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2021, accepted February 11, 2021, date of publication February 23, 2021, date of current version March 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3061450

Sentiment Classification Algorithm Based on Multi-Modal Social Media Text Information

MINZHENG XUANYUAN, LE XIAO, AND MENGSHI DUAN

Henan University of Technology, Zhengzhou 450052, China
Key Laboratory of Grain Information Processing and Control, Ministry of Education, Zhengzhou 450001, China

Corresponding author: Le Xiao (xiaole@haut.edu.cn)

This work was supported in part by the Natural Science Foundation of the Henan Province under Grant 152102210068, and in part by the National Key Research and Development Program of China, under Grant 2017YFD0401001.

ABSTRACT The issue of sentiment classification in short-term and small-scale data scenarios is considered in this paper. It is a hot topic because the text sentiment classification task in the public opinion analysis scene has two characteristics: short time and small data scale. Existing work focused on improving the accuracy at the cost of data and training time, without considering scenarios where time and data are lacked. The most commonly used method to solve the problem of small data scale is to use multi-modal information such as pictures, sounds and videos, which will lead to unbearable training time. The shorter training time determines that the classification model is generally selected as a deep neural network with fewer layers, such as TextCNN, TextRNN, and so on. However, such models are limited by the structure and have a low classification accuracy. In order to solve both short-term and small-scale data problems, a common information user attribute on social media is added to the model as multimodal information, which includes twelve attributes such as user age, location, and posting time. This paper proposed a sentiment classification algorithm based on multi-modal social media text information. The algorithm makes use of parallel convolutional neural networks (CNN) and recurrent neural network (RNN) to process text information and user attributes respectively, and combines the feature vectors of the two models for classification, which is called User attributes Convolutional and Recurrent Neural Network (UCRNN). The addition of user attributes can improve accuracy, and the CNN network used to extract user attributes features has fewer parameters, which proves that the algorithm can achieve high accuracy under short-term and small-scale data. Experiments verify that the training time of this model is slightly less than TextRNN. The classification accuracy can reach 90.2%, which is the state-of-the-art in the field of short-term and small-scale data sentiment classification.

INDEX TERMS UCRNN, sentiment classification, public opinion analysis, natural language processing, deep neural network, social media, multi-modal.

I. INTRODUCTION

The popularization of the Internet has brought the extreme convenience of information exchange. Hot issues can trigger a great quantity discussion on the Internet in a short time. The collection, analysis and response of the public opinion is called public opinion analysis. Its key technology is emotion classification, which is an important subtask in natural language processing. Because of the top trending search mechanism and public discussion feature of social media, the objects of online public opinion analysis are selected from Twitter, Weibo and other social media. Previous studies focused on improving the accuracy at the cost of data and training time, which is difficult to adapt to the short-term and small-scale requirements of social media public opinion analysis tasks.

Text emotion classification technology has developed by leaps and bounds over the years. Since bidirectional encoder representation from transformers (BERT) [1] published in 2018, the state of the art of natural language processing (NLP) each sub-tasks are monopolized by models based on Transformer and pre-training. That is because, on the one hand, the Transformer network has good parallel computing ability and has the advantages such as coding by location are

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif.

very suitable for NLP tasks [2], on the other hand, the two-stage mode of sub-tasks after the pre-training can improve the effect of text processing.

In contrast, convolutional neural networks (CNN) and recurrent neural network (RNN) series networks seem to have reached a bottleneck due to the problems existing in their respective network structures. It is difficult for CNN series network to capture long distance features, and the pooling layer cannot capture word location information. RNN series network is difficult to be used in large-scale parallel computing. Transformer networks solves the various issues of CNN and RNN [3] to achieve such excellent results for various NLP sub-tasks. However, this does not mean that Transformer network is perfect. Taking XLNet [4] as example, the premise of achieving extremely high accuracy is the support of big data, which means the demand for data scale and hardware. Although many scholars have made lightweight improvement work, training related networks still requires high hardware configuration and plenty of time. Hence, RNN and CNN series networks are still the choices with higher priority in lightweight requirements [5]–[9].

The task of emotion classification in the process of public opinion analysis is a lightweight application of natural language processing. When the hot event just happened, there were few relevant discussion data but many curious members of the public. How to use these small-scale data to complete the training of sentiment classifiers in this field in a short time has become the key to the problem.

For such short-term and small-scale sentiment classification tasks, the classification model is generally selected as a deep neural network with fewer layers, including TextCNN [5], CharCNN [6], FastText [7], TextRNN [8], TextRCNN [9], etc. Although these networks do not require numerous training time and large-scale corpus data, their classification effect is not good even after optimization and tuning, due to the limitation of the model structure

In order to improve the effect of sentiment classification under small data scale, a variety of methods have been applied to expand the information contained in the data. One of the simplest ideas is to expand the data size. Sun and He [10] proposed a hybrid neural network model based on data expansion technology. The data expansion technology can improve the data scale, enhance the generalization performance of the model, and then improve the accuracy. However, the technique only artificially expands the data scale, while long-term training on large-scale data is still required during model training.

Another feasible and popular method is to use multi-modal information. Kumar *et al.* [11] used a combination of pictures and text in social media to train a hybrid classifier and achieved higher accuracy than using two types of information alone. Bairavel and Krishnamurthy [12] added video information on this basis, and extracted features from the information of the three modalities to train a neural network with very high accuracy. Using multi-modal information in a small-scale data set can improve the classification effect, whilst brings several problems: First, the acquisition and processing of multiple modal information is more complicated; The second is that multiple modal information requires different feature extraction networks, and the required parameters are very large, which will increase the training time; The third is that most social network data only contains text, without information such as pictures and videos. The multi-modal information classifier does not have excellent classification accuracy in the case of single-modal input.

Abovementioned methods can tackle the problem of small data scale. Using multimodal information to increase input information and adjust the model structure can improve the classification effect. However, these models require numerous training time. In order to reduce the training time, a new type of multi-modal information can be found with the characteristics of widespread existence and fewer feature extraction model parameters. Using new multi-modal information can reduce the training time of the model while ensuring the accuracy of classification. After a lot of research, user attributes in social networks can be used as such multi-modal information. Therefore, a sentiment classification algorithm for social media text data based on user attributes is proposed. In summary, this paper makes the following contributions:

1) The user attributes in social media are added to the sentiment classification model, and thus the input information is easy to obtain.
2) Because the CNN network used to extract user attributes features has fewer parameters, the model training time is short.
3) Multimodal social media text information sentiment classification algorithm is the state-of-the-art in short-term and small-scale data sentiment classification.

This article is organized as follows:

Section II explains user attributes and its influence on user emotions. A model for sentiment classification of short-term and small-scale data is presented in section III. This section covers the structure and formula of User attributes Convolutional and Recurrent Neural Network (UCRNN) model. In section IV experiments and analyses of various models are discussed. Finally, conclusions are presented in section V.

II. USER ATTRIBUTES

There are mainly two types of data in social media, namely tweet data and user attributes including the characteristics of the poster and the time and space of the posting. Researchers have done little research on the relationship between user attributes and text emotions, focusing instead on emotional similarities in tweets posted by users with certain attributes in common. Mairesse *et al.* [13] divided users into five personalities based on attribute characteristics, and added different weights for each personality in different emotions to assist text emotion classification, proving that user attributes can be used to assist text emotion classification.

In addition to the emotional portrait of users, there are also studies on emotional similarity in time and space.

Mitchell *et al.* [14] combined the geo-tagged text set of social media with the corresponding emotional characteristics of the annual survey of more than 400 cities in the United States, proving that people in a region have certain similarities in word usage and emotional tendency. Li *et al.* [15] studied people's collective response to special events in a very short time through social media, and the emotional trend of the public towards an event in a short time is always similar.

All the above researches have proved that there is a direct relationship between user attributes and text emotions, but they do not further apply user attributes directly to text emotions classification. The proposed method constructs a multi-modal classification model using user attributes as auxiliary knowledge to increase the input information of the text classification. It should be noted that it is very convenient to obtain user attributes. Crawler technology can be used to obtain user attributes in natural language on social media.

Based on the research on preoccupant social media and linguistic information, the main user attributes are divided into two categories. The user's characteristics include age, gender, hometown, number of posts, number of follow and number of followers. And the tweet attributes include time, place, tool, number of likes, number of retweets, number of comments. The dimensionality reduction method, such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), may cause information loss. The neural network has the ability to process 12 features, so there is no need to reduce the dimension of user attributes.

III. UCRNN MODEL

What the model needs to accomplish is to predict emotional categories for text with user attributes. There are two types of input data that need to be spliced using two parallel models. For text feature extraction, RNN series network is used to retain word position information. For the feature extraction of discrete user attributes, CNN series network is more appropriate. In the end, the feature vectors of the two models are spliced for classification, and the final weight matrix is trained by full connection without fixed weight value. The whole model is called User attributes Convolutional and Recurrent Neural Network (UCRNN) due to the use of CNN based on user attributes and RNN based on text.

The input data are text data and user attributes, both of which are natural language text. After the text is segmented, it is represented by W_1, W_2 to W_n, and the user attributes of publishing the text are represented by U_1, U_2 to U_m. embedding from language model (ELMo) [16], BERT, XLNet and other models are not suitable for short-time text classification due to the long training time. Text W needs to consider context information, and use the model globalvectors for word representation (GloVe) [17] that can well represent global co-occurrence information as the word representation model of W. Because the user attribute U does not need to consider the full text information, the word2vec [18] model with a smaller window is used for word vector representation.

ELMo and its improved word representation model can take context information into account when training the word representation model, and can distinguish polysemy cases well. But neither GloVe nor word2vec has similar ability. Therefore, the same field text was used in the selection of training data to improve the ability of word vectors to adapt to the corresponding field. GloVe and word2vec trained word representation matrices P and Q respectively, and the dimensions were $S \times N$ and $T \times N$. One-hot representation of text vocabulary W_i is $V(W(i))$, and the dimension is $1 \times S$. The one-hot attribute vocabulary U_i represents the word vector as $V(U(i))$ and the dimension as $1 \times T$. Then the distributed representation X and Z of the two types of data are as follows:

$$X_i = V(W(i)) \times P \tag{1}$$

$$Z_i = V(U(i)) \times Q \tag{2}$$

where the text word vector $\mathrm{X}_i(i = 1, 2 \cdots \mathrm{n})$ as the input of bi-directional long-short term memory (Bi-LSTM), input forward channel and reverse channel. The calculation formula of h_t output by long-short term memory (LSTM) computing unit at time t is as follows:

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \\ i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \\ o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \\ \bar{c}_t = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \\ c_t = f_t \times c_t + i_t \times \bar{c}_t \\ h_t = o_t \times \tanh(c_t) \end{cases} \tag{3}$$

where f_t, i_t and o_t are the output values of forgetting gate, memory gate and output gate at time t, $\overline{c_t}$ and c_t are the temporary cell state and cell state at time t. W_f, W_i, W_o and W_c are forgetting gate, memory gate, output gate and cell state transition matrix, b_f, b_i, b_o and b_c are corresponding linear offsets. $\overrightarrow{h_n}$ and $\overleftarrow{h_n}$ are used to represent the output of the last computing unit of forward and reverse LSTM. The output H of the whole network is derived from the fusion of two vectors:

$$H = (\overrightarrow{h_n}, \overleftarrow{h_n}) \tag{4}$$

Attribute word vector $\mathrm{z}_i(i = 1, 2 \cdots \mathrm{n})$ is the input of CNN network, and the vector is combined to obtain the user attribute matrix Z. $Z_{i:j}$ is used to represent the splicing of all word vectors in Z_i and Z_j as follows:

$$Z_{i:j} = Z_i \oplus Z_{i+1} \oplus \cdots \oplus Z_j \tag{5}$$

where symbol $\oplus$ represents the splicing between vectors. The convolutional layer uses stitching vectors of different window widths to extract feature vectors C_i under various filter widths of Z as follows:

$$C_i = f(w \cdot Z_{i:i+h-1} + b) \tag{6}$$

$$f(x) = \begin{cases} x, & if : x > 0 \\ \lambda x, & if : x < 0 \end{cases} \tag{7}$$

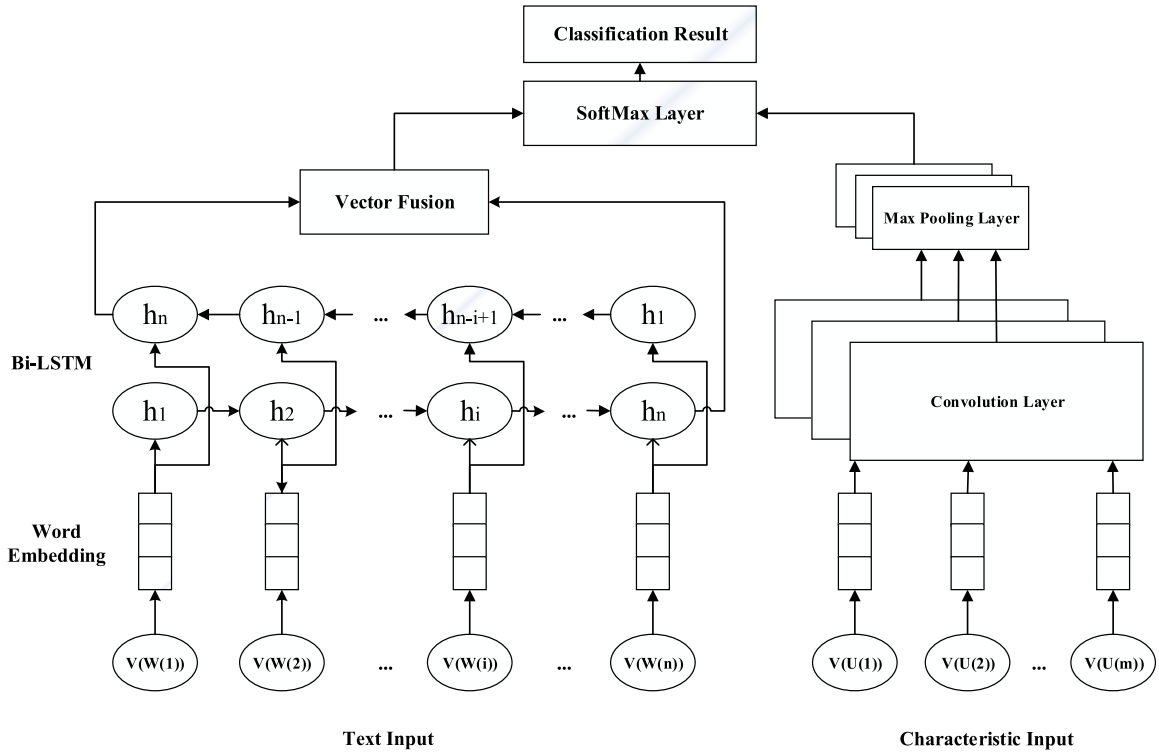

FIGURE 1. UCRNN model structure. The model is composed of a CNN and a BI-LSTM in parallel. The BI-LSTM is used to process the social media text, and the CNN part is used to process the user's attribute characteristics. The features extracted from the two models are fused and used for text emotion classification.

where h is the window width, $\lambda \in (0, 1)$ is the linear offset, b is a bias term and f is the nonlinear activation function Leaky ReLU. Combine different C_i into feature matrix C.

$$C = [C_1, C_2, \cdots, C_{n-h+1}] \tag{8}$$

Pooling of feature matrix C can reduce dimension and extract effective features. The final output N of attribute data obtained from the Max-pooling layer in the CNN module is as follows:

$$N = \max -pooling(C_1, C_2, \cdots, C_{n-h+1}) \tag{9}$$

The output feature vectors of the two networks are H and N, which are fully connected with the neurons of the number of category tags. Perform SoftMax probability calculation on the output of each neuron, and finally get the predicted classification result y as follows:

$$\mathrm{y} = g(W_1 \cdot H + W_2 \cdot N) \tag{10}$$

$$g(i) = \arg\max_{i \in S} \frac{e^i}{\sum_{j=1}^{S} e^j} \tag{11}$$

where W_1 and W_2 are the feature matrices, S is the number of emotion categories, and $g(i)$ indicates that the category with the highest probability is selected using the SoftMax activation function, which is the model prediction category y.

The UCRNN model uses CNN to extract user attribute features and RNN to extract text features. After combining the features, it is used to infer the sentiment category of user tweet data. The total number of model parameters is slightly more than that of the TextRNN model.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify the effectiveness of the algorithm, several commonly used models in short-term and small-data scenarios are used to conduct accuracy comparison experiments with the UCRNN model under the same other conditions. In this part, All simulations are based on Python language, and the configuration of running computer is 32G installed memory, 4 Intel®Cores(TM), i7-7700 3.60 GHZ CPU.

A. EXPERIMENTAL DATA

The most commonly used Chinese social media public data set is MicroblogPCU, which contains 230,000 Weibo content and corresponding user attributes. However, the investigation found that the data set only contains part of user attribute information, which cannot meet the needs of the experiment. In order to obtain all kinds of user attributes of Weibo, crawlers are used to obtain tweet data and user attribute data on the same social media Sina Weibo. After crawling and cleaning, 38,000 pieces of data were obtained. Emotion annotation of data is divided into six experimental data sets according to data scale and number of emotion categories. The data set used in the experiment is not benchmarked, but the acquisition method and acquisition medium are the same

TABLE 1. Experimental data sets.

Data	c	N	V	V_{pre}	Test
DES	2	3000	2555	2417	500
DEM	2	10000	6716	6135	1000
DEL	2	20000	10758	9504	2000
MES	7	3000	2555	2417	500
MEM	7	10000	6716	6135	1000
MEL	7	20000	10758	9504	2000

as the benchmarked data set MicroblogPCU, and the six sub-category data sets are all balanced.

Where c represents the number of emotion categories, divided into Double emotion (DE) and Multi emotion (ME); N represents the data set size, divided into Small(S), Medium(M), Large(L); V represents the word size, V_{pre} represents the pre-training word vector size, and Test represents the amount of data in the test set. Multiple data sets are constructed because it is necessary to prove the classification accuracy of the UCRNN algorithm on various data scales and emotional categories.

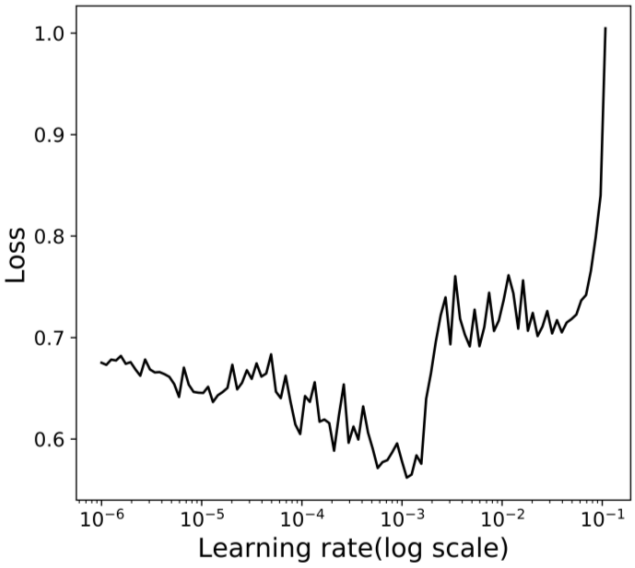

FIGURE 2. Learning rate vs. loss. With the deepening of training, loss continues to decrease in small-scale shocks, and then begins to rise sharply after reaching a certain level. The lowest learning rate is about 0.0012. In order to avoid the loss explosion that may occur when using the lowest loss value, the initial learning rate of the UCRNN model is selected as 0.001.

B. COMPARISON MODEL

In the existing research and applications, there are some commonly used deep neural network models that are effective in sentiment classification tasks in small-scale data and short-term scenarios. Choose the five most popular models with the same training time as UCRNN as the comparison model: TextCNN, CharCNN, FastText, TextRNN, TextRCNN.

· TextCNN[5]: A word-level text classification method based on a one-dimensional convolutional neural network uses convolution kernels of different sizes to extract features of different text lengths. It is the baseline model of the CNN part of the UCRNN model.

· CharCNN[6]: Character-level text classification method based on convolutional neural network. Character-level one-hot encoding of English letters and various commonly used symbols, and reverse encoding to convert each character into a vector. Finally, a convolutional neural network is used to classify character-level text vectors.

· FastText[7]: Character-level text classification method based on word representation and linear model. The model changes the input of the CBoW model to n-gram character vectors, and the output is sentiment classification labels. The hierarchical SoftMax layer is used when the number of categories is large, and the training time is extremely short.

· TextRNN[8]: A word-level text classification method based on Bi-LSTM. The two-layer LSTM extracts the information of the previous text and the following text respectively, and finally combines the output of the two for text classification. It is the baseline model of the RNN part of the UCRNN model.

· TextRCNN[9]: A text classification method based on the serial connection of Bi-LSTM and CNN. The output vectors in the two directions of each layer of Bi-LSTM are fully connected with the word representation vector and then activated and pooled for text classification. It takes a little longer and the accuracy is higher.

· UCRNN: The model proposed in this paper is based on the text classification method of Bi-LSTM and CNN in parallel. LSTM is used to process text, CNN is used to process user attributes, and the output of LSTM is fully connected with the output of CNN and then activated for text classification. Using the features of the two modules, a relatively high accuracy rate can be achieved especially on small-scale data.

C. MODEL TRAINING TRICKS

UCRNN is a new type of deep neural network, and the choice of hyperparameter values will affect the speed and accuracy of the training process. This article uses some tricks to determine a series of training measures and hyper-parameter values, including the selection of parameters such as weight initialization, learning rate, batch-size and dropout.

The choice of learning rate is a very critical step in model training. A learning rate that is too low will cause the model to converge slowly or even fail to converge, and a learning rate that is too high may cause convergence to the wrong position. The traditional learning rate selection method is to try a lot of different learning rates, so that good results can be obtained, but a plenty of time will be wasted. Smith [19] proposed that in the first epoch of a new training process, different mini-batch trainings should be selected with different learning rates that increase from small to large. Record the trend of loss as the learning rate changes, and the appropriate learning rate can be selected. This paper uses this method to find the

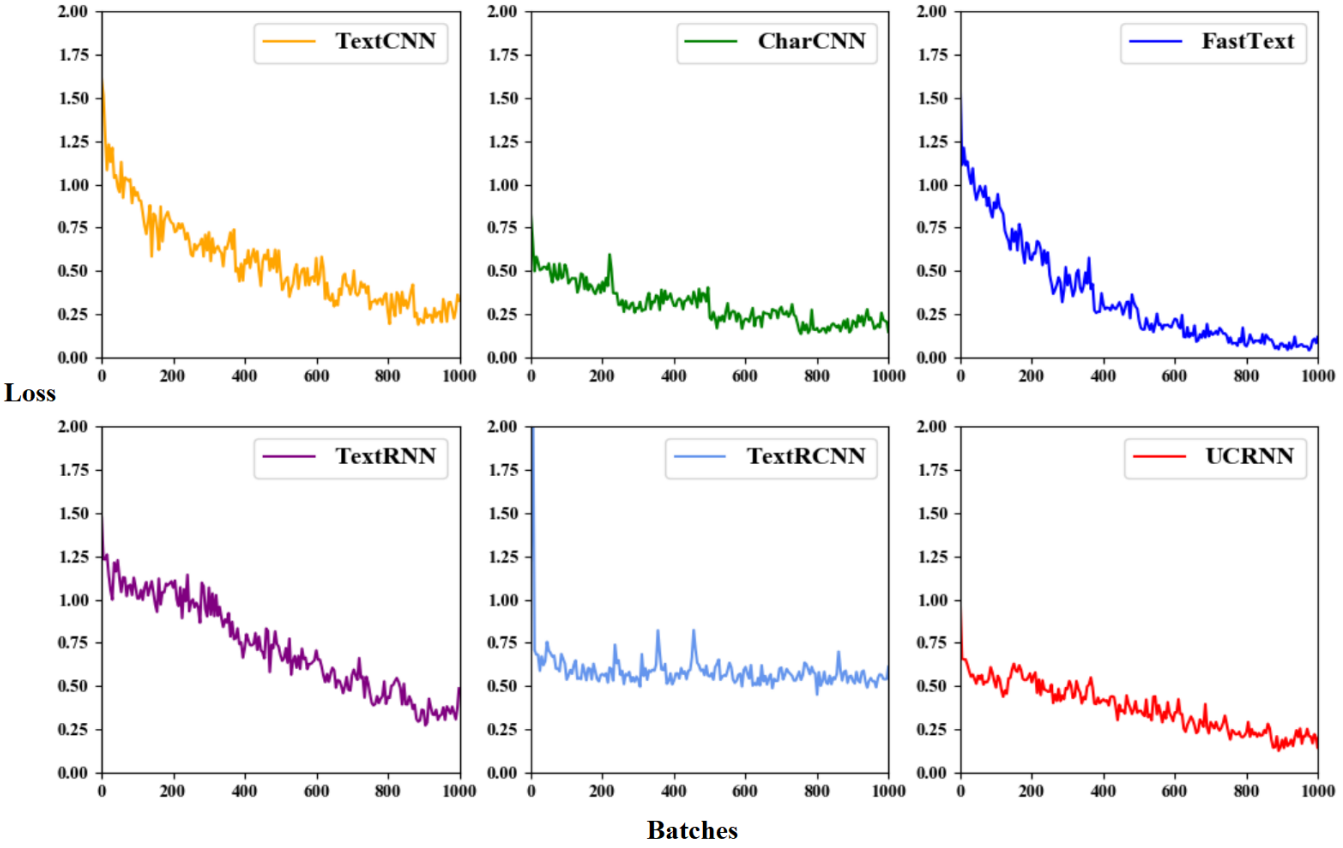

FIGURE 3. Loss function curve. It represents the change process of the loss function values of the six models as the training iteration increases. Each curve decreases with slight fluctuations and gradually tends to converge, which proves that there is no problem in the selection of hyperparameters during the training of each model. The FastText model has the fastest convergence rate, followed by UCRNN, and the remaining models converge slowly. It can be observed that the iteration required for UCRNN convergence is lower than its baseline model TextRNN.

appropriate learning rate of the UCRNN model, and draws a line graph of the trend of loss with the learning rate.

Whether the weight is initialized or not mainly affects the convergence speed of the neural network, and may affect the convergence effect. The UCRNN model is designed to solve the problem of emotion classification in short-term and small data scenarios. Without a large amount of data for pre-training, it is necessary to choose a suitable initialization method to improve the convergence speed. For the case where the activation function is Leaky ReLU, the He initialization method [20] can achieve very good results.

In the choice of batch-size, a large value requires more calculation space, and a small value requires a longer running time. Considering the characteristics of the tasks handled by UCRNN, a larger batch-size is selected after experiments, with a value of 64. UCRNN is a parallel combination of CNN and RNN. The feature fusion part uses full connection, and the weight is difficult to determine. Using a larger probability dropout can quickly find the most suitable weight for combining the two modal features, and finally add a 0.5 probability dropout to the fully connected layer where the two models are combined.

TABLE 2. Selection of the hyperparameters.

Hyperparameters	Options/Values
Activation Function	Leaky ReLU
Weight Initialization	He Method
Loss Function	Cross-entropy
Learning Rate	0.001
Batch Size	64
Epochs	15

D. EXPERIMENTAL ANALYSIS

In order to verify the efficiency and accuracy of the UCRNN model, under the same other conditions, six models were trained using the same data to show the relationship between the iterations and loss of each model, and to ensure that the hyperparameter selection of each model has no problems. The training time, accuracy, precision and recall of the six models are obtained and analyzed.

The accuracy of the UCRNN model on each data set is higher than the comparison model. Especially in the case

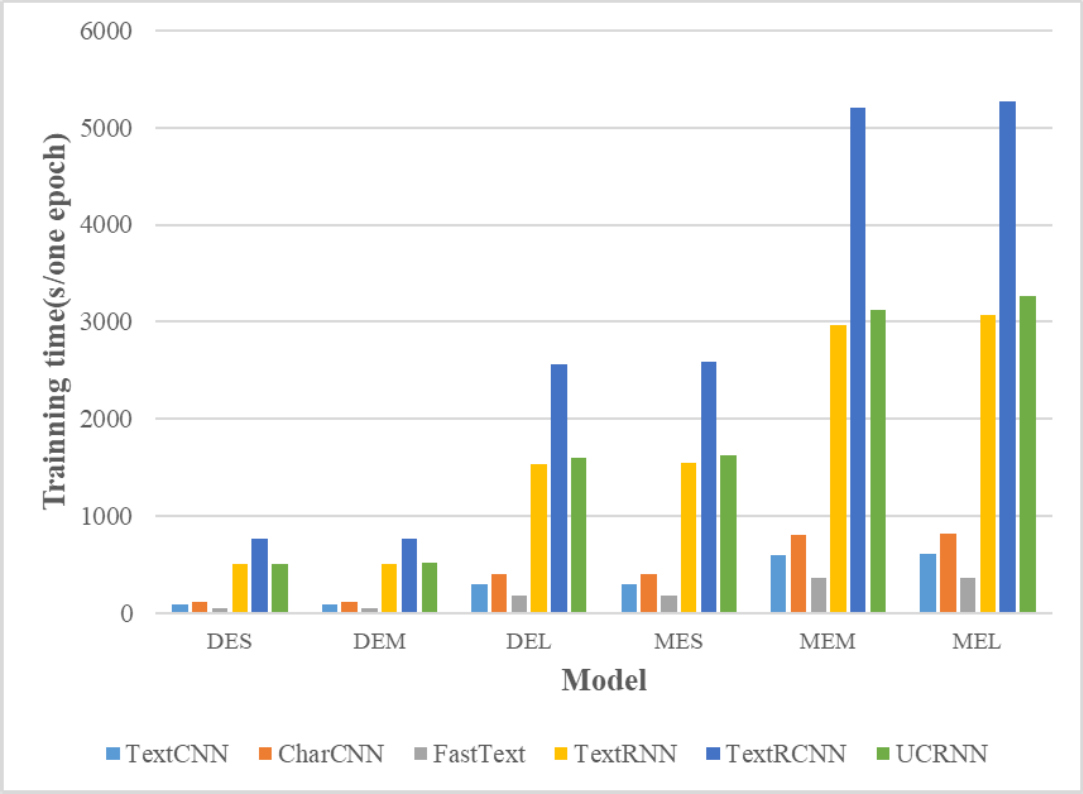

FIGURE 4. Model training time. The bar graph represents the time (s) required for the six models to train one epoch on each of the six data sets. Comparing the training time of each model, FastText is the shortest, TextRCNN is the longest, and the time required for UCRNN is ranked second, close to the third place TextRNN. Since the UCRNN model converges quickly, it requires less epoch than TextRNN, which can prove that UCRNN is faster than TextRNN in overall training time.

TABLE 3. Model accuracy.

Model	DES	DEM	DEL	MES	MEM	MEL
TextCNN	83.7	85.6	86.8	82.3	83.8	85.7
CharCNN	84.1	85.8	87.5	82.8	85.2	86.6
FastText	83.2	85.5	86.3	81.9	83.4	85.3
TextRNN	83.4	84.8	86.5	82.5	84.4	86.4
TextRCNN	84.4	86.1	88.7	83.7	85.9	87.9
UCRNN	**86.7**	**88.9**	**90.2**	**86.0**	**88.3**	**89.5**

TABLE 4. Model precision.

Model	DES	DEM	DEL	MES	MEM	MEL
TextCNN	85.2	86.1	87.2	81.8	83.5	85.5
CharCNN	85.3	86.2	88.0	82.2	84.9	86.4
FastText	84.6	86.0	86.8	81.1	83.2	85.2
TextRNN	84.8	85.0	86.9	81.8	84.2	86.2
TextRCNN	85.6	86.3	89.7	83.4	85.7	87.8
UCRNN	**87.2**	**89.6**	**90.9**	**85.7**	**88.1**	**88.7**

TABLE 5. Model recall.

Model	DES	DEM	DEL	MES	MEM	MEL
TextCNN	81.6	85.0	86.3	43.6	46.3	50.0
CharCNN	82.3	85.3	86.8	44.4	49.0	51.8
FastText	81.1	84.8	85.7	42.9	45.6	49.2
TextRNN	81.4	84.5	86.0	43.9	47.4	51.4
TextRCNN	82.7	85.8	87.5	46.1	50.4	54.8
UCRNN	**86.0**	**88.0**	**89.4**	**50.6**	**55.7**	**58.8**

of small-scale data training, the classification accuracy is significantly improved compared to the baseline model. In the comparison between data sets, the accuracy of binary classification is slightly higher than that of multi-class classification, and the increase in data size will also increase the classification accuracy.

The two tables respectively show the precision and recall of the six models on the six data sets. The precision and recall of UCRNN on each data set are higher than the comparison model. In the comparison within the model, the precision is higher than the recall, which indicates that the model has a higher rate of classifying data sentiment as negative in the binary classification, and it is easier to identify negative sentiment. In the comparison between the data sets, the precision

and recall of binary classification are higher than those of multi-class classification, and the increase in the number of data will also improve the precision and recall.

UCRNN can achieve the highest values in terms of classification indicators such as accuracy, precision, and recall, which proves that using CNN to extract user attribute features to assist Bi-LSTM to extract text features is very effective for user sentiment classification. The training time required for each epoch of UCRNN is slightly higher than that of TextRNN. However, UCRNN requires less epoch to converge than TextRNN, and its training time is lower. Experiments have proved that UCRNN can achieve state-of-the-art for short-term and small-scale emotion classification.

V. CONCLUSION

In order to ensure the short-term and small-scale data requirements of sentiment classification in public opinion analysis scenarios, this paper proposes a model UCRNN that uses user attributes to expand the input information. It used CNN network to extract user attribute features and used Bi-LSTM to extract text data features, and the features were fused in parallel for emotion classification. Through the data obtained on social media Sina Weibo, the experiment proved that UCRNN can achieve the best classification result with less training time than TextRNN.

UCRNN is a very effective solution for social media text emotion classification task, but there are still some problems. On the one hand, in terms of the selection of user attributes, some of the 12 user attributes selected in this paper have been confirmed by previous studies to be directly related to emotion categories, but some of them cannot be determined whether their addition has a positive or negative effect on the whole model. In the experiment, it is found that too many choices of user attributes may cause overfitting, and too few choices cannot greatly improve the classification effect. Further research will conduct a more in-depth analysis of this problem, select a more appropriate user attribute collocation and give the reasons. On the other hand, whether the selection of the network model and the integration of the dual model can be improved also needs further research. Using CNN to extract user attributes in UCRNN is a good method, but at the beginning of designing the model, a fully connected neural network was also considered. In the fusion part of the two models, this model uses the parallel splicing of feature vectors, and whether the use of weight distribution or serial connection can get better classification results still needs further verification.

The success of UCRNN is mainly due to its use of multi-modal information, which can increase the amount of features contained in small-scale data. The use of multi-modal information will be a very challenging and meaningful research direction in the field of artificial intelligence, which mainly includes two points: firstly, the choice of multi-modal information, which should have the characteristics of easy access and strong relevance; secondly, how to reduce the parameters of the model, because the amount of parameters of the neural network that extracts the multi-modal information features has a great impact on the training time.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, Minneapolis, MI, USA, Jun. 2019, pp. 4171–4186.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, Long Beach, CA, USA, Dec. 2017, pp. 5999–6009.

[3] G. Tang, M. Müller, A. Rios, and R. Sennrich, "Why self-attention? A targeted evaluation of neural machine translation architectures," 2018, *arXiv:1808.08946*. [Online]. Available: https://arxiv.org/abs/1808.08946

[4] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. NIPS*, Vancouver, BC, Canada, Dec. 2019, pp. 1–18.

[5] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: https://arxiv.org/abs/1408.5882

[6] D.-G. Ko, S.-H. Song, K.-M. Kang, and S.-W. Han, "Convolutional neural networks for character-level classification," *IEIE Trans. Smart Process. Comput.*, vol. 6, no. 1, pp. 53–59, Feb. 2017.

[7] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," 2016, *arXiv:1607.01759*. [Online]. Available: https://arxiv.org/abs/1607.01759

[8] F. Li, M. Zhang, G. Fu, T. Qian, and D. Ji, "A Bi-LSTM-RNN model for relation classification using low-cost sequence features," 2016, *arXiv:1608.07720*. [Online]. Available: http://arxiv.org/abs/1608.07720

[9] R. Wang, Z. Li, J. Cao, T. Chen, and L. Wang, "Convolutional recurrent neural networks for text classification," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Austin, TX, USA, Jan. 2019, pp. 2267–2273.

[10] X. Sun and J. He, "A novel approach to generate a large scale of supervised data for short text sentiment analysis," *Multimedia Tools Appl.*, vol. 79, no. 9, pp. 5439–5459, Feb. 2018, doi: 0.1007/s11042-018-5748-4.

[11] A. Kumar, K. Srinivasan, W.-H. Cheng, and A. Y. Zomaya, "Hybrid context enriched deep learning model for fine-grained sentiment analysis in textual and visual semiotic modality social data," *Inf. Process. Manage.*, vol. 57, no. 1, Jan. 2020, Art. no. 102141, doi: 10.1016/j.ipm.2019.102141.

[12] S. Bairavel and M. Krishnamurthy, "Novel OGBEE-based feature selection and feature-level fusion with MLP neural network for social media multimodal sentiment analysis," *Soft Comput.*, vol. 24, no. 24, pp. 18431–18445, Dec. 2020, doi: 10.1007/s00500-020-05049-6.

[13] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, "Using linguistic cues for the automatic recognition of personality in conversation and text," *J. Artif. Intell. Res.*, vol. 30, pp. 457–500, Nov. 2007.

[14] L. Mitchell, M. R. Frank, K. D. Harris, P. S. Dodds, and C. M. Danforth, "The geography of happiness: Connecting Twitter sentiment and expression, demographics, and objective characteristics of place," *PLoS ONE*, vol. 8, no. 5, May 2013, Art. no. e64417, doi: 10.1371/journal.pone.0064417.

[15] X. Li, Z. Wang, C. Gao, and L. Shi, "Reasoning human emotional responses from large-scale social and public media," *Appl. Math. Comput.*, vol. 310, pp. 182–193, Oct. 2017.

[16] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," 2018, *arXiv:1802.05365*. [Online]. Available: https://arxiv.org/abs/1802.05365

[17] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1532–1543.

[18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*. [Online]. Available: https://arxiv.org/abs/1301.3781

[19] L. N. Smith, "Cyclical learning rates for training neural networks," 2015, *arXiv:1506.01186*. [Online]. Available: https://arxiv.org/abs/1506.01186

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," 2015, *arXiv:1502.01852*. [Online]. Available: https://arxiv.org/abs/1502.01852

MINZHENG XUANYUAN was born in Henan, China, in 1995. He received the bachelor's degree from the Department of Mechanical Engineering, Shanghai University of Engineering Technology, China, in June 2017. He is currently pursuing the master's degree in computer science with the Department of Information Science and Engineering, Henan University of Technology, China. He is also serving as a Research Assistant with the Key Laboratory of Grain Information Processing and Control, Ministry of Education. His recent research interests include natural language processing, social networks, and sentiment analysis.

LE XIAO received the master's degree from the Department of Computer Science, Beijing Jiaotong University, Beijing, China. He is currently working as an Assistant Professor with the Department of Information Science and Engineering, Henan University of Technology, China. His recent research interests include machine learning, deep learning, natural language processing, sentiment analysis, and knowledge graphs.

MENGSHI DUAN was born in Henan, China, in 1995. She received the bachelor's degree from the Department of Software Engineering, Huanghuai College, China, in June 2019. She is currently pursuing the master's degree in computer science with the Department of Information Science and Engineering, Henan University of Technology, China. She is also serving as a Research Assistant with the Key Laboratory of Grain Information Processing and Control, Ministry of Education. Her recent research interests include natural language processing and knowledge graphs.