Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2021.018896
Article

Tech Science Press

Research on Viewpoint Extraction in Microblog

Yabin Xu[1,2,*], Shujuan Chen[2] and Xiaowei Xu[3]

[1]Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing, 100101, China
[2]Computer School, Beijing Information Science and Technology University, Beijing 100101, China
[3]Department of Information Science, University of Arkansas at Little Rock, Little Rock, 72204, USA
*Corresponding Author: Yabin Xu. Email: xyb@bistu.edu.cn
Received: 25 March 2021; Accepted: 29 April 2021

Abstract: In order to quickly get the viewpoint of key opinion leaders(KOL) on public events, a method of opinion mining in Weibo is put forward. Firstly, according to the characteristics of Weibo language, the non-viewpoint sentence recognition rule is formulated, and some non-viewpoint sentence is eliminated accordingly. Secondly, based on the constructed FastText-XGBoost viewpoint sentence recognition model, the second classification is carried out to identify the opinion sentence according to the dominant and recessive features of Weibo. Finally, the group of evaluation object and evaluation word is extracted from the opinion sentence, according to our proposed multi-task learning BiLSTM-CRFs model. In design, the "BIO" tagging mode is adopted. The sequence tagging of evaluation object and evaluation word based on LSTM-CRFs is conducted as the main task, and the loss function of the main task is optimized by the part of speech tagging based on BiLSTM-CRFs. The experiment result shows that the view recognition model based on FastText-XGBoost has obvious advantages over other recognition models in classification efficiency and accuracy, and the results of the MTL-BiLSTM-CRFs mining is more accurate and the model is more applicable.

Keywords: Viewpoint extraction; opinion sentence recognition; recessive features; multitask learning; sequence tagging

1 Introduction

With the rapid development and popularization of mobile Internet and Web3.0 technology, the social network has become an important channel for people to obtain information. As "new things can be found anytime, anywhere", more and more people get, comment and forward messages through Weibo. Weibo has become a mainstream media of opinion dissemination, and key opinion leaders (KOL) often play a pivotal role in guiding public opinions.

Therefore, the viewpoint extraction of KOL's microblog will help regulators to position the role of opinion leaders in opinion dissemination, which is helpful to understand the development trend of public opinion as a whole. In this way, regulators can supervise and manage social network effectively, by

guiding and consolidating the mainstream ideology. The government can also create a good public opinion environment, which contributes to the sustained and stable development of society.

Our research ideas are shown in Fig. 1. First of all, we set up the non-opinion sentence recognition rule and eliminate obvious non-opinion sentences according to the rule matching method. Thus we reduce the amount of data to be processed. Then, we use the opinion sentence recognition model based FastText-XGBoost to identify the opinion sentence. Finally, we propose the BiLSTM-CRFs model based on multi-task learning. In this way the sequence tagging about evaluation object and evaluation word is conducted as main task, and the part-of-speech tagging is conducted as secondary task, thus realizing a more objective and effective viewpoint extraction.

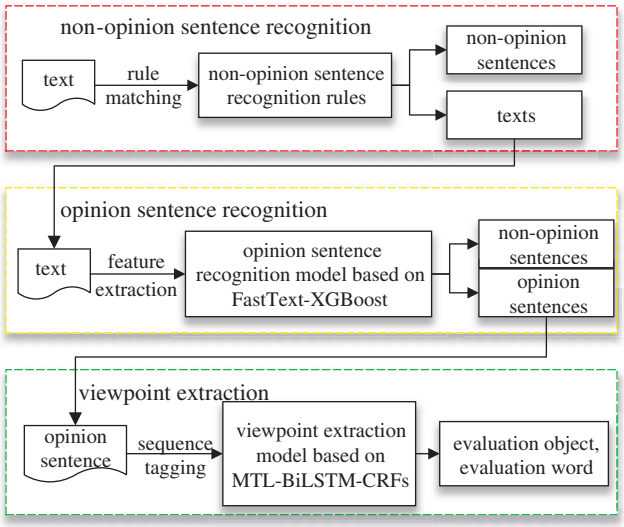


Figure 1: Research ideas

The main innovations of this paper are as follows:

1. Six rules for identifying non-opinion sentences are proposed. We use the recognition rules to match and delete some sentences that are obviously not opinion sentences, so as to reduce the influence of garbage text on the classifier.
2. The opinion sentence recognition model based on FastText-XGBoost is proposed. We classify the dominant and recessive features twice to improve the classification accuracy.
3. The viewpoint extraction model based on MTL-BiLSTM-CRFs is proposed. By increasing the secondary task of part-of-speech tagging, the model can effectively reduce loss value, and greatly improve the accuracy of viewpoint extraction.

2 Related Work

At present, the opinion sentence recognition methods are mainly divided into rule-based unsupervised methods, semi-supervised methods based on boosting optimization, supervised methods based on classifier, and methods based on graph models.

Li et al. [1] formulated a five-level emotion dictionary, a polarity dictionary, and a negative word library. On this basis, he also put forward the non-observation rule and the view rule to identify the non-opinion

sentence and opinion sentence. Hou et al. [2] constructed a phrase-based emotion dictionary and used the template form of keyword matching to construct the phrase rule to identify opinion sentences.

Liu et al. [3] calculated the semantic features of opinion sentences and non-opinion sentences in the test set according to the semantic extraction algorithm. He combined lexical features with part-of-speech features, trained small-scale *corpus* through the bootstrapping algorithm, and got the naïve Bayesian classifier. Finally, He used the classifier to classify large-scale unlabeled *corpus* and added the high-trusted samples into the training model for iterative training until no samples are added.

Hu et al. [4] extracted features depended on sentence dependency and the position of an affective word in sentence dependency, and then applied features to the maximum entropy model to identify opinion sentences. Guo et al. [5] used single word and two part-of-speech as classification features and then used the D-S theory to fuse the results using SVM and naïve Bayesian classifier, respectively, thus, form a multi-classifier to identify the opinion sentence. Literatures Zhao et al. [6,7] selected a variety of classification features and used an SVM classifier to recognize opinion sentences.

Research on viewpoint extraction mainly has the following representative results. Wang et al. [8] used the topic word classification and association rules supplemented by a series of pruning, screening, and delimitation rules to extract the evaluation object in an opinion sentence. Jiang et al. [9] formulated the corresponding extraction rules and the execution order of the rules according to the emotional words, the grammatical components acted by extended emotional words, and the dependency of the emotional word and evaluation object.

Liu et al. [10] fused the lexical features, syntactic features, semantic features, and relative position of the evaluation object into the feature template of CRFs, and proposed a method to extract the recessive evaluation object based on the forwarding relation and the similarity calculation. Literatures Sui et al. [11–13] proposed a sequence tagging method based on BiLSTM-CRFs. Firstly, they used BiLSTM model to learn feature word vectors from the context and then used CRFs to tag the evaluation object.

To sum up, rule-based opinion sentence recognition can simply and quickly separate opinion sentences and non-opinion sentences, but rule-setting is limited by *corpus* and language. In different fields, there are some limitations in opinion sentence recognition. Because the context of microblog presents fragmented features, short space, and messy sentence structure. If we directly summarize several features and use classifier to train without semantic analysis of text, the accuracy and recall rate will be reduced. Using BiLSTM-CRFs model can not only capture the forward and backward information of text, but also ensure the order of label and solve the problem of rare words in sentences. But it ignores the problem that different lexical words in sentences have different weights for viewpoint extraction, and do not increase the efficiency of extraction by adding part-of-speech tagging.

3 Opinion Sentence Recognition

3.1 Recognition of Non-Opinion Sentence

Opinion sentence refers to evaluating a particular object, not including the expression of inner personal wish or mood. Moreover, there must be evaluation object and evaluation word in the opinion sentence.

By analyzing microblog, we can find that some non-opinion sentences have the following characteristics. The content is too short to have no evaluation object and evaluation word at the same time. The sentence beginning with “数据显示” and “调查表明” is a specific introduction to the event. The sentence with “【” and “】” is introductory usually. In addition, according to the definition of opinion sentence, sentences with words that express personal feelings, such as “希望” and “应该”, is not opinion sentences.

Therefore, this paper proposes non-opinion sentence recognition rules, which filter out the dataset that can be directly judged as non-opinion sentence by rule matching, which avoids the influence of junk text on classification.

Rule 1: The sentence that does not contain "#topic#" and has less than 5 words is judged as non-opinion sentence.

Rule 2: The sentence that does not contain "//@username" and has less than 5 words is judged as non-opinion sentence.

Rule 3: The sentence that begins with an objective marker such as "数据显示" and "事实表明" is judged as non-opinion sentence.

Rule 4: The sentence containing "【" and "】" is judged as non-opinion sentence.

Rule 5: The sentence that only has hyperlinks or emoticons without actual words is judged as non-opinion sentence.

Rule 6: The sentence that contains words such as "应该", "希望" and "但愿" is judged as non-opinion sentence.

Among them, rules 1-2 and rules 4-5 directly use the string matching method to discriminate. Rule 3 depends on the established objective identification word dictionary. Rule 6 depends on the established willingness emotional word dictionary. If there is an objective identification word or willingness emotional word the sentence is directly judged as non-opinion sentence.

For example: "【a new iPad will be announced!】Apple will hold a major product launch at 2 am on March 8 at the Fangcao Land Arts Center in San Francisco". this is obvious introductory content, so it can be directly judged as non-opinion sentence. "I hope the Lakers can go into the playoffs and really play against the strong team." This sentence can also be directly judged as non-opinion sentence because of the word "hope".

3.2 Recognition of Opinion Sentence Based on FastText-XGBoost

3.2.1 Dominant Feature of Opinion Sentence

The dominant feature refers to the features that can directly identify the opinion sentence, which are as follows.

(1) The continuous occurrence of punctuation. (Most people often use continuous punctuation to express their feelings)

(2) The occurrence of "!" and "?". (Most people often use exclamatory, rhetorical and ironic sentences to express their opinions.)

(3) The number of emotional words. (Most people often use emotional words to indicate their emotional attitude.)

(4) Whether the sentence contains at least noun, verb, or adjective. (Microblog is very concise and often omits some grammatical elements. Thus the meaning of the sentence is not clear.)

(5) The specific emoticons.

(6) The length of the sentence.

The values of the various features are shown in Tab. 1. 1 denotes non-existence. 0 denotes existence. a denotes the number of emotional words. n denotes the length of the sentence.

Table 1: The values of dominant feature

No.	Feature	Value
1	continuous punctuation exists	[0|1]
2	rhetorical or exclamation question	[0|1]
3	the number of emotional words	a
4	contain at least a noun/verb/adjective	[0|1]
5	emoticons exists	[0|1]
6	length of sentence	n

The statistical algorithm of dominant features are as shown in Algorithm 1:

Algorithm 1: The dominant features statistical algorithm

Input: Collection of sentences $S[]$
Output: The six features results of the sentences $R[]$

1. $FOR(i = 0; i < 26250; i++)\{//no\ more\ than\ train\ set$
2. $GET(S[i]); //get\ the\ corrent\ sentence\ i$
   $//the\ current\ position\ and\ the\ next\ are\ punctuation$
3. $IF\ x\ in\ S[i]\ and\ (x, x+1)\ in\ list\ ('\text{,.!?}, \circ!?')$
   $THEN\ R[i][0] = 1$
   $ELSE\ R[i][0] = 0\ //Rule1$
   $//a\ rhetprical\ or\ exclamation\ question$
4. $IF\ x\ in\ S[i]\ and\ x\ in\ list\ ('!?!?')$
   $THEN\ R[i][1] = 1$
   $ELSE\ R[i][1] = 0\ //Rule2$
5. $IF\ S[i].complie('[\backslash w + \backslash]')//compile\ emoticons$
   $THEN\ R[i][4] = 1$
   $ELSE\ R[i][4] = 0\ //Rule5$
6. $R[i][5] = len(deletePUC(S[i]))\ \ //Rule6$
7. $A = jiebaCUT(S[\text{i}])\ //\ pariciple$
   $IF\ x\ in\ A\ and\ x\ is\ affective\ words\ Count++$
   $THEN\ R[i][2] = Count\ //Rule3$
8. $B = jiebaPOS(S[\text{i}])\ //part\ \ of\ speech\ analysis$
   $IF\ x\ in\ B\ and\ x\ is\ noun||verb||adjective$
   $THEN\ R[i][3] = 1$
   $ELSE\ R[i][3] = 0\ \ //Rule4$
9. $\}\ \ //ENDFOR$

3.2.2 Recessive Feature of Opinion Sentence

Recessive feature refers to analyzing the text from a semantic degree and extract the meaning of the text at a deep level. Since microblog produces thousands of text per second on average, the hidden feature extraction for mass text should not only ensure fast and effectively but also ensure high accuracy. The FastText model produces a simple and efficient method for supervised classification and representation learning. It incorporates n-gram features and takes local word order into account. So the semantic expression can be

more accurate. Thus it has a comparable accuracy with deep learning model, but the training time is nearly ten thousand times less than that of deep learning model. Therefore, this paper takes the value predicted by FastText model as recessive feature of the opinion sentence, valuing 0 or 1 respectively.

As shown in Fig. 2, we firstly delete stop-words and participles in text. Then, we carry out wordEmbedding and get the corresponding word vector$x_i$took as the input of FastText model. We add 2-gram feature in the model. In the example of "各位大腕演技", if we do not consider word order, we will get characteristic words such as "各位", "大腕" and "演技". If we add 2-gram feature into FastText, we will get additional characteristic words such as "各位大腕" and "大腕演技", which can be distinguished from "各位演技大腕". In this way the semantic expression is more accurate.

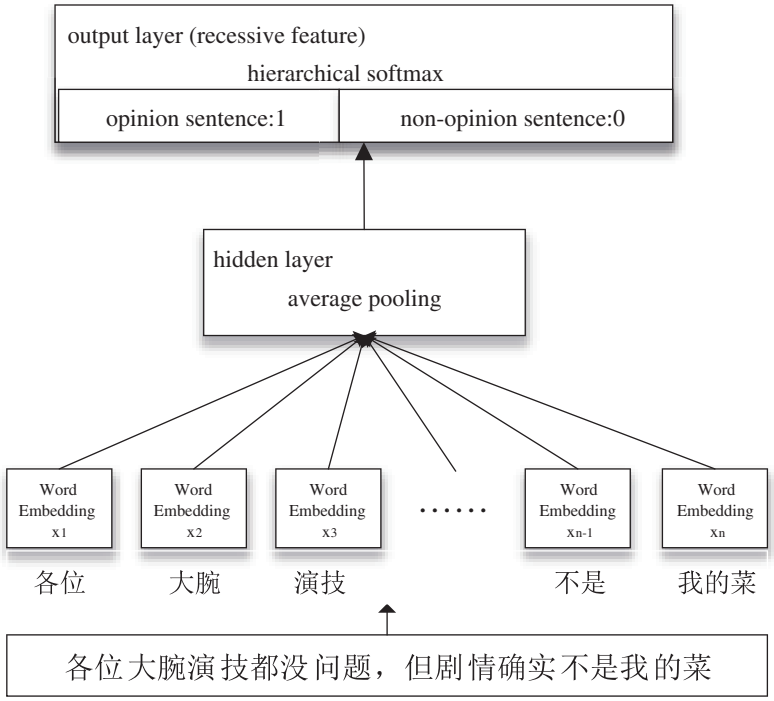


Figure 2: The model of recessive feature extraction based on FastText

The model averages all the word vector x_i in each sentence by using the hidden layer and gets the document vector y_i correspondingly. So we can get the whole document vector Y, as shown in Eq. (1).

Among them, n denotes the number of feature words, and $(x_{1i}, \ldots, x_{mi})$ denotes m-dimensional vector of feature word i.

The classification vector B is the result of document vector Y multiplied by the weight matrix A of the hidden layer. The calculation formula is shown in Eq. (2).

$$Y = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} = \frac{1}{n} \begin{pmatrix} \sum_{i=1}^{n} x_{1i} \\ \vdots \\ \sum_{i=1}^{n} x_{mi} \end{pmatrix} \tag{1}$$

$$B = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} = A \cdot Y \tag{2}$$

Finally, we construct the Huffman tree based on classification vector B and model parameters. Then, we use the hierarchical softmax function to calculate the label, which is shown in Eq. (3).

$$p(z) = R(\prod_{i=1}^{L(z)-1} \sigma(||n(z,l+1) = LC(n(z,l))|| \cdot {\theta_{n(z,l)}}^T B)) \tag{3}$$

Among them, $\sigma(z)$ denotes sigmoid function. $LC(z)$ denotes the left child of node n. $\theta_{n(z,l)}$ is the parameter of the middle node. $||x||$ is a special function, as shown in Eq. (4). $R()$ is a rounding function as shown in Eq. (5).

$$||x|| = \begin{cases} 1, & x == true \\ -1, & otherwise \end{cases} \tag{4}$$

$$R(x) = \begin{cases} x \geq 0.5, & 1 \\ x < 0.5, & 0 \end{cases} \tag{5}$$

3.2.3 Opinion Sentence Recognition Model Based on FastText-XGBoost

The most commonly used method in the existing literature is SVM model, which trains multiple influence features. However, The SVM model is sensitive to the missing values. If there are multiple zeros in eigenvalues, the accuracy will be reduced. In addition, with the emergence of new words, the text often contains implicit emotional words. So we should analyze text features from the semantic point of view because we cannot represent the features only according to dominant features.

0 appears more frequently in dominant features so that we choose the XGBoost model to compensate for the effect of missing value. Moreover, in order to synthesize the opinion sentence recognition from the perspective of text features and semantics, we propose an opinion sentence recognition model based on FastText-XGBoost, as shown in Fig. 3.

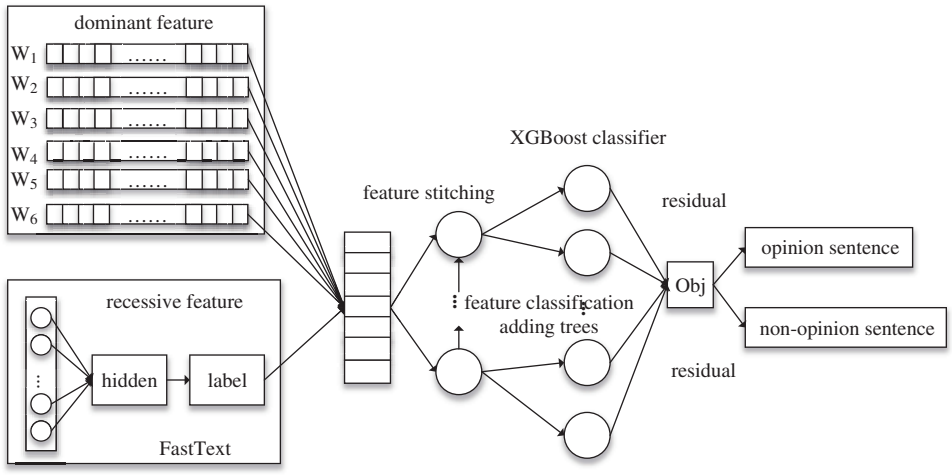


Figure 3: The opinion sentence recognition model based on FastText-XGBoost

In Fig. 3, the model of opinion sentence recognition is a two-classification model. Firstly, we get the dominant features of $W_1, W_2, W_3, W_4, W_5, W_6$. Then we quickly get the recessive features of W_7 through FastText. Finally, we splice together the seven features and get the sentence feature vector x_i, which acts as the input of XGBoost. The relation between x_i and category label y_i is $f(x_i) = y_i$.

Each training for the XGBoost is based on the residuals obtained from the previous round. At the t iteration, the objective function of the resulting tree is:

$$\begin{aligned} Obj^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \\ &= \sum_{i=1}^{n} l(y_i, \hat{y}_i{}^{(t+1)} + f_t(x_i)) + \sum_{k=1}^{K} \Omega(f_k) \end{aligned} \tag{6}$$

Among them, $l(y_i, \hat{y}_i{}^{(t-1)} + f_t(x_i))$ indicates the loss between true score and the residuals of predicted distribution. $\Omega(f_k)$ indicates the complexity of nascent tree. It can control the number and fraction of leaf nodes not to be too large to prevent overfitting.

XGBoost performs Taylor second-order expansion on the Obj function at $x = 0$, so that we can pay more attention to some samples by adjusting parameters. The experiment gradually generates the optimal tree structure by dividing the existing leaf nodes on each step. The final classification model is obtained when the gain value of segmentation is continuously less than the fixed value or when the number of segments reaches the specified maximum depth. Eventually, each sample falls into a leaf node corresponding to a fraction. The predicted value is the result of adding the fraction of each tree.

4 Viewpoint Extraction

4.1 Tagging Pattern of BIO

This paper transforms the problem of extracting the evaluation object and evaluation word from the opinion sentence into the problem of the sequence tagging of each word. We adopt the BIO tagging pattern, in which the meaning of each label is shown in Tab. 2.

Table 2: Tagging pattern of BIO on viewpoint extraction

label	meaning
B-TARGET	the first word of evaluation object
I-TARGET	other words of evaluation object
B-OPINION	the first word of evaluation word
I-OPINION	other words of evaluation word
O	other words

Fig. 4 illustrates the process of extracting evaluation object and evaluation word from opinion sentence using sequence tagging. T of B-T and I-T is the abbreviation of TARGET. O of B-O and I-O is the abbreviation of OPINION. In the end, the extracted binary is "<科比，战神>", in which "科比" is the evaluation object and "战神" is the evaluation word.

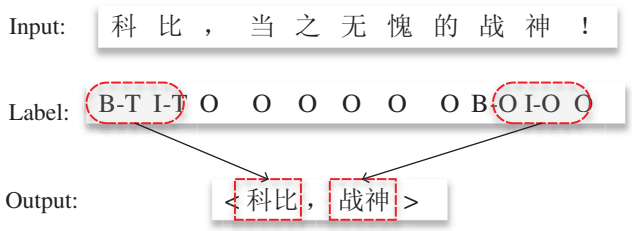


Figure 4: Example of sequence tagging

4.2 Model of BiLSTM-CRFs

BiLSTM is a bidirectional LSTM model consisting of a forward LSTM and a backward LSTM. AN LSTM can learn what to remember and forget by training, so the model can capture the dependencies of long words. But BiLSTM can better capture the information forward and backward.

However, in the tagging pattern of BIO, the labels are interdependent, such as I must after B, and the forward and backward tag of evaluation object and evaluation word are O. Therefore, considering the dependence between labels, we add a CRFs layer after the output of BiLSTM to learn label transfer probability of sentence. The sequence tagging model of BiLSTM-CRFs is shown in Fig. 5.

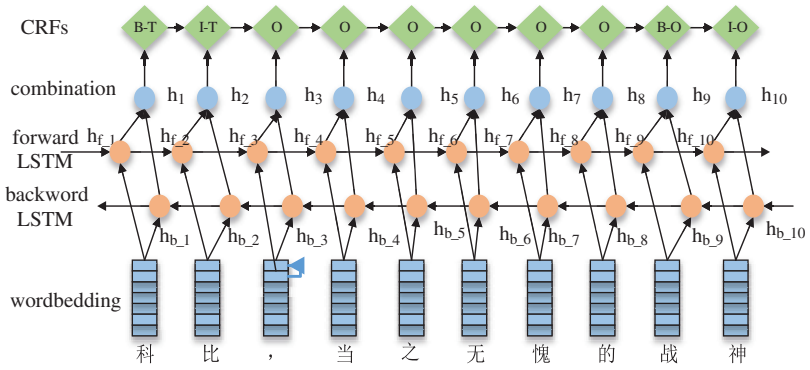


Figure 5: Sequence tagging model of BiLSTM-CRFs

As can be seen from Fig. 5, we put text into segment and map each word into a 180-dimensional word vector, which considers as the input of model.

Then, according to the forgetting gate, the memory gate, and the output gate of LSTM, we can get the sequence of hidden layer state $\{h_0, h_1, \ldots, h_{n-1}\}$, which are the features of a forward and backward sentence. The calculation process is shown in Eqs. (7)–(12).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{7}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{8}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{9}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{10}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{11}$$

$$h_t = o_t * \tanh(C_t) \tag{12}$$

Among them, W_f, W_i, W_C are the weight matrix. b_f, b_i, b_C are the deviation. σ is the sigmoid activation function. f_t denotes the information that x_t need to be forgotten through the forgetting gate at the t moment. i_t denotes the information that x_t need to be remembered through memory gate at the t moment. $\tilde{C}_t$ denotes the temporary cell state at the t moment. tanh is the activation function. C_t denotes the cell state at the t moment. o_t indicates the information of output gate at the t moment. h_t indicates the hidden state at the t moment.

Therefore, the hidden state sequence obtained by forward LSTM is $h_f = \{h_{f_0}, h_{f_1}, \ldots, h_{f_{n-1}}\}$, and the hidden state sequence obtained by backward LSTM is $h_b = \{h_{b_0}, h_{b_1}, \ldots, h_{b_{n-1}}\}$. The two sequence are fused in the combination layer. The final hidden state sequence is $h = \{h_0, h_1, \ldots, h_{n-1}\} = \{h_{f_0} \oplus h_{b_0}, h_{f_1} \oplus h_{b_0}, \ldots, h_{f_{n-1}} \oplus h_{b_{n-1}}\}$, which is used as the input of CRFs $x = \{x_0, x_1, \ldots, x_{n-1}\}$.

The CRFs model adopts the linear chain component to annotate sequence. In the conditional probability model $P(X|Y)$, Y severs as the output variable, representing the sequence of labels, and X servers as the input variable, representing the sequence to be labeled. Then, under the condition that the random variable X takes the value x, the conditional probability that the random variable Y takes the value y is:

$$p(x|y) = \frac{1}{Z(x)} \exp[\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_{i-1}, y_i, x, i)] \tag{13}$$

$$Z(x) = \sum_{y} \exp[\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_{i-1}, y_i, x, i)] \tag{14}$$

Among them, λ_k and μ_l are weight parameters. $Z(x)$ is the normalization factor, and the summation is performed on all possible output sequences. t_k is the transfer feature function, which depends on the current and the previous position. s_l is the state feature function, which depends on the current position. In general, the feature functions t_k and s_l are valued at 1 or 0. The value is 1 when the feature condition is satisfied, otherwise the value is 0.

4.3 Viewpoint Extraction Model Based on MTL-BiLSTM-CRFs

The BiLSTM-CRFs model can learn context features to solve the problem of the emergence of missing words. However, the performance of model depends on the size of tagged *corpus*, and it also increases the semantics through the method of replacing missing words with the "UNK" symbol. Considering that different lexical words in a sentence have different weights for the evaluation object and evaluation word, this paper proposes a viewpoint extraction model of multi-task learning based on BiLSTM-CRFs to solve the above problem, as shown in Fig. 6.

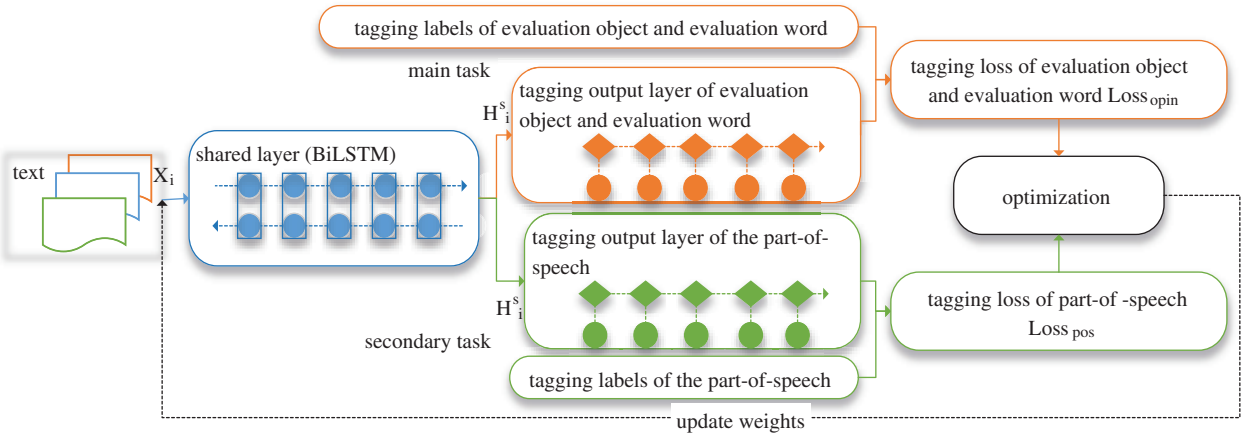


Figure 6: Viewpoint extraction model based on MTL-BiLSTM-CRFs

We use two tasks to construct the viewpoint extraction model based on MTL-BiLSTM-CRFs as Literature Xu et al. [14]. The main task is the sequence tagging of evaluation object and evaluation word based on BiLSTM-CRFs. The secondary task is the part-of-speech tagging based on BiLSTM-CRFs. With the help of the secondary task's loss, the main task is optimized, which makes the model learn more abundant knowledge in the limited tagged *corpus*. For example, we can get the noun evaluation object, the verb evaluation object, the noun evaluation word, the adjective evaluation word, the adverb and adjective evaluation word, and so on. The part-of-speech to be tagged are noun, verb, adjective, and adverb. The task also uses the tagging pattern of BIO, as shown in Tab. 3.

Table 3: Tagging pattern of BIO on part-of-speech

label	meaning
B-N	first word of nouns
I-N	other words of nouns
B-V	first word of verbs
I-V	other words of verbs
B-ADJ	first word of adjectives
I-ADJ	other words of adjectives
B-ADV	first word of adverbs
I-ADV	other words of adverbs
O	other words

The viewpoint extraction algorithm based on MTL-BiLSTM- CRFs is shown in Algorithm 2.

Algorithm 2: The algorithm of viewpoint extraction based on MTL-BiLSTM-CRFs.

Input: training set with corresponding main task label y_1 and secondary task label y_2, testing set.

Output: main task y_1 label and secondary task y_2 label of testing set.

Step 1: count word frequency, delete words that appear less than 2 times, and form each word into a 180-dimensional vector through word embedding. The label y_1 is made into a 5-dimensional vector through one-hot, and the label y_2 is made into a 6-dimensional vector through one-hot.

Step 2: use Keras to build a multi-task learning model, and take 180-dimensional vectors, 5-dimensional vectors, and 9-dimensional vectors as input to the model.

Step 3: After the input vector is encoded by the shared layer BiLSTM in 256 dimensions, the model appears branched. One of the branches is the prediction output of evaluation object and evaluation word after passing through a CRFs layer. The other branch is the prediction output of part-of-speech after passing a CRFs layer.

Step 4: take 80 percent of text as the training set and 20 percent of text as the validation set for model training [15,16]. In order to prevent overfitting, set Epoch is 10. Take 16 batches as sample for each training. Use the loss function of CRFs as the loss function of the model. And also use the Adam optimizer to calculate the adaptive learning rate with different parameters.

Step 5: use the trained model to predict the test set and obtain the corresponding label y_1 and y_2.

5 Experiment and Analysis

5.1 Experimental Data and Environment

5.1.1 Experimental Data

The experimental data contains two public datasets, which are NLPC2012 and NLPC2013. But considering the update rate of new words, we add a large number of new text. The whole dataset contains 43395 sentences. But 6434 non-opinion sentences are deleted in advance by using the non-opinion sentence recognition rule. Eventually, there are 5528 opinion sentences and 31979 non-opinion sentences in the dataset.

In the opinion sentence recognition experiment, 37507 sentences are divided into training set and test set according to 7:3 ratio. In the viewpoint extraction experiment, 5528 opinion sentences are divided into a training set and a test set according to 7:3 ratio.

The form of microblog text is complex and changeable. The text includes various citing topics, emoticons, links, pictures, and "@username" to increase the readability. Therefore, in order to avoid the influence of these special expressions on text classification, it is necessary to preprocess text and replace the above special expressions with empty strings.

5.1.2 Experimental Environment

The experimental environment is shown in Tab. 4.

Table 4: Environment of experiment

Environment	Version
CPU	Intel®Core™i5-4590CPU@3.3 GHz
server	Sugon Xmachine W780-G20
GPU	TESLA P100 16 GB -E3x16250W
internal memory	8 GB
operation system	linux-ubuntu16.04
developing language	Python 3.6
developing environment	Pycharm
database	MySQL

5.2 Experiments of Opinion Sentence Recognition

5.2.1 Experiment on Parameter Selection of FastText

In order to select the optimal word vector dimension, we select eight dimensions (50 dim, 100 dim, 150 dim, 200 dim, 250 dim, 300 dim, 350 dim and 400 dim) for 10-fold cross-validation. In the same experimental environment, we observe the variation of F1 value of FastText in different word vector dimensions. The experimental results are shown in Fig. 7.

It can be seen from Fig. 7 that when the word vector dimension is 300 dim, there is an average optimal value of F1. So we choose the word vector dimension of FastText as 300 dim.

In order to select the optimal n-gram value to make FastText have higher accuracy of classification. In the same experimental environment, we do experiment with different n-gram of 1, 2, 3, 4 and 5 to get the classified effect on accuracy rate, recall rate, F1 value, training time and test time. The experimental results are shown in Fig. 8.

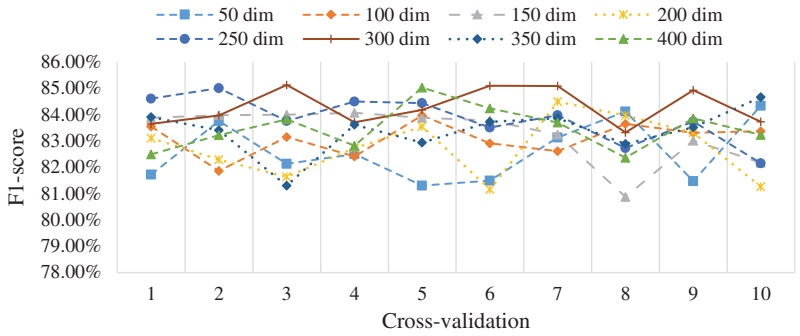


Figure 7: Cross validation of different word vector dimension of FastText

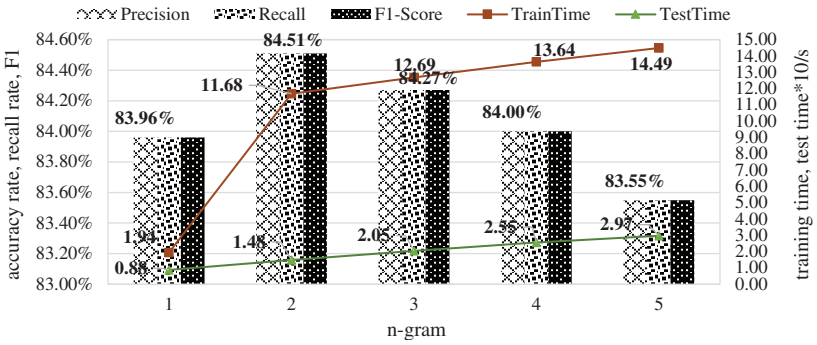


Figure 8: Classification effect of different n-gram of FastText

It can be seen from Fig. 8 that the model has the best classification effect when the value of n-gram is 2. After that, as the n-gram value continues to increase, accuracy rate, recall rate and F1 value have been declining, and the training time and test time have been increasing. Therefore, when the n-gram is 2, the opinion sentence recognition effect of FastText is the best.

5.2.2 Experiment on Parameter Selection of XGBoost

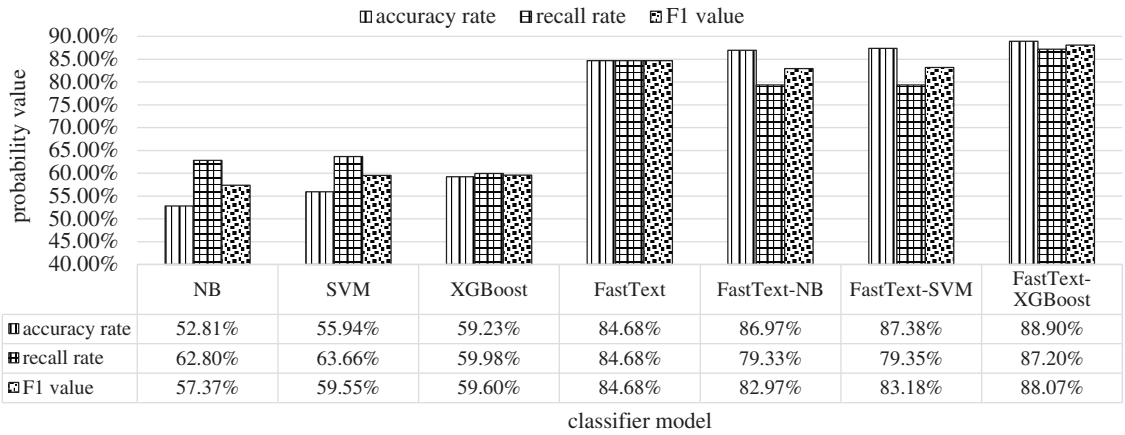
In order to achieve the best classification effect of XGBoost, we optimize parameters by cross-validation. In the same experimental environment, we use the average absolute error MAE to measure the error between true label and predicted label [17–19]. The setting of parameters is shown in Tab. 5. The MAE in training set and test set is about 0.35.

Table 5: Parameters setting of XGBoost

parameters	value	MAE	
num_boost_round: the number of trees	200	**train**	**test**
max_depth: the depth of trees	6	0.3519	0.3590
min_child_weight: the smallest sample weights of child nodes	7		
gamma: the value of loss reduction	0.2		
subsample: random sampling of training samples	0.7		
colsample_bytree: column sampling when the tree is generated	0.8		
eta: learning rate	0.08		

5.2.3 Contrast Analysis of Opinion Sentence Recognition

The first experiment is to verify the effect of opinion sentence recognition on XGBoost. We do experiments with XGBoost compared to the naïve Bayesian (NB) and the support vector machine (SVM). Under the same test set and experimental environment, the contrast experimental results on accuracy rate, recall rate, and F1 are shown in Fig. 9.



	NB	SVM	XGBoost	FastText	FastText-NB	FastText-SVM	FastText-XGBoost
accuracy rate	52.81%	55.94%	59.23%	84.68%	86.97%	87.38%	88.90%
recall rate	62.80%	63.66%	59.98%	84.68%	79.33%	79.35%	87.20%
F1 value	57.37%	59.55%	59.60%	84.68%	82.97%	83.18%	88.07%

Figure 9: Contrast experimental results of opinion sentence recognition

It can be seen from Fig. 9 that the accuracy rate and F1 value are higher than those of NB and SVM model. NB model has a good effect on small-scale data, but its classification results of large-scale data have a large error rate. SVM model is more sensitive to the missing value. So it reduces the classification effect due to the emergence of multiple 0 in dominant features of the sentence in the training set. Whereas XGBoost model is insensitive to missing values. It can learn the splitting direction of features that contain the missing value and add regularization into objective function in order to control the complexity of the model. Therefore, its accuracy rate and efficiency are better than NB and SVM classifier.

The second experiment is to verify the effect of opinion sentence recognition based on FastText-XGBoost. We do experiments with FastText-XGBoost compared to XGBoost, FastText, FastText-NB and FastText-SVM. Under the same experiment environment and test set, the contrast experimental results are shown in Fig. 9.

It can be seen from Fig. 9 that the deep learning model of FastText has a strong linear fitting ability, and is able to self-learn recessive features. So the accuracy is 25% higher than that of the machine learning model. The two-classification model based on FastText-XGBoost considers the dominant and recessive features of text synthetically so that its classification effect is higher than that of one-classification model. The effect of XGBoost is better than NB and SVM, so the accuracy rate, recall rate, and F1 value of FastText-XGBoost are the highest. Therefore, the opinion sentence recognition model based on FastText-XGBoost can effectively improve the recognition effect.

5.3 Experiments of Viewpoint Extraction

The first experiment is to verify the rationality and validity of the proposed viewpoint extraction model based on MTL-BiLSTM-CRFs. We do experiments comparingthe BiLSTM-CRFs based on a single task with the BiLSTM- CNN-CRFs model adding CNN layers. In order to prevent overfitting, we perform 10 Epoch tests on the same test set. The accuracy comparison results of each evaluation object and evaluation word extracting experiment are shown in Fig. 10.

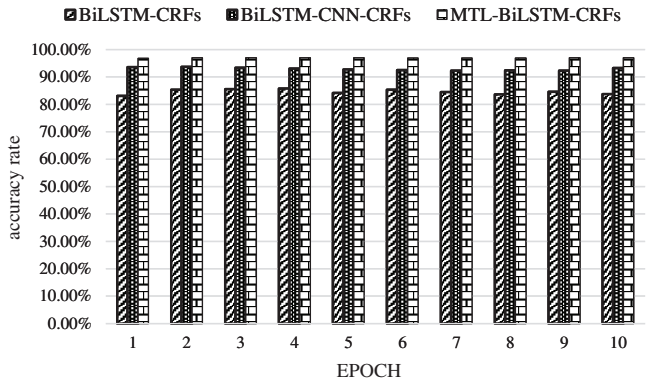


Figure 10: Experimental results of viewpoint extraction

It can be seen from Fig. 10 that the viewpoint extraction accuracy of BiLSTM-CRFs is the lowest. When CNN layers are added to the BiLSTM-CRFs model, the accuracy is further improved. This is due to the addition of word embedding at the character level to extract deep local features.

Since the MTL-BiLSTM-CRFs proposed in this paper uses the multi-task learning framework. It adds the secondary task of part-of-speech, considering that the word of different parts-of-speech has different weights on the evaluation object and evaluation word. And it can use the loss of secondary tasks to optimize the loss of the main task, reduce the loss of main task, and learn more complex knowledge. Therefore, the model greatly improves the accuracy rate.

The second experiment is to verify the applicability of the proposed MTL-BiLSTM-CRFs for viewpoint extraction under different topics. We choose three different topics of "crazy scallion", "Philip ship malicious impact" and "official property publicity" in the NLPC2012 dataset. Each topic has 300 opinion sentences. In the same experimental environment, the contrast results of viewpoint extraction by MTL-BiLSTM-CRFs model, BiLSTM-CRFs model, and BiLSTM-CNN-CRFs model are shown in Tab. 6.

Table 6: Experimental results of viewpoint extraction under different topics

topic	#crazy scallion#		#official property publicity#		#Philip ship malicious impact#	
evaluating indicators	loss	accuracy rate	loss	accuracy rate	loss	accuracy rate
BiLSTM-CRFs	11.0846	78.21%	5.8474	75.09%	4.6402	73.98%
BiLSTM-CNN-CRFs	0.1023	92.20%	0.1099	91.16%	0.1112	93.29%
MTL-BiLSTM-CRFs	**0.0771**	**95.76%**	**0.0716**	**94.77%**	**0.0739**	**94.59%**

It can be seen from Tab. 6 that because BiLSTM-CRFs model is limited by the training *corpus*, the accuracy of viewpoint extraction is the lowest and the loss is the largest under different topics. The BiLSTM-CNN-CRFs model greatly improves accuracy by adding character-level features. The MTL-BiLSTM-CRFs model can learn more complex knowledge in viewpoint extraction because adding the secondary task of part-of-speech. Thus, it further improves the accuracy of viewpoint extraction. Because the loss function of secondary task can optimize the loss of main task in the multi-task learning framework, the loss of the model for viewpoint extracting under different topics is minimized. It can be seen that the MTL-BiLSTM-CRFs model proposed in this paper has the highest applicability for viewpoint extracting under different topics.

6 Conclusion

This paper first uses recognition rules to identify non-opinion sentence according to the linguistic features of microblog. Thus, it can reduce the impact of spam text on feature selection. In order to solve the problem of low accuracy in the existing methods, this paper proposes an opinion sentence recognition model based on FastText-XGBoost. Firstly, the label of text predicted by FastText is used as the recessive feature. Then, the XGBoost model is used to classify both the recessive feature and the six dominant features. In this way, opinion sentence and non-opinion sentence are identified. On this basis, this paper further proposes a viewpoint extraction model based on MTL-BiLSTM-CRFs. The model adds the secondary task of part-of-speech tagging in order to assist the main task of viewpoint extraction, so that we can learn more complex language knowledge and extract the evaluation object and evaluation word more effectively and accurately.

The experiment results show that this method can not only identify opinion sentences accurately, but also extract evaluation objects and evaluation words in opinion sentences with high accuracy. The method has high applicability and low loss for viewpoint extraction in different fields.

Funding Statement: This research is supported by The National Natural Science Foundation of China under Grant (No. 61672101), Beijing Key Laboratory of Internet Culture and Digital Dissemination Research (No. ICDDXN004) and Key Lab of Information Network Security of Ministry of Public Security (No. C18601).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. G. Li, G. X. Zhou, Y. Sun and H. G. Zhang, "Research and implementation of Chinese microblog sentiment classification," *Journal of Software*, vol. 28, no. 12, pp. 3183–3205, 2017.

[2] M. Hou, Y. Teng, X. Y. Li, Y. Chen, S. Zheng *et al.,* "Study on the linguistic features of the topic-oriented microblog and the strategies for its sentiment analysis," *Applied Linguistics*, vol. 10, no. 2, pp. 135–143, 2013.

[3] R. Liu, X. Y. Hao and Y. Li, "Research on Chinese perspective sentences identification based on semantic pattern with semi-supervised machine learning," *Journal of Nanjing University (Nature Science)*, vol. 54, no. 5, pp. 967–973, 2018.

[4] M. Z. Hu and T. F. Yao, "Recognition of Chinses micro-blog sentiment polarity and extraction of opinion target," *Journal of Shandong University (Natural Science)*, vol. 51, no. 7, pp. 81–89, 2016.

[5] Y. L. Guo, Y. B. Pan, Z. Y. Zhang and L. Li, "Multiple-classifiers opinion sentence recognition in Chinese micro-blog based on D-S theory," *Computer Engineering*, vol. 40, no. 4, pp. 159–163, 2014.

[6] J. Zhao and R. Wen, "Recognition of opinion bearing sentences in microblogs based on new words extension and feature selection," *Journal of the China Society for Scientific and Technical Information*, vol. 32, no. 9, pp. 945–951, 2013.

[7] Y. X. Pan and T. F. Yao, "Recognition of microblog customer opinion sentences in automobiles domain," *Journal of Chinese Information Processing*, vol. 28, no. 5, pp. 148–154, 2014.

[8] G. Q. Wang, X. Tian, D. G. Huang and J. Zhang, "Research on Chinese microblog opinion sentence recognition and element extraction," *Data Acquisition and Processing*, vol. 31, no. 1, pp. 160–167, 2016.

[9] T. J. Jiang, C. X. Wan, D. X. Liu, X. P. Liu and G. Q. Liao, "Extracting target-opinion pairs based on semantic analysis," *Chinese Journal of Computers*, vol. 40, no. 3, pp. 617–633, 2017.

[10] Q. C. Liu, H. Y. Huang and C. Feng, "Co-extracting opinion targets and opinion-bearing words in Chinese micro-blog texts," *ACTA Electronica SINICA*, vol. 44, no. 7, pp. 1662–1670, 2016.

[11] G. Q. Sui, N. Zhao and Z. Peng, "Approach to extracting opinion from products reviews based on deep learning and CRFs," *Journal of Intelligence*, vol. 38, no. 5, pp. 177–185, 2019.

[12] Z. Huang, W. Xu and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," *arXiv preprint arXiv*, vol. 15, no. 3, pp. 1–10, 2015.

[13] Y. Zhang, H. S. Chen, Y. H. Zhao, Q. Liu and D. W. Yin, "Learning tag dependencies for sequence tagging," in *Proc. of the Twenty-Seventh Int. Joint Conf. on Artificial Intelligence (IJCAI-18)*, Stockholm, Sweden, pp. 4581–4587, 2018.

[14] Y. Xu, X. Meng, Y. Li and X. Xu, "Research on privacy disclosure detection method in social networks based on multi-dimensional deep learning," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 137–155, 2020.

[15] Y. Li, X. Wang, W. Fang, F. Xue, H. Jin *et al.,* "A distributed ADMM approach for collaborative regression learning in edge computing," *Computers, Materials & Continua*, vol. 59, no. 2, pp. 493–508, 2019.

[16] M. Luo, K. Wang, Z. Cai, A. Liu and Y. Li, "Using imbalanced triangle synthetic data for machine learning anomaly detection," *Computers, Materials & Continua*, vol. 58, no. 1, pp. 15–26, 2019.

[17] H. Wang, Q. Xue, T. Cui, Y. Li and H. Zeng, "Cold start problem of vehicle model recognition under cross-scenario based on transfer learning," *Computers, Materials & Continua*, vol. 63, no. 1, pp. 337–351, 2020.

[18] K. Yang, J. Jiang and Z. Pan, "Mixed noise removal by residual learning of deep CNN," *Journal of New Media*, vol. 2, no. 1, pp. 1–10, 2020.

[19] L. Chen, B. Wang, W. Yu and X. Fan, "CNN-based fast HEVC quantization parameter mode decision," *Journal of New Media*, vol. 1, no. 3, pp. 115–126, 2019.