Context Sentiment Classification Based on Improved Deep Extreme Learning Machine

Liu Ronghui* and Zhang Jingpu

Abstract—**In order to improve context sentiment classification accuracy, a sentiment classification algorithm based on deep extreme learning machine with the nearest neighbor and sparse representation (NS-DELM) is presented. Firstly, the extreme learning machine (ELM) is combined with the auto encoder. Secondly, the idea of sparsity and nearest neighbor is integrated into the deep network, and the data integrity is maintained through sparse representation in the projection process. Thirdly, the local manifold structure of the data is maintained by the nearest neighbor representation, and the deep features of the data are extracted layer by layer unsupervised; Finally, the least squares is solved by supervised learning for context sentiment classification. The method is applied to the context sentiment classification experiment of shopping comments. Compared with support vector machine (SVM), stacked auto-encoder (SAE) and convolutional neural network (CNN), the experimental results show that NS-DELM algorithm has higher accuracy than other existing algorithms.**

Index Terms—**sentiment analyzing, deep learning, nearest neighbor representation, sparse representation**

I. Introduction

As the rapid development of internet, the Internet mode transforms from "reading internet" to "interactive internet". As a result, internet network is not only a place to obtain information, but also a way to express people opinion and share their own experience, namely an important platform to show people sentiment information directly [1-5]. Therefore, sentiment information has great meaning for business to guide its development to satisfy customer need.

Sentiment classification is proposed by Bo Pang in 2002 [6], which is also called sentiment analyzing or opinion mining. Sentiment analyzing utilizes some methods and technologies to analyze and mine the emotion information from context. Sentiment classification algorithms mainly contain two kinds: unsupervised learning method and supervised learning method. For unsupervised learning method, sentiment classification is performed without prior annotation labels. Literature [7] proposes unsupervised linguistic approach for sentiment classification from online reviews using SentiWordNet 3.0. Literature [8] presents a novel cross-lingual topic model framework which can be easily combined with the state-of-the-art aspect/sentiment models. The presented algorithm solves the problem that existing cross-lingual topic models are not suitable for sentiment classification because sentiment factors are not considered therein. For supervised learning method, sentiment classification is treated as a special context classification process. Literature [9] propose a method for sentiment classification based on word2vec and SVM. In literature [10], several neural networks are applied to sentiment classification, such as back propagation neural network (BPN), probabilistic neural network (PNN) and homogeneous ensemble of PNN. Experiment result shows that the homogeneous ensemble of PNN method provides better performance. As a supervised learning algorithm, extreme learning machine (ELM) is a novel machine learning algorithm for single-hidden layer feed-forward neural networks (SLFNs) [11, 12, 13]. It has been applied in the fields of traffic sign recognition[14], fault diagnosis[15], time series prediction[16], etc. However, for conventional ELM, on the one hand, ELM with too few hidden nodes may encounter under-fitting problem; on the other hand, ELM with too many hidden nodes may lead to over-fitting, and this causes matrix singularity and ill-posed problems. Both under-fitting and over-fitting problem may reduce sentiment classification accuracy. In addition, ELM has poor robustness because its input weights and thresholds are generated randomly and there is only one hidden layer.

In recent years, deep learning algorithm has become a research hotspot in the field of context sentiment classification [17-20]. This algorithm can get rid of the shortcomings of traditional shallow learning algorithm. The shallow features of data can be learned at the bottom layer, and then be input to the next layer to form a more complex representation of input data. The deep features of data can be learned in the deep structure. Hinton et al. proposed a concept of deep neural network and its training method. The algorithm takes the input as the output, and uses unsupervised greedy layer by layer training method to train each layer of the network in turn. Then softmax and other classifiers to train the last layer are used. SAE (stacked auto-encoder) is one model to use the mentioned algorithm to train. However, due to the increase of network layers, the training time of this deep network is too long, and it is easy to fall into the local optimal solution.

In order to overcome the afore-mentioned problem, a novel sentiment classification algorithm based on nearest neighbor and sparse representation deep extreme learning machine (NS-DELM) is proposed. The proposed method adds the idea of sparse and nearest neighbor according to ELM auto-encoder structure. In the process of projection, the global structure of data is preserved by sparse representation, and the local manifold structure of data is preserved by

Manuscript received March 3rd, 2021; revised June 28th, 2021.

Liu Ronghui is an Associate Professor of the School of Computer and Data Science, Henan University of Urban Construction, Pingdingshan, 467036, China (corresponding author, e-mail: liurh_126@126.com).

Zhang Jingpu is a Lecturer of the School of Computer and Data Science, Henan University of Urban Construction, Pingdingshan, 467036, China (e-mail: 20181027@hncj.edu.cn).

nearest neighbor representation. The compressed representation of data is extracted layer by layer unsupervised. Finally, supervised learning is used for context sentiment classification.

II. Neighbor and spare representation based deep extreme learning machine

A. Deep extreme learning machine

ELM is essentially a single hidden layer feedforward neural network. Given a supervised learning problem with N different data samples $\{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{N}$, the training process consists of two stages: ① according to the number of hidden layer nodes, the input weight matrix $\boldsymbol{a}_i \in \mathbb{R}^n$ and biases $b_i \in \mathbb{R}$ are randomly generated to calculate the hidden layer output; ② according to the data label, the output matrix $\boldsymbol{\beta}$ is obtained by solving the least square.

$$f_L(\boldsymbol{x}) = \sum_{i=1}^{L} \boldsymbol{\beta}^i G(\boldsymbol{a}_i, b_i, \boldsymbol{x}) \tag{1}$$

$$\hat{\boldsymbol{\beta}} = \boldsymbol{H}^{\dagger}\boldsymbol{T} \tag{2}$$

where

$$\boldsymbol{H} = \begin{bmatrix} G(\boldsymbol{a}_1, b_1, \boldsymbol{x}_1) & \cdots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_1) \\ \vdots & & \vdots \\ G(\boldsymbol{a}_1, b_1, \boldsymbol{x}_N) & \cdots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_N) \end{bmatrix}_{N\times L} = \begin{bmatrix} \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_N \end{bmatrix},$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}^1 \\ \vdots \\ \boldsymbol{\beta}^L \end{bmatrix}_{L\times m}, \quad \boldsymbol{T} = \begin{bmatrix} \boldsymbol{t}_1 \\ \vdots \\ \boldsymbol{t}_N \end{bmatrix}_{N\times m} \tag{3}$$

where $\boldsymbol{H}^{\dagger}$ is a Moore-Penrose pseudo-inverse of $\boldsymbol{H}$. G (·) is the activation function, whose usual type is sigmoid function or tanh function.

Deep ELM is a multiple layer network structure that combines the idea of self-coding with ELM. The structure is shown in Figure 1. Deep ELM combines the learning efficiency of ELM with the deep structure of auto-encoding to obtain better prediction performance. The classical ELM algorithm is a supervised learning algorithm. It is combined with auto-encoder to approximate the original input matrix by minimizing the reconstruction error. The structure of deep ELM is divided into two parts: ① unsupervised feature learning; ② supervised feature classification. In the first part, ELM auto-encoder structure is used to obtain the compressed representation of data layer by layer. In the second part, the classical ELM supervised learning algorithm is used for classification. ELM algorithm can make full use of the label information of samples, and extract the hidden information of data from the reduced samples, and improve the classification performance of the algorithm. The auto-coding structure here is different from the traditional deep auto-coding network. Once the features of the first hidden layer are extracted, the weight and parameters of the layer will be fixed, and there is no need to fine tune by back propagation as in the deep auto-coding network. Therefore, the training speed of deep ELM is faster than BPNN (back propagation neural network) and other back propagation algorithms. Compared with single-layer ELM, due to the dimension reduction training method of deep ELM auto-coding layer by layer, it can often achieve better test results in the face of high-dimensional data

B. Neighbor and spare representation

Neighborhood representation: the purpose of neighborhood preserving algorithm is to preserve the local neighborhood structure of data. For a data set, the weight matrix between each point is constructed firstly. For each data point, it can be expressed as a linear combination of its neighbors. The coefficient of linear combination is specified by the weight matrix. In the sample set $\boldsymbol{X} = [x_1, x_2, \cdots, x_m]$, for each sample x_i, its k nearest neighbor samples are used to reconstruct linearly. The objective function is as follow.

$$\min \left\| x_j - \sum w_{ij} x_j \right\| \tag{4}$$

where j=1, 2, ⋯ , k, k is the number of nearest neighbor. The k nearest neighbors can be obtained by calculating the Euclidean distance between different samples; w_{ij} is the nearest neighbor representation coefficient.

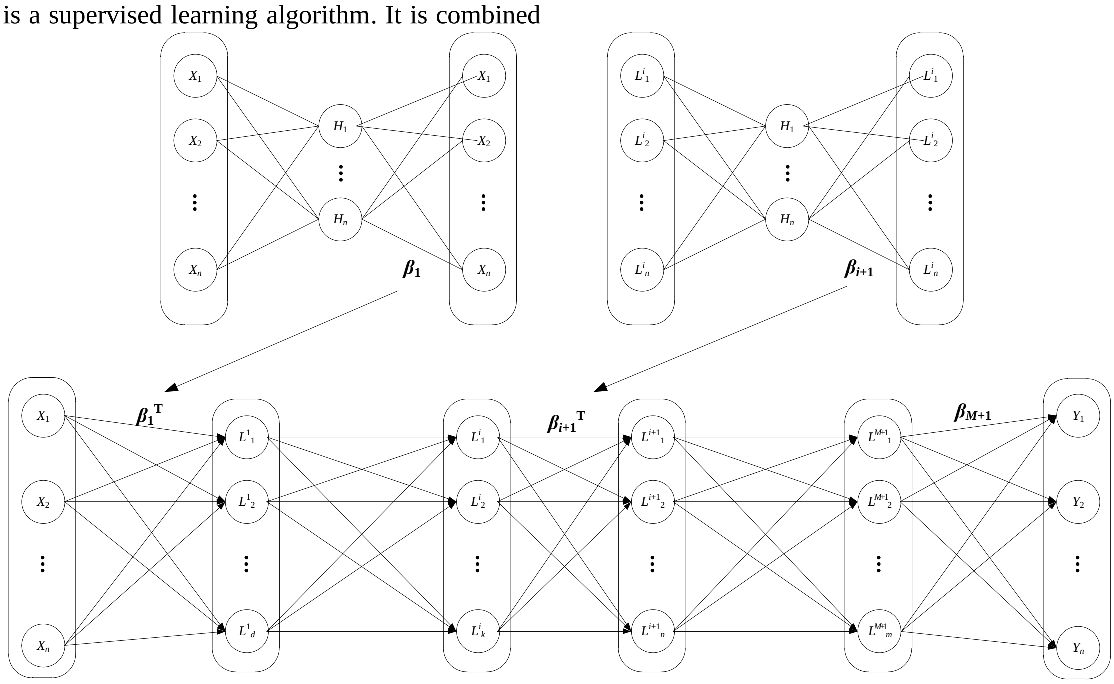


Fig. 1. Deep ELM network structure

Sparse representation: in the sample set $\boldsymbol{X}=\left[x_1,x_2,\cdots,x_m\right]$, the sparse representation expects each sample x_i to be linearly reconstructed by as few vectors as possible in the whole space. Due to the interference of noise and the problem of mathematical solution, the reconstruction error e is introduced. The objective function is as follow.

$$\begin{cases}\min\left\|q_j\right\|_0 \\ \text{s.t.}\left\|x_j-Xq_j\right\|_2<e\end{cases} \tag{5}$$

where q_j is sparse reconstruction vector. $1=\boldsymbol{I}^{\mathrm{T}}q_j$, $\boldsymbol{I}$ is the column vector whose element is 1.

To solve the above problem, orthogonal matching pursuit algorithm is used. The steps are as follows:

(1) Initializing the residual signal $r = x_j$, non-zero element set $\Lambda = \{\}$, iteration number $j = 0$;

(2) All the signals in the data set $\boldsymbol{X}$ are searched to find the coordinate k_c of the signal which has the greatest correlation with the residual signal r.

$$k_c=\arg\max\left|\boldsymbol{X}_k r\right| \tag{6}$$

(3) The k_c is merged into the coordinate set $\Lambda = \Lambda \cup \{ k_c \}$ corresponding to the non-zero element;

(4) The original observation signal x_j is orthogonally projected to the current set of non-zero elements, and the projection coefficient is calculated as follow

$$z_{\wedge}=\arg\min\left|x_j - \boldsymbol{X}_{\wedge}q_{\wedge}\right| \tag{7}$$

where $\boldsymbol{X}_{\wedge}$ is the set of signals in the original data set corresponding to the coordinates in the set Λ; $q_{\wedge}$ is the current non-zero element.

(5) Updating residual signal

$$r=x_j - \boldsymbol{X}_{\wedge}q_{\wedge} \tag{8}$$

where iteration number $j = j + 1$.

C. NS-DELM

In auto-encoding structure of deep extreme learning machine, a traditional single hidden layer feedforward neural network structure is constructed by taking the input as the output. The unsupervised learning is transformed into the supervised learning mode, which can be obtained from the sparse representation and the nearest neighbor representation. For training set $\boldsymbol{X}=\left[x_1,x_2,\cdots,x_m\right]$, there are nearest neighbors representation coefficient vectors $\boldsymbol{w}_j=\left[0,w_{j,1},\cdots,w_{j,k},\cdots,0\right]$ and sparse representation vector $\boldsymbol{s}_j=\left[s_{j,1},\cdots,s_{j,j-1},0,s_{j,j+1},\cdots,s_{j,m}\right]$.

$$\boldsymbol{x}_j=w_{j,1}x_{k,1}+w_{j,2}x_{k,2}+\cdots+w_{j,k}x_{k,k} \tag{9}$$

$$\boldsymbol{x}_j=s_{j,1}x_1+s_{j,2}x_2+\cdots+s_{j,m}x_m \tag{10}$$

where $x_{k,1},x_{k,2},\cdots,x_{k,k}$ is k nearest neighbors of x_j, $w_{j,1},w_{j,2},\cdots,w_{j,k}$ is the corresponding weight coefficient.

To keep the sparse reconstruction relationship and the nearest neighbor relationship in the original high-dimensional space into the projected low dimensional subspace, a spare and neighbor representation based deep ELM structure is proposed, whose objective function is as follow.

$$\begin{aligned}&\min_{\beta}0.5\|\boldsymbol{\beta}\|_F^2+0.5\lambda\sum\left\|\left(h(x_j)\boldsymbol{\beta}\right)^{\mathrm{T}}-\right.\\&\left.\left(H(\boldsymbol{X})\ \boldsymbol{\beta}\right)^{\mathrm{T}}\left(\alpha \mathrm{w}_j+\delta s_j\right)\right\|_2^2+\\&0.5C\left\|\boldsymbol{T}-H(\boldsymbol{X})\ \boldsymbol{\beta}\right\|_2^2\end{aligned} \tag{11}$$

where α and δ is the weight coefficient of nearest neighbor and sparse separately. $\boldsymbol{T}$ is the label matrix of data set.

The purpose of the second item is to keep the sparse and neighbor structure of the data after ELM projection. To solve formula (8), let $\boldsymbol{Z}=\alpha\boldsymbol{W}+\delta\boldsymbol{S}$, then we can obtain

$$\begin{aligned}&\sum\left\|\left(h(x_j)\boldsymbol{\beta}\right)^{\mathrm{T}}-\left(H(\boldsymbol{X})\ \boldsymbol{\beta}\right)^{\mathrm{T}}z_j\right\|_2^2=\\&\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}(\sum(h(x_j)^{\mathrm{T}}-H(\boldsymbol{X})^{\mathrm{T}}z_j)\bullet\\&(h(x_j)^{\mathrm{T}}-H(\boldsymbol{X})^{\mathrm{T}}z_j)^{\mathrm{T}}\boldsymbol{\beta})\end{aligned} \tag{12}$$

Let $\boldsymbol{e}_i$ be an n-dimensional unit vector, then we have

$$\begin{aligned}&\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}(\sum(\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}e_j-H(\boldsymbol{X})^{\mathrm{T}}z_j)\bullet\\&(\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}e_j-H(\boldsymbol{X})^{\mathrm{T}}z_j)^{\mathrm{T}}\boldsymbol{\beta})=\\&\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}(\sum(e_j-z_j)(e_j-z_j)^{\mathrm{T}})\bullet\\&H(\boldsymbol{X})\ \boldsymbol{\beta})=\\&\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\bullet\sum(e_je_j^{\mathrm{T}}-z_je_j^{\mathrm{T}}-e_jz_j^{\mathrm{T}}+\\&z_jz_j^{\mathrm{T}})\bullet H(\boldsymbol{X})\ \boldsymbol{\beta})=\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\bullet\\&(\boldsymbol{I}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}}+\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z})H(\boldsymbol{X})\ \boldsymbol{\beta})\end{aligned} \tag{13}$$

The above model (14) is changed as follow

$$\begin{aligned}&\min_{\beta}0.5\|\boldsymbol{\beta}\|_F^2+0.5\mathrm{tr}(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\bullet\\&(\boldsymbol{I}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}}+\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z})H(\boldsymbol{X})\ \boldsymbol{\beta})+\\&0.5\left\|\boldsymbol{X}-H(\boldsymbol{X})\ \boldsymbol{\beta}\right\|_2^2\end{aligned} \tag{14}$$

Let $\boldsymbol{B}=(\boldsymbol{I}-\boldsymbol{Z}-\boldsymbol{Z}^{\mathrm{T}}+\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z})$, the partial derivative of the objective function to $\boldsymbol{\beta}$ is obtained.

$$\begin{aligned}&\nabla=\boldsymbol{\beta}+\delta\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{BH}(\boldsymbol{X})\boldsymbol{\beta}+\\&C\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{-1}C\boldsymbol{H}^{\mathrm{T}}\boldsymbol{X}\end{aligned} \tag{15}$$

Let $\nabla=0$, then we have

$$\begin{aligned}&\boldsymbol{\beta}=\boldsymbol{I}+\delta\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{BH}(\boldsymbol{X})+\\&C\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{X})^{-1}C\boldsymbol{H}^{\mathrm{T}}\boldsymbol{X}\end{aligned} \tag{16}$$

When the number of sample data is less than the number of hidden neurons, the number of rows of $H(x)$ is greater than the number of columns. This leads to the problem of underdetermined least squares, as a result, $\boldsymbol{\beta}$ will have countless solutions. In this condition, let $\boldsymbol{\beta}=\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\sigma$, the left and right sides of $\nabla=0$ are multiplied by $(\boldsymbol{H}(\boldsymbol{X})\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}})^{-1}\boldsymbol{H}(\boldsymbol{X})$. Then the solution is obtained.

$$\begin{aligned}&\boldsymbol{\beta}=\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}\sigma=\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}(\boldsymbol{I}+C\boldsymbol{H}(\boldsymbol{X})\bullet\\&\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}}+\delta\boldsymbol{BH}(\boldsymbol{X})\boldsymbol{H}(\boldsymbol{X})^{\mathrm{T}})^{-1}C\boldsymbol{X}\end{aligned} \tag{17}$$

III. Context Sentiment Classification Process Based on NS-DELM

The context sentiment classification process mainly contains two stages: context sentiment prediction model training stage and context sentiment predicting stage.

(1) Context sentiment model training

For model training, the context sentiment classification training set is $\{\boldsymbol{X},\boldsymbol{T}\}$, where $\boldsymbol{X}$ is training samples, $\boldsymbol{T}$ is sample labels.

① Determine the whole network structure of deep network, i.e., the number of network layers n, the number of neurons in each hidden layer and the parameters k, α, γ, δ and C of each layer;

② Normalize the training data and test data;

③ Initialize the input weights for the first layer to (n-1) layer. Take the input as the output, and calculate the sparse and nearest neighbor matrices of each layer according to equations (4) and (5). Then calculate the output weight matrix $\boldsymbol{\beta}$ through equations (14);

④ For the nth layer, according to the label matrix, the output weights of the last layer are obtained by solving the least square.

(2) Context sentiment predicting

The unknown test data samples $\boldsymbol{X}'$ are segmented, and then the context sentiment classifier model trained by machine learning is used to predict. Then the context sentiment classification results are obtained. Then the feasibility of the classifier is measured according to the classification criteria.

IV. Experimental results and analysis

A. Experimental dataset and evaluation method

1 Experimental dataset

English corpus from four fields are used in this experiment: DVD comment, book comment, electronics comment, kitchen comment. All the comment texts are from English comments on Amazon shopping website. The emotional orientation of each text is manually marked by experts, where each emotional orientation is positive (+ 1) or negative (- 1). There are 1000 positive and 1000 negative texts in each field, shown in table 1. The features of each text are in the form of unitary and binary mixed grammar, such as "even_enjoy", "ruined", "could_not", "you_like". The training set consists of 800 positive texts and 800 negative texts, namely a total of 1600 texts. The test set consisted of 200 positive texts and 200 negative texts, namely a total of 400 texts.

2 Evaluation method

In this paper, accuracy Acc is used to evaluate the results of sentiment classification. Accuracy is a commonly used measure, which takes into account the correct probability of positive and negative categories. It can reflect the overall classification effect of the two categories

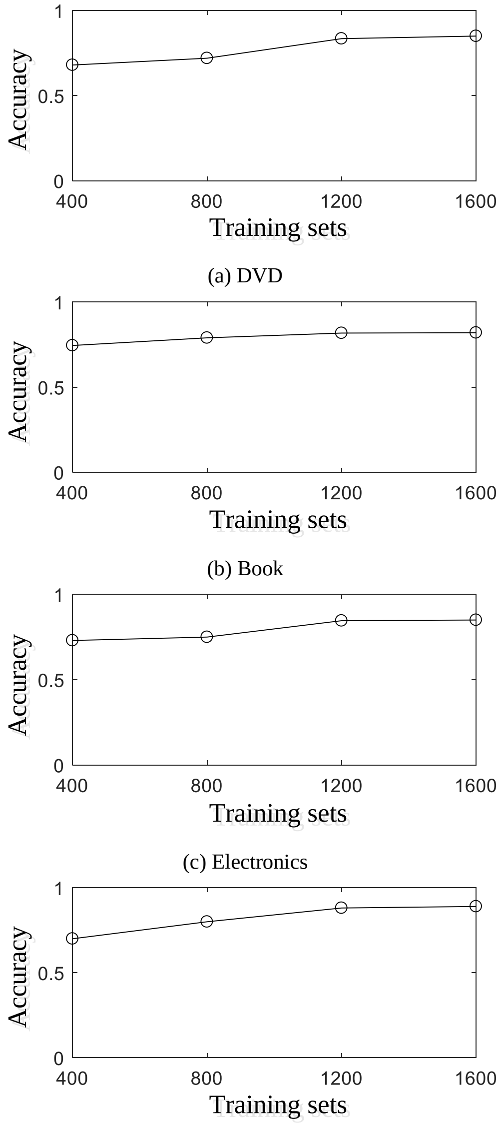
$$Acc = \frac{N_{\mathrm{TP}} + N_{\mathrm{TN}}}{N_{\mathrm{TP}} + N_{\mathrm{TN}} + N_{\mathrm{FP}} + N_{\mathrm{FN}}} \tag{18}$$

where N_{TP} is the number of positive comments predicted to be positive; N_{FN} is the number of negative comments predicted to be negative; N_{FP} is the number of negative comments predicted to be positive; N_{TN} is the number of negative comments predicted to be negative.

B. Training set scale experiment

In order to test the influence of different size training sets on the accuracy of emotion classification, 1/4 training sets (400 training sets), 1/2 training sets (800 training sets), 3/4 training sets (1200 training sets) and 1/1 training sets (1600 training sets) are selected for experiments separately. The classification accuracy of the 4 testing set under different size of training sets are shown in figure 2.

The experimental results show that the scale of training set has an impact on the result of emotion classification. When the scale of training set is small, the larger the scale of training set, the higher the accuracy of classification. However, when the size of training set reaches a certain value, the accuracy will be improved little or no longer with increasing the scale of training set. As a result, the size of training set can be expanded to improve the accuracy. All the training data are used in the following experiments.



(a) DVD

(b) Book

(c) Electronics

(d) Kitchen

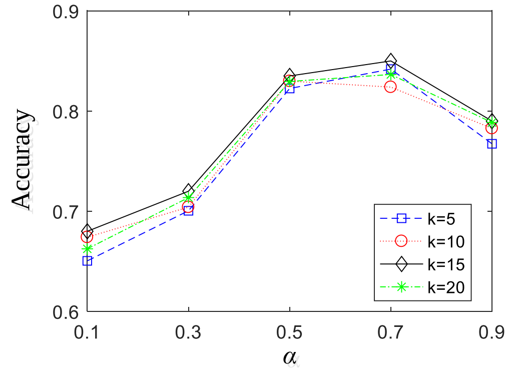
Fig. 2. The classification accuracy under different size of training sets

TABLE I
EXPERIMENTAL DATA

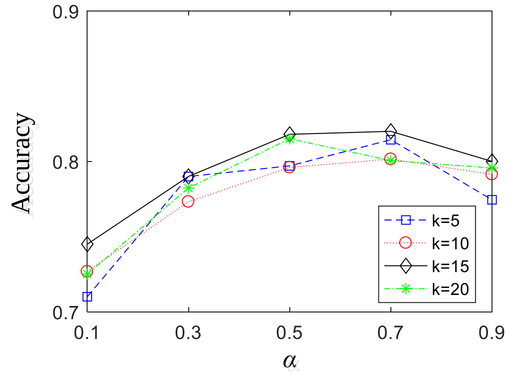
Corpus	Total number	Positive number (+ 1)	Negative number (- 1)	Training set number	Testing set number
2000	1000	1000	1600	400	2000
2000	1000	1000	1600	400	2000
2000	1000	1000	1600	400	2000
2000	1000	1000	1600	400	2000

C. *Influence of parameters α and k on NS-DELM*

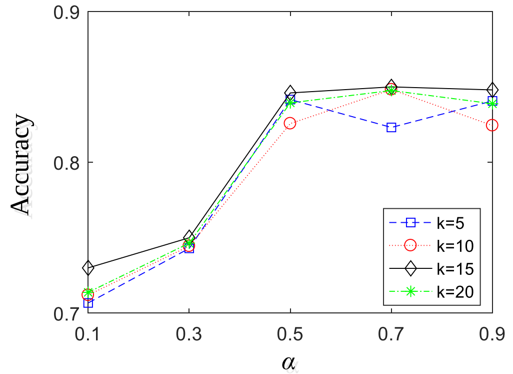
The different parameter values of α (neighbor weight coefficient) and k (the number of neighbor) have a certain influence on the results of classification accuracy. In this experiment, let $\alpha = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $k = \{5\ 10\ 15\ 20\}$, the classification accuracy under different parameter values is shown in figure 3.



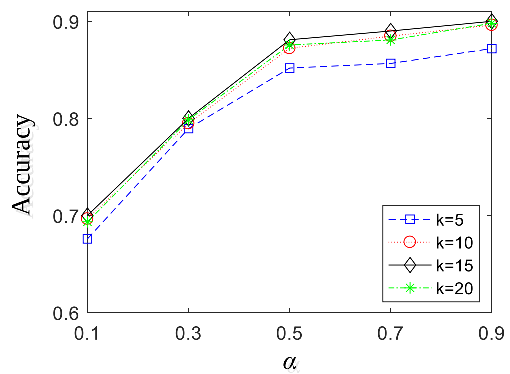
(a) DVD



(b) Book



(c) Electronics



(d) Kitchen

Fig. 3. Classification accuracy under different value of α and k

In general, with the increase of α value, although the average accuracy of electronics and kitchen is relatively stable, it generally shows a trend of first improving and then decreasing, and the effect is better when α value is 0.7 or 0.9. With the increase value of k, the classification accuracy changes from low to high and then to low. When k value is 15, the classification accuracy is the highest. From this experiment , the best combination of α and k value for each field data is obtained, as shown in Table 2. In subsequent experiments, the corresponding combination of α and k value in Table 2 is used for different fields.

TABLE II
BEST COMBINED VALUES OF α AND k FOR THE FOUR DOMAINS

Corpus	α	k
DVD	0.7	15
book	0.7	15
electronics	0.7	15
kitchen	0.9	15

D. *The comparative experiment of different emotion classification methods*

The classification performance of the proposed NS-DELM is compared with conventional support vector machine (SVM), stacked auto-encoder (SAE) and convolutional neural network (CNN). SVM is single layer classification model, while SAE and CNN are two multiple layer classification model. The comparison results of the four methods are shown in Table 3 and figure 4.

TABLE III
THE COMPARISON CLASSIFICATION RESULTS OF THE FOUR METHODS

Corpus	SVM	SAE	CNN	NS-DELM
DVD	0.8250	0.8400	0.8350	0.8500
book	0.8025	0.8150	0.8125	0.8200
electronics	0.8225	0.8450	0.8475	0.8550
kitchen	0.8675	0.8825	0.8850	0.8900

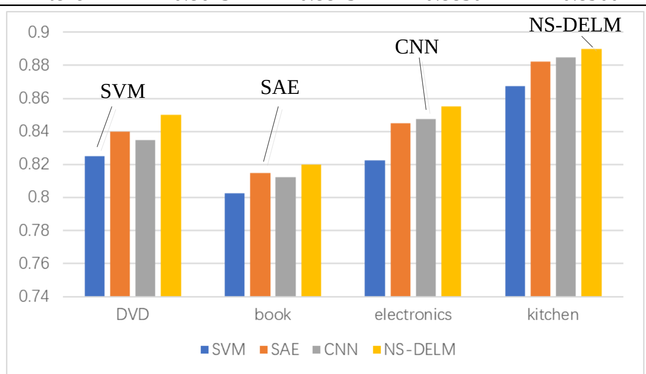


Fig. 4. The classification results of the four methods

Seen from table 3 and figure 4, for DVD, book, electronics and kitchen, the classification results of kitchen is obviously better than other domain, and their highest classification accuracy can reach 0.8500, 0.8200, 0.8550 and 0.8900 respectively. This shows that text emotion classification is a domain related problem. For all the domain of context sentiment classification, other methods are better than SVM. This shows that the text classification based on deep learning strategy is helpful to improve the classification accuracy. Compared with SAE and CNN, the average accuracy of NS-DELM is improved, which indicates that the effect of using spare and neighbor representation strategy is better than traditional deep learning model. For DVD corpus, the classification result of NS-DELM is increased by 0.01 compared with SAE. For book corpus, the classification result of NS-DELM is increased by 0.005 compared with SAE. For electronics corpus, the classification result of NS-DELM is increased by 0.0075 compared with CNN. For kitchen corpus, the classification result of NS-DELM is increased by 0.0050 compared with CNN.

V. CONCLUSIONS

This paper has addressed a novel context sentiment classification method using deep extreme learning machine with the nearest neighbor and sparse representation. By using self coding, the input is taken as the output, and the structure of deep network is generated. Unsupervised feature extraction is carried out layer by layer. Finally, classification is carried out by classical extreme learning machine. The idea of sparsity and nearest neighbor is integrated into deep extreme learning machine. By optimizing the solution model, the spatial manifold structure of data can be maintained in the process of layer by layer feature extraction, so as to improve the dimension reduction performance of deep extreme learning machine and improve the accuracy of algorithm classification. Experiment of comments on Amazon shopping website is utilized to evaluate the proposed method performance. Test results indicate that the proposed sentiment classification capability is better than other conventional models.

REFERENCES

[1] V. V. Asch and W. Daelemans. “Predicting the Effectiveness of Self-Training: Application to Sentiment Classification”, 2016.

[2] L. Shang, Z. Zhou and X. Liu. “Particle swarm optimization-based feature selection in sentiment classification”, *Soft Computing*, pp.1-14, 2016.

[3] M. A. Masood , R. A. Abbasi and N. W. Keong. “Context-Aware Sliding Window for Sentiment Classification”, *IEEE Access*, pp.1-1, 2020.

[4] K. Shuang , M. Gu and R. Li. “Interactive POS-aware network for aspect-level sentiment classification”, *Neurocomputing*, vol.420, pp.181-196, 2021.

[5] N. F. F. D. Silva, L. F. S. Coletta and Eduardo R. Hruschka. “Using unsupervised information to improve semi-supervised tweet sentiment classification”, *Information Sciences*, 2016.

[6] B. Pang, L. Lee and S. Vaithyanathan. “Sentiment Classification using Machine Learning Techniques”, *Proceedings of Emnlp*, pp.79—86, 2002.

[7] P. Wang, M. Gao and S. Xiao. “Unsupervised linguistic approach for sentiment classification from online reviews using SentiWordNet 3.0”, *Advanced Materials Research*, pp.1560-1564, 2014.

[8] Z. Lin, X. Jin and X. Xu. “An Unsupervised Cross-Lingual Topic Model Framework for Sentiment Classification”, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol.24, no.3, pp.432-444, 2016.

[9] Z. Dongwen, X. Hua and S. Zengcai. “Chinese comments sentiment classification based on word2vec and SVM perf”, *Expert Systems with Applications*, vol.42, no.4, pp. 1857-1863, 2015.

[10] G. Vinodhini, R. M. Chandrasekaran. “A comparative performance evaluation of neural network based approach for sentiment classification of online reviews”, *Journal of King Saud University - Computer and Information Sciences*, vol.28, no.1, pp.2-12, 2015.

[11] H. Guang-bin, Z. Qin-yu and S. C. Kheong. “Extreme learning machine: Theory and applications”, *Neurocomputing*, vol.70, pp. 489-501, 2006.

[12] P. Feng, Z. Hai-bo. “Online Sequential Extreme Learning Machine Based Multilayer Perception with Output Self Feedback for Time Series Prediction”, *Journal of Shanghai Jiaotong University*, vol.18, no.3, pp.366-375, 2013.

[13] A. Lendasse, H. Qing and M. Yoan. “Advances in extreme learning machines (ELM2012)”, *Neurocomputing*, vol.128, pp.1-3, 2014.

[14] S. Zhan-li, W. Han and Lau Wai-shing. “Application of BW-ELM model on traffic sign recognition”, *Neurocomputing*, vol.128, pp. 153-159, 2014,.

[15] Y. Xu, C. Yan-jing and Z. Qun-xiong. “An extension sample classification-based extreme learning machine ensemble method for process fault diagnosis”, *Chemical Engineering & Technology*, vol.37, no.6, pp.911-918, 2014.

[16] K. Javed, R. Gouriveau and N. Zerhouni. “SW-ELM: A summation wavelet extreme learning machine algorithm with a priori parameter initialization”, *Neurocomputing*, vol.123, pp.299-307, 2014.

[17] H. Yang, B. Zeng. “Enhancing Fine-grained Sentiment Classification Exploiting Local Context Embedding”, 2020.

[18] Z. Wu, Y. Li and J. Liao. “Aspect-Context Interactive Attention Representation for Aspect-Level Sentiment Classification”, *IEEE Access*, pp.1-1, 2020.

[19] J. Zhao, Y. Yang and G. Pang. “Learning Discriminative Neural Sentiment Units for Semi-supervised Target-Level Sentiment Classification”, *Neurocomputing*, 2020.

[20] K. P. Lai, W. Lam and J. C. S. Ho. “Domain-Aware Recurrent Neural Network for Cross-Domain Sentiment Classification”, *DSIT 2020: 2020 3rd International Conference on Data Science and Information Technology*, 2020.

Liu Ronghui acts as an associate professor in the School of Computer and Data Science, Henan University of Urban Construction. His interest is neural network and database technology.

Zhang Jingpu is a lecturer in the School of Computer and Data Science, Henan University of Urban Construction. His interest includes computing technology and application.