Research on correlation of alarm data of refining and chemical plant based on optimized CBOW method

Jinqiu Hu[*1,2], Qin Chen[1,2], and Zhenqiang Wei[3]

[*1]College of Safety and Ocean Engineering, China University of Petroleum (Beijing), Beijing, China
[2]Key Laboratory of oil and gas production safety and emergency technology by Ministry of Emergency Management of the People's Republic of China
*E-mail: hujq@cup.edu.cn
[3]CNPC safety and Environmental Protection Research Institute, Huanghe North Street, Changping District, Beijing

Abstract: As the refining equipment gradually becomes more complex and intelligent, more and more alarm points need to be set in the industrial process, However, the functions of some alarm points are repetitive and unnecessary, and some alarm thresholds are set unreasonably, which will lead to a large number of alarm prompts in a short period of time and interfere with judgment of operator, and brings challenges to the alarm management. Therefore, it is necessary to analyze the alarm log and optimize the alarm management system. In order to process the alarm information more conveniently and quickly, and improve the efficiency of alarm log information analysis. This paper introduces the CBOW model and negative sampling optimization method commonly used in natural language processing to embed the alarm log, embed the alarm information, retain the correlation of the alarm information in time and semantics, calculate the Euclidean distance after the dimensionality reduction of high-dimensional vectors, and obtain the clustering results, which provides help for optimizing the process alarm point setting and alarm threshold setting. Experimental results show that the proposed method can retain the correlation of alarm data in time and semantics, and effectively identify alarms with high correlation.

Keywords: alarm management; Word embedding; Clustering; correlation analysis

1. INTRODUCTION

At present, the monitoring and maintenance of refining and chemical units in major domestic oil refining enterprises are developing towards unmanned, centralized and intelligent direction, and alarm systems are widely used to monitor operating units and equipment. Inform the operator of the problems through acoustic signal, optical signal, text, etc., so that he can handle the abnormal conditions in time. The alarm system is a combination of software and hardware for monitoring the alarm status. It is responsible for transmitting the problems to the operator and recording the changes of the alarm status [1]. The performance of alarm management system will directly affect the production safety and production efficiency of refining and chemical enterprises.

Alarm management system plays an important role in the alarm and monitoring of refining and chemical enterprises, but it also faces many problems. With the wide application of distributed control system (DCS) in enterprises, the acquisition of alarm data becomes more convenient. When a large amount of alarm information continues to flow to operators, it will not only occupy a large amount of storage space, but also make key alarm information submerged in the flood of alarm. On the other hand, due to the complexity of modern industry, many equipment and processes are not independent, but there are upstream and downstream related relationships. When an equipment fails and sends an alarm message, the alarm probability related to it will follow. Rothenberg [2] defines this related alarm as related alarm. According to the survey and statistics, the industrial enterprises suffer losses of more than 100 million yuan each year due to untimely alarm handling and neglect of key alarms. For example, in the explosion and fire accident of the unit in Texaco refinery in the UK in 1994, it was precisely because the number of alarms was too large that the operator could not grasp the actual situation, and the key alarms were not handled, which ultimately led to the explosion. In particular, 275 alarms flooded in the last 11 minutes before the explosion, making the operator unable to make a correct response [3].

Therefore, in order to process alarm information more conveniently and quickly, it is of practical engineering significance to mine effective information from a large number of alarm data and analyze the correlation between alarm information. The traditional log data analysis method is time-consuming, laborious, heavy workload, and easy to ignore useful information. Therefore, the rich information of the alarm log is combined with the natural language processing algorithm in the computer field to study the alarm correlation and improve the processing efficiency and effect of large sample size text.

2. THEORIES OF WORD EMBEDDING

2.1. Word embedding

In the field of natural language processing, text, as unstructured character data, needs to be converted into computable numerical data. There are three kinds of text processing methods using natural language processing. One is the grammar level task of word segmentation, part of speech tagging and named entity recognition; Second, word embedding is word vector, which maps words to vector space; The third is sentence vector, which maps sentences to vector space. Word embedding is a key technology in natural language processing. Word embedding mainly includes discrete representation and neural network representation.

The discrete representation includes one hot coding and co-occurrence matrix. One hot coding uses a string of very long vectors to represent a word. The length of the vector is the total number of words. The component of the vector is only one "1", and the others are all "0". The position of "1" corresponds to the position of the word in the dictionary. The one hot coding program is relatively simple and easy to implement, but there are still limitations in practical use. First, the vector generated by one hot coding does not contain semantic information, and words are isolated from each other, which cannot represent the relationship between words; Second, one hot coding, as a sparse vector, will cause a dimension disaster when the vocabulary is particularly large [4]. The co-occurrence matrix method counts the number of occurrences of each word as a neighboring word to obtain a symmetrically distributed co-occurrence matrix, where each row corresponds to one word. The co-occurrence matrix can better represent the relationship between words, but it is only applicable to the case where there are few texts. When there are too many texts, the vector length of each word surges and the matrix dimension explodes.

The principle of neural network representation is to represent a word through its context. The encoded text can retain the meaning of the text to the maximum extent. The distance between words is represented by Euclidean distance of vectors. However, when the text corpus is large, the amount of calculation will become very large. After many years of research, Mikolov et al. [5] proposed word2vec algorithm to calculate the distributed representation of feature words in text. The feature words are expressed as a low dimensional vector by using the relationship between feature words and context. This vector contains semantic information and can represent the relationship between contexts. The optimized word2vec algorithm can capture the word order relationship between words at the level of word vector representation, and has a low computational load and vector dimension. After years of development, word2vec algorithm is one of the most widely used word embedding algorithms [6].

2.2. Model and model

(1) Word2vec model

Word2vec includes the training models of Skip-Gram and CBOW (continue bag of words). The goal of CBOW is to predict intermediate words through context words, which is suitable for small data processing. The structure is shown in Figure 1.

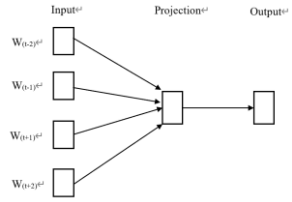


Fig.1 CBOW model structure diagram

Skip-Gram model predicts the context through intermediate words, and performs better in large-scale training corpus. The Skip-Gram model structure is opposite to the CBOW model. The input layer is a specific word vector, and the output layer is the context word vector corresponding to the specific word vector, as shown in Figure 2.

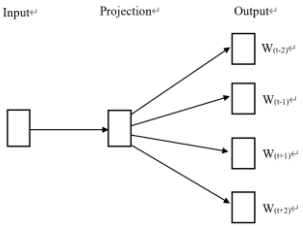


Fig.2 Skip-Gram model structure diagram

Each time the Skip-Gram model is executed, the current word is sequentially extracted from the text, the vector of the candidate word is taken as the model input, and the output layer predicts the context content corresponding to the current word.

(2) Optimization method

In the process of training, CBOW and Skip-Gram models need to constantly update the input and output. From the perspective of calculation, when the corpus is too large, the calculation time is long and the training is difficult. Therefore, two optimization schemes, hierarchical softmax and negative sampling, are introduced to improve the model performance. Hierarchical softmax is to construct a binary tree from top to bottom according to the word frequency of words in the text, and transform the N classification problem into a sub classification problem. Negative sampling allows one training sample to update only a part of the weight at a time, and all other weights are fixed. This can reduce the amount of calculation, increase randomness to a certain extent, and reduce the loss value [7].

In this paper, the CBOW model is used to vectorize the alarm, and the negative sampling (NS) method is used to optimize the model. This combination has the characteristics of fast operation speed.

3. CORRELATION ANALYSIS OF ALARM DATA BASED ON CBOW METHOD

3.1. Data preprocessing

In the alarm management system, the alarm information is stored in the database to form an alarm log.

The alarm data in the table includes alarm message, source, condition name, source time, occurrence area and other attributes. Each row represents an alarm occurrence. The source, condition name and source time jointly identify the alarm, indicating the location, type and time of the alarm. The alarm types include LL, LO, HH, HI etc.

The word vector model is implemented by using the CBOW model proposed in Section 2.2 and the negative sampling optimization method. The specific steps of the optimized CBOW algorithm are as follows:

(1) The alarm time series is cut according to a certain time threshold, and the sentences in natural language processing are compared, and the alarms with a relatively short time are divided into one alarm sentence.

(2) Set parameters such as embedding dimension and window size. The gensim toolkit in Python is used for word embedding modeling and training based on tensorflow 2.0 and keras framework. The divided alarm sentence is regarded as a sentence composed of words as the input of the model. The hyperparameters of the model can be balanced between the computational requirements and the training results.

(3) The optimized CBOW algorithm is used to learn the semantic information of the alarm sentence, and the vector represents the feature information of each alarm variable in the alarm sentence.

3.2. Vector dimension reduction processing

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Number equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop. To make your equations more compact, you may use the solidus (/), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols.

The dimension of distributed vectors trained by CBOW method is high, and the spatial distance between vectors cannot be seen intuitively. Therefore, the t-SNE dimension reduction method (t-distributed stochastic neighbor embedding, t-distributed random neighborhood embedding) is selected to reduce the dimension of the vector in the high-dimensional space under the condition that the alarm semantics and time sequence information are less lost.

t-SNE is a nonlinear dimension reduction algorithm based on random neighbor embedding, which converts Euclidean distance in high-dimensional space into conditional probability representing similarity [8]. By constructing a probability distribution between high-dimensional spatial data, more similar data have a higher probability of being selected, while the object data with large differences are excluded. The heavy tailed t distribution is used to construct a probability distribution in the low dimensional space, so that the probability distribution in the high dimensional space and the position space are as similar as possible. The conditional probability between two points of data in the high-dimensional space $P_{j|i}$ is expressed as:

$$P_{j|i} = \frac{\exp(-\frac{\| x_i - x_j \|^2}{2\sigma_i^2})}{\sum_{k \neq i} \exp(-\| x_i - x_k \|^2 / 2\sigma_i^2)} \quad (1)$$

Where, x_i, x_j represents two data points in the high-dimensional space, and represents the Gaussian variance centered on i.

Let the mapping of high-dimensional data x_i, x_j in low-dimensional space be y_i, y_j, and calculate the conditional probability $q_{j|i}$as:

$$q_{j|i} = \frac{\exp(-\| y_i - y_j \|^2)}{\sum_{k \neq i} \exp(-\| y_i - y_k \|^2)} \quad (2)$$

To ensure that the similarity and distance between the data can be preserved after mapping to the low dimensional space, t-SNE adopts gradient descent method to optimize the KL divergence of all objects. The objective function T and the optimization method are defined as:

$$\bar{T} = KL(P \| Q) = \sum_{i,j} p_{j|i} lb \frac{p_{j|i}}{q_{j|i}} \quad (3)$$

$$\frac{\delta T}{\delta y_i} = 4 \sum_j (p_{j|i} - q_{j|i})(y_i - y_j)(1 + \| y_i - y_j \|^2)^{-1} \quad (4)$$

3.3. cluster analysis

(1) K-means clustering

K-means clustering analysis calculates the distance between each object. First, K alarm variables are randomly selected as the initial cluster center, and the distance between other alarms and the cluster center is calculated. According to the calculation results, the alarms with relatively close distance are classified into this category. By continuously calculating the distance between each alarm and the cluster center in each category, the average value of the distance is taken as the new cluster center, and the distance between other alarms and the cluster center is calculated again to divide the

categories until the cluster center reaches the convergence state. The calculation formula is:

$$\text{center} = \frac{1}{n_g}\sum_{i \subset g} d_{ig} \qquad (5)$$

where center is the new cluster center and d_{ig} is the distance between each alarm in each cluster and the cluster center. n_g is the total number of alarms in the class g.

(2) Agglomerative Nesting

Agglomerative clustering (Agnes) algorithm is a bottom-up agglomerative hierarchical algorithm. It takes each initial object as a cluster, merges the cluster with the smallest distance among two objects belonging to different clusters by calculating the distance between each object, and merges the cluster with the closest distance by continuously calculating the distance until the set termination condition is reached, Or all objects are merged [9]. After the midpoint of two-dimensional space is determined, the alarm vector can be regarded as numerical data, and the Euclidean distance is used as the correlation measurement index between vectors. The Euclidean distance calculation formula between i, j two vectors is shown in eq.(6):

$$\text{dist}_{ij} = \sqrt{(i_1 - j_1)^2 + (i_2 - j_2)^2} \qquad (6)$$

The specific steps of cluster analysis are as follows:

(1) 20 variables are randomly selected from the alarm variables with more than 10 alarm times. For convenience, each alarm is represented by $A_i (i = 0,1,2,\dots,n)$. Each alarm is regarded as a class, the Euclidean distance is calculated by the formula, and the distance matrix between classes is established.

(2) In the distance matrix, find the two alarm $A_m$and A_n which have the closest distance and the largest degree of correlation, and merge the two classes into a new class A_{mn}.

(3) The distance matrix is updated by calculating the distance between the newly merged class A_{mn} and other individual classes.

(4) Repeat step 2 and step 3 until all alarms are grouped into one class or the number of clusters reaches the specified upper limit.

4. EVALUATING INDICATOR

(1)*Accuracy*

Word analogy is to set questions in the trained model, such as "which word corresponds to the word vector with the highest similarity to 'vector a'?", The answer is given according to the word contact information contained in the word vector. If the word given by the model is consistent with the preset word, the answer is correct. If the word calculated by the model is inconsistent with the preset word, the answer is wrong. Set the part of speech ratio task as: what are the top three alarms related to alarm a. When there are correct answers in the three alarms answered, the answer is correct. In the part of speech ratio task, calculate how many questions are answered correctly. The proportion of the number of correct answers to the total number of questions is defined as the part of speech ratio accuracy rate. The higher the accuracy rate, the better the quality of the word vector. Calculation formula:

$$\text{Accuracy} = \frac{AR}{N} \qquad (7)$$

Where, Accuracy represents the accuracy rate of the word analogy, AR represents the number of correct answers in the part of speech ratio task, and N represents the total number of parts of speech ratio tasks.

(2) Similarity SIM

SIM is used to evaluate the similarity between two alarms. The value range is [0, 1]. When SIM value is 1, it indicates the highest similarity. First, according to the determined alarm groups, the similarity of alarms in the same group is compared. The higher the similarity in the same group, the higher the quality of the word vector obtained. The calculation formula is:

$$\text{Sim} = \frac{s_1 + s_2 + \cdots + s_n}{g} \qquad (8)$$

Wherein, s_n represents the similarity of each group of alarms and g represents the total number of alarm groups.

5. CASE STUDY

5.1. Results

The CBOW model is trained with gensim Library Based on Python 3.7. The alarm sequence is used as the sentence input model in natural language processing, and the negative sampling method is used for optimization. The embedding dimension selected for initialization parameters is 100, the minimum word frequency is 1, and the window size is 5. Other parameters are shown in Table 1.

Tab. 1 Model parameter selection

Parameter name	Parameter representation	Parameter value
embedding dimension	size	100
Window size	window	5
Model selection	sg	0
optimization method	negative	5
Minimum word frequency	min_count	1

After training, each alarm variable is mapped into a vector with a dimension of 100 and stored in a text file. Some alarm vectors are shown in Fig. 3.

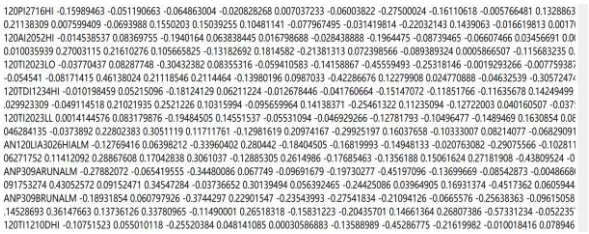


Fig.3 Partial alarm vector

40 kinds of alarms occurred in the wax oil hydrogenation unit in April are selected, and the vector is reduced to two dimensions by t-SNE method. The alarm variables and their specific descriptions are shown in Table 2(in the *appendix*). The visualization results are shown in Figure 4, where x and Y represent the horizontal and vertical coordinates of the two-dimensional coordinate system. For convenience, the results of dimension reduction and clustering are explained by the serial number of the alarm variable.

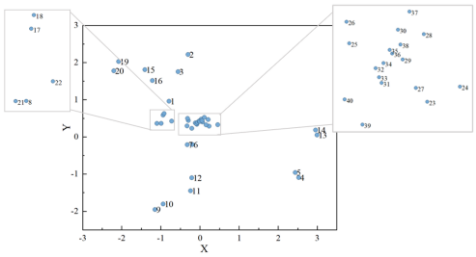


Fig.4 Visualization results and local enlargement based on t-SNE dimensionality reduction method

According to the distribution of the scatter plot, the number of clusters of 40 alarms is set to 12. On the basis of the scatter plot in Fig. 7, the K-means clustering method is fused, and 12 alarm clusters are formed through 20 iterations. It can be seen from the clustering results in Fig. 5 that the clustering effect is better, and the alarms closer to each other are classified into the same category, and the alarm categories are distinguished by different colors.

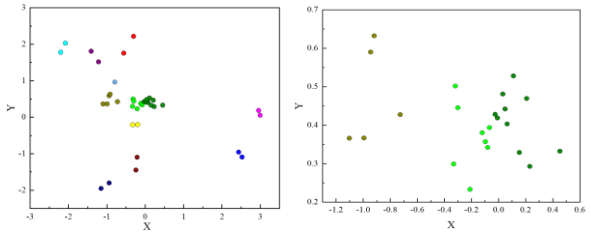


Fig.5 Visualization and local enlargement based on K-means clustering

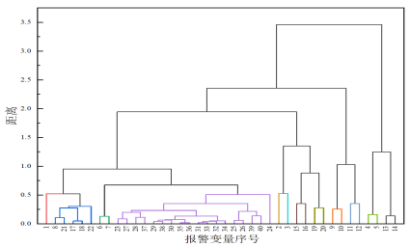


Fig.6 Agglomerative hierarchical clustering grouping results

It can be seen from Fig. 4 and Fig. 5 that there are obviously cases where related alarms are clustered around each other. The strong or weak correlation shows that the same type of alarm generated at the same device is close to each other in the scatter diagram, followed by the alarm from different directions of the same device. Finally, the alarm at different positions of the alarm is far away, and the correlation is also weaker.

For example, alarm 4 refers to the low liquid level alarm of light slop tank D301, alarm 5 refers to the low liquid level alarm of light slop tank D301, alarm 17 refers to the alarm indication of pump stop p309a, alarm 18 refers to the alarm indication of pump stop p309b, alarm 4 and alarm 5 both indicate the low liquid level alarm of light slop tank D301. Through analysis of the alarm data in April, it is found that the two alarms occur 123 times. Alarm 17 and alarm 18 are also pump stop alarms for 309 pump group, which have similar semantics. At the same time, these two alarms are also related in time. Alarm 17 and alarm 18 appear in succession as many as 311 times in one minute. The nine alarms from alarm 30 to alarm 38 are all sent out by the positions of the pipelines of furnace F101. They are flow control alarms and point to the high flow. The time interval in the alarm log is two minutes and the correlation is high. Alarm 9 and alarm 10 form an alarm category. Alarm 11 and alarm 12 are another group of alarms. The four alarms and other alarms are far away from each other and are all sent by the location at the aviation coal stripper C204. Alarm 9 is a high-level control alarm, and alarm 10 is a high-level control alarm; In the other group, the alarm 11 represents the liquid level control low-low alarm, and the alarm 12 represents the liquid level control low alarm, which is just opposite to the alarm 9 and the alarm 10.

The agglomerative hierarchical clustering method has mined the alarms with unclear correlation in the two-dimensional visualization diagram, as shown in Fig. 6. Among them, alarm 15 and alarm 16, alarm 19 and alarm 20 all indicate the start and stop alarms of the same pump, which are highly correlated. Their direct correlation is not given in the cluster scatter diagram. The higher level relationship between the start and stop alarms in the grouping belongs to the same category. Alarm 9, alarm 10, alarm 11 and alarm 12 are the high and low alarms for the level control of aviation kerosene stripper C204 and the high-high and low-low alarms of the interlocking system. Therefore, by combining the dimension reduction visualization method based on word2vec method with aggregation hierarchical clustering for association analysis, the association between alarm data can be more comprehensively mined.

Based on the results of clustering, optimization suggestions can be put forward for the alarm management system. If alarm 2 and alarm 3 are divided into one group, high and high alarm will be given when the flow PV of dry gas to PSA device is 18000Nm3 / h or above, and high alarm will be given when the flow PV of dry gas to PSA device is 17000Nm3 / h or above. Alarm 4 and alarm 5 are divided into the same group at a

relatively close distance. The alarm conditions are that the liquid level PV of light slop tank D301 is lower than 20%, the liquid level gauge sends a low-level alarm, and the alarm below 15% sends a low-level alarm. The design of the alarm threshold is relatively close, resulting in that the two alarms are basically sent out at the same time, affecting the judgment of the operator. Therefore, it is recommended to redesign the alarm threshold for this situation. Alarm 15 to alarm 22 are all start and stop alarms of the pump unit, which belong to system alarms. When the alarm is sent, the operator is not required to take measures, and it is only used as a prompt for system start and stop. Therefore, it is recommended to set partitions for such cases, move the system alarms into the partitions for display, and keep non-system-alarms in the main display area, which is conducive to the timely study and treatment of the operator.

5.2. Comparative analysis

Combining the two evaluation indexes of part of speech ratio accuracy and similarity, the method proposed in this paper is compared with the skip-gram model method of word2vec and the two optimization methods of negative sampling and hierarchical softmax. The calculated evaluation index values are shown in Table 3.

Tab. 3 Comparison of different word embedding methods

Different analysis methods	Accuracy	Sim
Skip-gram+ Negative sampling	81.82%	97.16%
Skip-gram+Hierarchical softmax	77.27%	92.82%
Methods in this paper	86.36%	98.25%
Maximum improvement effect	9.09%	5.33%

From table 3, the word embedding effects of different word embedding methods are ranked as follows: alarm data association analysis algorithm based on CBOW model and negative sampling optimization, Skip-gram model and negative sampling method, Skip-gram model and hierarchical softmax method. Different optimization methods are used in Skip-gram model, and the quality of word vector is different. The effect of using negative sampling method is obviously better than hierarchical softmax method. In this case, the accuracy of part of speech compared with the task was increased by 9.09%, and the similarity between alarms was increased from 92.82% to 97.16%. This indicates that the negative sampling optimization method is more suitable for this database. Compared with Skip-gram model method, the word embedding method based on CBOW model and negative sampling optimization has greatly improved in two indicators, the accuracy of part of speech compared with the task has increased by 9.09%, and the similarity of related alarms has increased by 5.33%. The better performance of accuracy and similarity shows that the word embedding processing using this method can better preserve the correlation between alarm data and improve the reliability and accuracy of clustering grouping results.

6. CONCLUSION

In this paper, a correlation analysis method for alarm data of refining and chemical equipment based on CBOW model and negative sampling optimization is proposed. The alarm is converted into variable form for clustering analysis. The clustering results provide help for optimizing the process alarm point setting and alarm threshold setting. From the evaluation indexes of part of speech ratio task accuracy and similarity, this method is superior to Skip-gram model and negative sampling optimization method, Skip-gram model and hierarchical softmax optimization method. The accuracy of part of speech is improved by 9.09% and the similarity of related alarms is increased by 5.33%., It is verified that the CBOW model combined with the negative sampling method can retain the alarm correlation to the maximum extent, and provide guidance for reducing the number of alarms and designing the alarm management system reasonably. In addition, the method of treating alarm sequences as sentences in natural language processing also provides a new idea for the research of alarm related fields.

Acknowledgements

The authors gratefully acknowledge the support from the Natural Science Foundation of China (52074323), PetroChina Strategic Cooperation Science and Technology Project (ZLZX2020-05-02).

REFERENCES:

[1] Zhu Qunxiong, Gao Huihui, Xu Yuan Research progress of industrial process alarm management [J] Journal of automation, 2017,43 (06): 955-968

[2] ROTHENBERG D. Alarm Management for Process Control[M]. New York: Momentum Press,2009.

[3] EEMUA, 2013, EEMUA-191: Alarm Systems-a guide to design, management and procurement, Engineering Equipment and Materials Users Association. London, U.K.

[4] Yan Hong Review on the development of word vector [J] Modern computer (Professional Edition), 2019 (08): 50-52.

[5] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient Estimation of Word Representations in Vector Space[J]. arXiv preprint: 1301.3781, 2013.

[6] Wang Ling Implementation of text classification based on word2vec word embedding and bidirectional long-term and short-term memory network [J] Electronic technology and software engineering, 2020 (15): 70-71.

[7] Wu Deping, Hua Gang Text case classification of work safety accidents based on word2vec word embedding and clustering model [J] Computer system application, 2021,30 (01): 141-145.

[8] Dong Yingchao, Wang Bin, Ma Sasa, Liu Hui, Xiong Xin, Xue Jie Research on dimension reduction method of brain network state observation matrix based on t-sne [J] Computer engineering and application, 2018,54 (01): 42-47.

[9] Zhang Yanan, Feng Jianwen Microblog hot topic discovery method based on hybrid clustering [J] Journal of Hangzhou University of Electronic Science and Technology (NATURAL SCIENCE EDITION), 2018,38 (01): 59-64 + 80.