Chinese community topic classification method based on graph model

Shu'an Zhang[a], Xi Wang[b], Rencheng Sun*[a], He Gao[a]
[a] College of Computer Science & Technology, QingDao University, QingDao, Shandong, China, 266071;
[b] Communication dispatching department, Qingdao emergency center, Qingdao, Shandong, China, 266035
* Corresponding author: qdsunstar@163.com

ABSTRACT

Community topic classification is the basis of hot topic discovery. Existing graph models ignore the importance of key information to the text when performing text classification and increase the influence of irrelevant data. To address these problems, we propose a community topic classification model DGAT that incorporates key information as well as information about the topic itself. An integrated algorithm is proposed to extract keywords to avoid the problem of inaccurate keyword extraction. Then a composite complex network model containing both topic and keyword nodes is built. Finally, the graph attention mechanism is used to update node features and incorporate semantic-level attention to learn the effect of different graph structures on the current node classification. An example validation on the Qingdao community topic dataset proves the effectiveness of the method and outperforms the baseline models.

Keywords: community topics, keywords, graph model, attention mechanism, topic classification

1. INTRODUCTION

The data submitted by residents in the community platform is called community topic data. It is difficult for administrators to filter the data submitted by residents, and they need to classify the topics first and then select the events that residents need to solve urgently. Therefore, it is especially important to design a method that fits the classification of community topics.

Topic data is a kind of short textbook. With the development of Deep Learning[1], models based on Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) have been used in text classification tasks. Kim[2] proposed using convolutional kernels of different sizes to obtain sentence features. Kalchbrenner[3] used wide convolution to extract long-distance text information. Liu[4] proposed three RNN text classification models for different tasks. Li[5] proposed a low-complexity deep CNN architecture. Vaswani[6] Proposed Transformer model based on Attention mechanism and Codec structure. All of the above methods alleviate the reliance on manually constructed features, but cannot extract the original structured information in the graph when processing graph data, while graph representations of text have the advantage of capturing discontinuous and long-range semantics.

In recent years, graph neural networks (GNN)[7-9] have attracted extensive academic attention. Defferrard[10] first applied GNN to text classification task by implementing spatial graph convolution operation. Yao[11] proposed Text-GCN model to treat text classification as a node classification task. Based on Yao, Huang[12] proposed to build an independent graph with shared parameters for each document to reduce the storage space consumption. Zhang[13] proposed TextING to update node information using Gated GNN[14]. Literatures[15-16] introduce other models to alleviate the sparsity problem, but increase the model complexity. In addition, all of the above methods ignore the importance of key information to the text when building graph models, and construct a single graph by treating all words equally, increasing the influence of irrelevant data.

The topic data has the following characteristics. The number of words is small and the features are sparse. Some of the category labels to which the topic data belong appear in the topic, and local key information is especially important for this part of the topic. The topics are directly posted by residents, and there is a lot of spoken information, and the global information of this part of the topics is especially important.

Considering the above aspects, this paper proposes to build a composite complex network[17] containing two kinds of nodes, keywords and topics, to make up for the shortage of graph model construction in existing methods. The keywords

Third International Symposium on Computer Engineering and Intelligent Communications (ISCEIC 2022), edited by Xianye Ben, Proc. of SPIE Vol. 12462, 1246210 · © The Authors. Published under a Creative Commons Attribution CC-BY 3.0 License · doi: 10.1117/12.2660992

are extracted by an integrated algorithm[18], focusing on the influence of semantics and word frequency on the text to strengthen the role of key information. To obtain global information, a bi-directional long and short term memory network (BiLSTM) is added for feature enhancement from the topic itself features. Two graph structures are extracted from the established network, and then the node information is updated based on graph attention network (GAT) and semantic attention is added for feature fusion in order to complete the task of text classification of community topics.

2. A COMMUNITY CLASSIFICATION METHOD BASED ON GRAPH MODEL

The proposed graph model-based community topic classification method DGAT includes the following steps. First, the keywords of the topic data are extracted by the integration algorithm, and the features are enhanced by using BiLSTM. Then, a keyword-topic composite complex network is built, from which two graph structures of topic nodes are mapped and then the node features are updated and fused.

2.1 Graph model construction

Keywords are the core words that characterize the single topic data. The established complex complex network model is shown in Figure 1, which contains two kinds of nodes, keyword, topic, belonging relationship, and similar relationship with two kinds of connected edges. The two graph structures are generated by keyword-topic affiliation mapping and topic similarity relationship respectively.

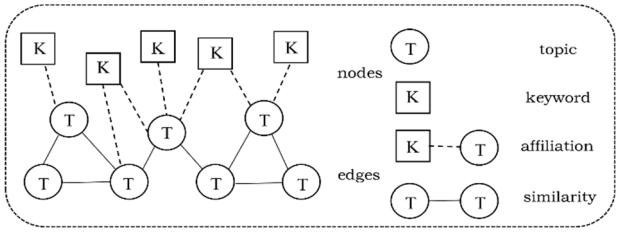

Fig 1. Keywords-topic composite network mod

2.1.1 Model Description

Definition 1 Topic. All the topic data form a topic set, denoted as *T*. Each sentence in *T* is called t_i, where *i*=1,2,3... |*T*|, |*T*| is the number of topics in the topic set.

Definition 2 Keywords. For $K_i \in T$, multiple words can be extracted to characterize the topic, called the keywords of t_i, denoted as k_i. All keywords extracted in T are noted as K, where $K = K_1 \cup K_2 \cup K_3 \cup ... \cup K_i$.

Definition 3 TopicNet. Denoted as TopicNet $=<T,E,S>$, where *T* is the topic set. $E = \{e \mid e =< t_i, t_j >, t_i, t_j \in T\}$ is the set of undirected edges between similar topics.S is the similarity between topics.

Definition 4 Key-TopicNet. Denoted as Key-TopicNet$=<N, E, W>$, where $N = \{k_1, k_2, ..., k_n\} \bigcup \{t_1, t_2, ..., t_m\}$ is the set of nodes. $E = \{e \mid e =< t_i, k_j >, t_i, k_j \in N\}$ is the set of edges, $e =< t_i, k_j >$ indicating the undirected edges of the k_j belonging to the t_i. *W* indicates the probability of the word becoming a keyword.

2.1.2 Keywords-topic affiliation mapping generates graph structure

We propose to use an integrated algorithm to extract keywords, the integration operation H is as defined in Equation (1).

$$\mathrm{H}(U,F) = \sum_{i=1}^{n} u_i f_i \tag{1}$$

Where the set of weights $U = \{u_1, u_2, ..., u_n\}$, the magnitude of the weights indicates the magnitude of the impact that each algorithm has on the results. $\mathrm{F} = \{f_1, f_2, ..., f_n\}$, *n* is the total number of base algorithm results and requires $u_i > 0, \sum_{i=1}^{n} u_i = 1$.

Considering that the TextRank algorithm[19] and LTP technique can extract semantic relationships between words, and the TF-IDF algorithm[20] calculates word frequency relationships to complement the former, the three algorithms are integrated to extract keywords and obtained with an integration ratio of 1:1:2.

The relationship between keywords and topics is many-to-many. If there exists edge $e_i =< t_i, k_i >$ and edge $e_j =< t_j, k_i >$, it means that both t_i and t_j are connected to keyword node k_i, so t_i and t_j may belong to the same category. Therefore, a kind of undirected graph between topics and topic nodes is mapped according to the belonging relationship between topics and keywords. This undirected graph is represented as $G=(A, X)$, where $\boldsymbol{A} \in \mathrm{R}^{n \times n}$ is a symmetric adjacency matrix, a_{ij} is an element in A, $a_{ij}=1$ indicates that there are connected edges between t_i and t_j, n=|T|, $X \in \mathrm{R}^{n \times d}$ is the feature matrix of topic nodes, and d is the dimension of features.

2.1.3 Topic similarity generation graph structure

To capture the influence of non-critical information on the topic data, the feature similarity S between topic nodes is first calculated, and the formula is shown in (2). Then the K-nearest neighbor idea is used to obtain the K nodes with the greatest similarity to the current node for concatenating edges. Finally, the undirected graph structure $G_k =(A_k , X)$ is extracted from the composite network, and $A_k \in \mathrm{R}^{n \times n}$ is the symmetric adjacency matrix of the KNN graph.

$$S_{ij} = \frac{\boldsymbol{x}_i * \boldsymbol{x}_j}{|\boldsymbol{x}_i||\boldsymbol{x}_j|} \tag{2}$$

Where $\boldsymbol{x}_i$、$\boldsymbol{x}_j \in \mathrm{R}^{d \times 1}$ are the features of nodes t_i and t_j respectively, which are one-dimensional vectors. $|\boldsymbol{x}_i|$ and $|\boldsymbol{x}_j|$ are the modes of nodes t_i and t_j respectively.

In addition, in order to make the initial feature matrix X of topic node e obtain more comprehensive information, we use BiLSTM model for feature enhancement. First, the word vector of t_i is initialized using word2vec, so the initialized feature matrix $\boldsymbol{x}_i$ of t_i is obtained. Then the features in both directions are obtained using the forward and backward LSTMs respectively, and finally the two are spliced. Based on this, the initial feature matrix $X = \{x_1, x_2, ..., x_n\}$ of the whole topic set T is obtained.

2.2 A community topic classification model based on graph attention mechanism

GAT can effectively filter the noise information and preserve the global structure information of the graph, so GAT is used to update the node information.

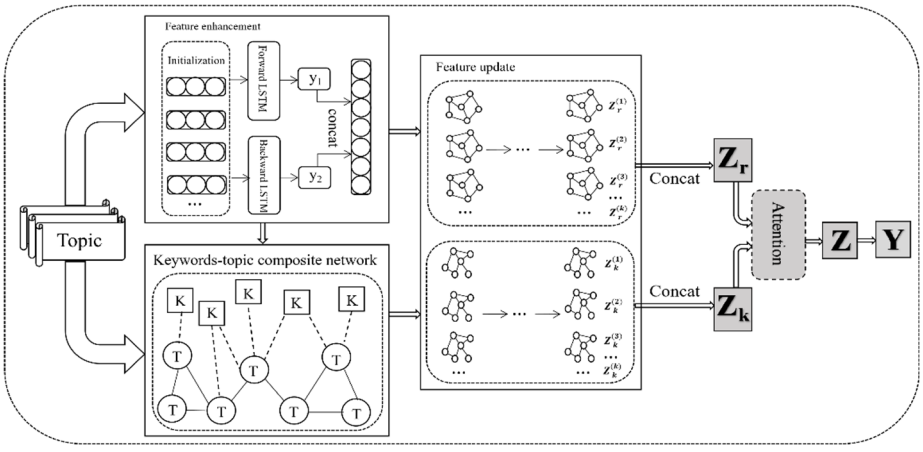

Fig 2. The architecture of DGAT

After the previous steps, two graph structures G and G_k are obtained, and different structures have different levels of importance for different topic data. Based on this, this paper proposes to add a semantic attention layer to learn the importance weights of different structures for the current nodes. Figure 2 shows the model architecture of DGAT, which is mainly divided into three parts: graph model construction, feature updating and fusion, and node classification.

2.2.1 Node Updates

Using the topics as nodes, enter the adjacency matrices A, A_k, and X in the model. First the model calculates the attention fraction α_{ij} between node pairs $< t_i, t_j >$.

$$\alpha_{ij} = \text{soft max}(\text{Leaky Re LU}(\beta^{T}[\gamma x_i \,||\, \gamma x_j])) \tag{3}$$

Where γ is the shared weight matrix obtained from training. β is the attention parameter vector.

z_i is the feature of t_i after one nonlinear transformation and k denotes the use of a multi-headed attention mechanism at the intermediate layer.

$$z_i = \frac{1}{K}(\sum_{j \in N_i} \alpha_{ij}^{(k)} * \gamma^{(k)} x_j) \tag{4}$$

Where $\alpha_{ij}^{(k)}$ and $\gamma^{(k)}$ are the attention coefficients and shared parameter matrices obtained from the training of the *k*th head attention mechanism, respectively.

2.2.2 Feature Fusion

The semantic attention added to learn the importance of different structures is shown in Equation (5).

$$(\theta_r, \theta_k) = \text{Att}_{NN}(Z_r, Z_k) \tag{5}$$

Where, θ_r and θ_k are the importance weights of different semantic features, respectively. For node t_i, z_i^r is its feature under matrix Z_r. Specifically, the attention coefficients θ_i^r for the influence of different semantic features on the current node classification result are obtained using nonlinear transformation and normalization as follows.

$$\partial_i^r = \mu^{T} * \tanh(W * z_i^r + b) \tag{6}$$

$$\theta_i^r = \text{softmax}(\partial_i^r) = \frac{\exp(\partial_i^r)}{\exp(\partial_i^r) + \exp(\partial_i^k)} \tag{7}$$

The features of node t_i under Z_r are mapped to a real weight ∂_i^r by a nonlinear transformation, and similarly ∂_i^r is the weight under the feature matrix Z_k of node t_i. Then the two are normalized to the attention coefficients θ_i^r and θ_i^k by softmax .Finally, the two features are weighted and summed by semantic attention coefficients to obtain *Z*.

$$Z = \theta_r * Z_r + \theta_k * Z_k \tag{8}$$

2.3 Loss function

The loss *Loss* of the model is minimized using the cross-entropy function, while the L_2–norm is added to prevent overfitting.

$$Loss = -\sum_{i=1}^{|T|} \sum_{j=1}^{C} y_{ij} \ln p_{ij} + \lambda \| \vartheta \|^2 \tag{9}$$

Where *C* is the label of the topic data, y_{ij} is the true label of the topic data, p_{ij} is the probability value of the model for

the predicted label of the topic data. p_{ij} is the regularization factor, and ϑ is the model parameter.

3. EXPERIMENTS AND RESULTS ANALYSIS

3.1 Dataset

Since there is no public dataset of topic data, we use the Qingdao community topic dataset to verify the validity of the method. Firstly, 4000 randomly selected data are labeled and experimented, and the data are divided into 10 categories with labels of epidemic, handling, mask, garbage, maintenance, disinfection, volunteer, virus, quarantine, and environment. The complex complex network built is shown in Figure 3, where the topic nodes are in blue and the keyword nodes are in orange for the topic classification task.

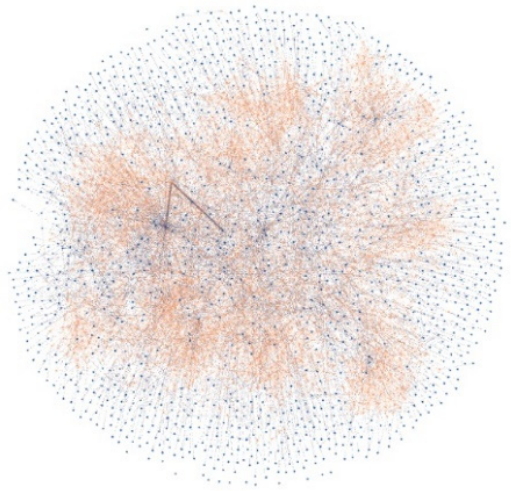

Fig 3. Keywords-topic composite network

3.2 Comparison experiment

The data set was randomly divided into training set, validation set, and test set in the ratio of 8:1:1, and a total of four cross-validations were performed to observe the classification accuracy accuracy (acc) and macro-F1 (F1-score) values for each experiment. If the acc value and F1-score value are higher, the better the model classification effect is.

$$\text{accurary}=\frac{T}{T+F} \tag{10}$$

$$\text{macro-F}_1=\frac{2*\text{macro-precision}*\text{macro-recall}}{\text{macro-precision}+\text{macro-recall}} \tag{11}$$

Where T is denoted as the number of samples with all correct predictions, F is denoted as the number of samples with all incorrect predictions, macro-precision is the precision rate, and macro-recall is the recall rate.

We use six benchmark models, namely TextCNN, BiLSTM, DPCNN, Transformer, TextGCN and TLGNN. The word vectors are initialized using word2vec with a dimension of 300 dimensions. The results and average values of the cross-validation experiments are shown in Table 1.

Table 1. Comparison of classification performance of different models

Models	Experiment I		Experiment II		Experiment III		Experiment IV		Average	
	acc	F1-score	acc	F1-score	acc	F1-score	acc	F1-score	acc	F1-score
TextCNN	0.8711	0.8697	0.8910	0.8818	0.8902	0.8712	0.8405	0.8296	0.8732	0.8631
BiLSTM	0.8496	0.8502	0.8688	0.8449	0.8786	0.8639	0.8113	0.8045	0.8521	0.8409
DPCNN	0.8383	0.8436	0.8409	0.8127	0.8531	0.8230	0.7936	0.7631	0.8315	0.8106
Transformer	0.8257	0.8141	0.7951	0.7662	0.8074	0.7916	0.7843	0.7523	0.8031	0.7811
TextGCN	0.7305	0.7410	0.7019	0.6785	0.7117	0.6799	0.6670	0.6419	0.7028	0.6853
TLGNN	0.7494	0.7368	0.7189	0.7223	0.7406	0.7338	0.7524	0.7381	0.7403	0.7328
DGAT	0.8996	0.8973	0.9113	0.9034	0.9085	0.8947	0.8815	0.8776	0.9002	0.8933

In Table 1 we can see that the average accuracy as well as the average F1-score are highest on the DGAT model with 0.9002 and 0.8933, respectively, which indicates that DGAT can be applied to community topic classification and can achieve better results. The TextCNN model achieves an average accuracy of 0.8732, which can outperform most of the baseline methods because the convolutional operations can mine the key information in the short text. The Transformer model requires a large training set for training, so it does not work well on this community topic data. The TextGCN, TLGNN and DPCNN models are all designed for long text and do not work well in the face of the data sparsity problem of short texts of topic data.

3.3 Ablation experiments

In this paper, a total of two edge structures were generated when building the graph model structure. Therefore, this experiment is designed to see which of the connected edges contributes more to the final result and whether it is reasonable. The experimental results are shown in Table 2.

Table 2. Ablation experiment

Indicators		$G_k=(A_k,X)$	$G=(\boldsymbol{A},\boldsymbol{X})$	DGAT
Experiment I	acc	0.8205	0.8762	0.8996
	F1-score	0.8194	0.8724	0.8973
Experiment II	acc	0.8086	0.8990	0.9113
	F1-score	0.7907	0.8918	0.9034
Experiment III	acc	0.7926	0.9001	0.9085
	F1-score	0.7809	0.8824	0.8947
Experiment IV	acc	0.8015	0.8820	0.8815
	F1-score	0.7961	0.8457	0.8776
Average	acc	0.8058	0.8893	0.9002
	F1-score	0.7968	0.8731	0.8933

As can be seen from Table 2, the acc value is only 0.8058 when using only feature similarity as the edge concatenation method. The acc value can reach 0.8893 for the keyword-generated concatenation method. The F1-score values are also in a sequential increasing relationship. This indicates that the key information plays a greater role, but using both alone does not exceed the performance of using both simultaneously.

3.5 Effect of training set ratio

To understand the effect of training set proportions on the classification performance of the model, 10%, 20%, 40%, 60%, and 80% proportions of samples from the dataset of Experiment 2 were randomly selected as the training set for the experiments. The comparison with TextCNN, BiLSTM, and DPCNN was performed to observe the acc values, and the experimental results are visualized using line graphs, as shown in Figure 4.

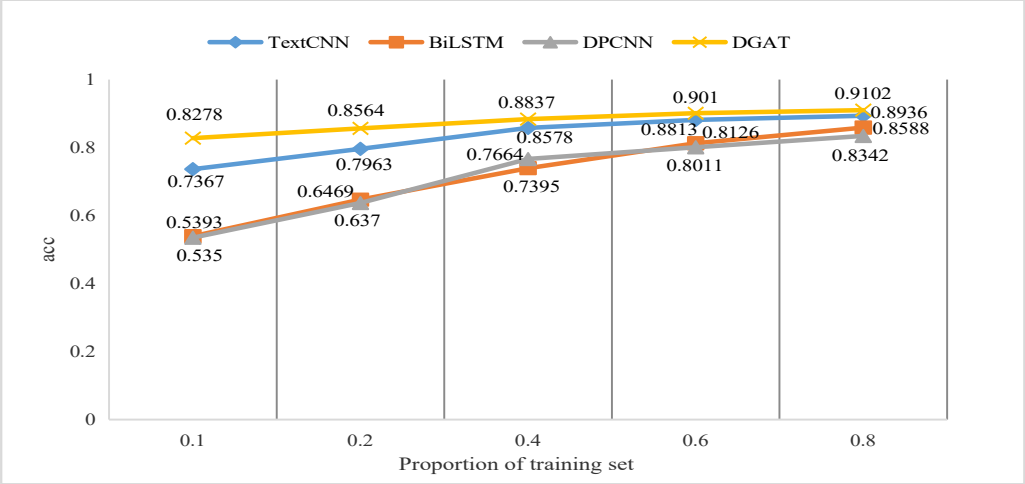

Fig 4. Effect of training set on results - line graph

It can be seen that DGAT can achieve an accuracy of 0.8278 with only 10% of the dataset. DGAT establishes connected edges by combining keywords with information about the topic itself, calculates the weights of node neighbors based on attention, and is able to capture local information without the need for the entire graph structure.This combination of information allows DGAT to show good performance with a small training set.

4. CONCLUSION

In this paper, we propose a community topic classification method based on graph attention mechanism. For the characteristics of community topics, keywords are extracted using an integration algorithm to build a kind of composite complex network containing two kinds of connected edges and two kinds of nodes, and two graph models are extracted from them.Then the features of nodes are updated using graph attention mechanism and semantic attention is added to learn the influence of two graph structures on nodes, and finally the feature representation of topic nodes is obtained.Experiments show that the method proposed in this paper works well for the Chinese community topic classification task.

REFERENCES

[1] Zhao, A. T., LI, J. B. and Dong, J. Y., "Multimodal gait recognition for neurodegenerative diseases," IEEE transactions on cybernetics (2021).
[2] Kim, Y. , "Convolutional Neural Networks for Sentence Classification," Eprint Arxiv (2014).
[3] Kalchbrenner,N., Grefenstette, E. and Blunsom, P., “A convolutional neural network for modelling sentences,” Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics(2014).
[4] Liu, P. F., Qiu, X. P. and Huang, X. J., "Recurrent neural network for text classification with multi-task learning," arXiv preprint arXiv:1605.05101 (2016).
[5] Johnson, R. and Zhang, T., "Deep pyramid convolutional neural networks for text categorization," Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)(2017).
[6] Vaswani, A., Shazeer, N., Parmar, N. and Uszkoreit, J., "Attention is all you need," Advances in neural information processing systems 30 (2017).
[7] Kipf, T. N. and Welling, M., "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907 (2016).
[8] Velikovi, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y., "Graph Attention Networks," arXiv preprint arXiv:1710.10903 (2017).
[9] Gao, H., Yu, X., Sui, Y., Shao, F. J. and Sun, R. C.,"Topological Graph Convolutional Network Based on Complex Network Characteristics," IEEE Access 10: 64465-64472(2022).
[10] Defferrard, M., Xavier B. and Pierre, V., "Convolutional neural networks on graphs with fast localized spectral

filtering," Advances in neural information processing systems 29 (2016).
[11] Liang, Y., Mao, C. S. and Luo, Y., "Graph convolutional networks for text classification," Proceedings of the AAAI conference on artificial intelligence(2019).
[12] Huang, L., Ma, D. and Li, S., "Text level graph neural network for text classification," arXiv preprint arXiv:1910.02356 (2019).
[13] Zhang, Y. F., Yu, X. L. and Cui, Z. Y., "Every document owns its structure: Inductive text classification via graph neural networks," arXiv preprint arXiv:2004.13826 (2020).
[14] Li, Y. J., Tarlow, D. and Brockschmidt, M., "Gated graph sequence neural networks." arXiv preprint arXiv:1511.05493 (2015).
[15] Hu, L. M., Yang, T., Shi, C., Ji, H. and Li, X., "Heterogeneous graph attention networks for semi-supervised short text classification," Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (2019).
[16] Yuan, Z. Y., Gao, S, Cao, J., “Small sample short text classification based on heterogeneous graph convolution network,” Computer Engineering. Papers 47(12), 87-94(2021).
[17] Sui, Y., “Research on multi subnet complex network model and its related properties,” Qingdao University(2012).
[18] Zhang, S. A., Wang, X., Dai, J. P., Sui, Y. and Sun, R. C., “Keywords extraction algorithm based on keyword co-occurrence network,” Complex systems and complexity science(2022).
[19] Mihalcea, R. and Pau,l T., "Textrank: Bringing order into text," Proceedings of the 2004 conference on empirical methods in natural language processing(2004).
[20] Li, J. Z., Fan, Q. N. and Zhang, K., "Keyword extraction based on tf/idf for Chinese news document," Wuhan University Journal of Natural Sciences. Papers 12(5), 917-921(2007).