

Patent Text Classification based on Deep Learning and Vocabulary Network

Ran Li¹, Wangke Yu^{2*}, Qianliang Huang³, Yuying Liu⁴

Intellectual Property Information Service Center & School of Management and Economics
Jingdezhen Ceramic University, Jingdezhen, China^{1,3,4}
School of Information Engineering, Jingdezhen Ceramic University
Jingdezhen, China²

Abstract—Patent documents are a special long text format, and traditional deep learning methods have insufficient feature extraction ability, which results in a weaker classification effect than ordinary text. Based on this, this paper constructs a text feature extraction method based on the lexical network, according to the inner relation between words and classification. Firstly, the inner relationship between words and classification was obtained from linear and probability dimensions and the lexical network were constructed. Secondly, the lexical network is fused with the features extracted from the deep learning model. Finally, the fusion features are trained in the original model to get the final classification result. This method is a classification enhancement method that can classify patent text alone or enhance the accuracy of various types of neural networks in patent text classification. Experimental results demonstrate that the accuracy of BERT combined with lexical network method is as high as 82.73%, and the accuracy of lexical network method combined with CNN and LSTM is increased by 2.19% and 2.25% respectively. In addition, it was demonstrated that the lexical network feature extraction method accelerated the convergence speed of the model during training and improved the classification ability of the model in Chinese patent texts.

Keywords—Text classification; deep learning; network vocabulary; patent; feature extraction

I. INTRODUCTION

As the main carrier of scientific and technological information, patents contain about 90% of global technical information. In 2021, the effective number of Chinese invention patents will be 3.597 million, facing a large number of patent applications, patent text review takes up a lot of time for patent examiners. Patent classification is an important part of the patent review, which can reduce the time and cost of subsequent retrieval and management of patent documents. At present, the main patent classification used in China is the International Patent Classification [1]. IPC classification has five levels, belonging to a multi-level and multi-label classification system. At present, China mainly uses manual methods to classify patents. For example, after receiving patent application documents, patent examiners need rich professional knowledge to accurately classify patents, making subsequent patent texts easy to review and retrieve. In the face of massive patent texts, intelligent and effective implementation of

automatic classification of Chinese patent texts has become the focus of the industry and academia.

Automatic text classification is one of the important fields in NLP (Natural Language Processing). In recent years, there has been much research on text classification using machine learning and deep learning, Early text classification models generally refer to the support vector machine[2][3], Bayesian classifier[4][5], k-Nearest Neighbor algorithm[6][7] and other machine learning[8] methods for text classification. However, text classification of machine learning methods relies on manually constructed features and ignores the context between words and sentences, which makes it difficult to improve the accuracy of such methods. After Hinton[9] and others first proposed the concept of deep learning in 2006, scholars proved that deep learning technology has good performance in text classification. Since then, in the field of text classification, deep learning has gradually replaced machine learning as the mainstream research direction.

Therefore, on the basis of previous studies, based on the analysis of a large number of patent data, this paper proposes a deep learning feature extraction method based on a vocabulary network, which improves the accuracy of deep learning on patent text classification through feature fusion between the extracted features and the deep learning model. The practice shows that this method combined with deep learning has a certain improvement in Chinese patent text classification, which confirms the contribution of this paper in the following aspects.

1) In this study, a method of combining deep learning with vocabulary network feature extraction is proposed theoretically.

2) It is proved in practice that this method combined with other deep learning models can speed up the model convergence and improve the accuracy of model classification.

The rest of this paper is organized as follows: In Section II, we introduced the relevant work and our contributions. In Section III, we introduce our research framework. In Section IV, we selected text data to verify and illustrate the relevant parameters of this model. Finally, the conclusion and further work are given.

*Corresponding Author

II. RELATED WORK

Automatic patent text classification is a technical means to use computers to classify and mark the text according to a specific label system, which is convenient for the patent document management and text search. This paper mainly solves the task of automatic patent text classification through deep learning. Therefore, this section mainly introduces the current research status of scholars using deep learning methods to classify text.

Deep learning text classification is mainly represented by a neural network model, attention mechanism and pre-training model.

1) The text classification method based on the neural network model presents the text in low dimension by word embedding and achieves the classification effect by using an encoder. Representative models include convolutional neural network (CNN)[10], recurrent neural network (RNN)[11] and graphical neural network (GCN)[12], Among them, CNN can acquire local relations[13], RNN uses a recurrent neural network to have the ability to learn long-distance texts[14], GCN can effectively integrate the features of references.

2) Compared with the neural network model, the attention mechanism can focus on a part of the information related to the classification task while ignoring the irrelevant part when processing tasks, In 2014, Bahdanau et al[15].Used the attention mechanism to perform machine translation tasks for the first time, making great progress in machine translation. In text classification tasks, the attention mechanism is often combined with the neural network model, She et al[16]. Proposed a text classification algorithm based on the hybrid CNN-LSTM model, used the feature vector output from CNN as the input of LSTM and used Softmax classifier for classification, which can effectively improve the accuracy of text classification. Yang et al[17]. Based on the combination of convolutional neural networks and long-term and short-term memory, are used to capture local features to enrich feature information and use weight values to adjust the enhancement intensity.

3) The pre-training model is represented by the BERT model, and the improvement of model classification results mainly depends on the adjustment process of the model[18]. BERT has a strong sentence and word expression ability and has an excellent performance in NLP tasks at the word level[19]. Lu et al[20]. Proposed the VGCN-BERT model, which combines BERT's capabilities with vocabulary graph convolutional network (VGCN). The different layers of BERT interact to enable them to interact and jointly build the final classification representation. Zheng et al[21]. Proposed a BERT-CNN model for text classification. By adding CNN to the task-specific layer of the BERT model, you can obtain information about important fragments in the text.

Some of the above methods have been successfully applied to patent text classification tasks, and have achieved some results. As a branch of special long texts in text classification, Chinese patent texts are difficult to improve the accuracy of

classification. Many scholars have carried out a lot of research on this. For example, Wen et al. [22] proposed a multi-level patent text classification model, called ALBERT-BiGRU, which combines ALBERT and bidirectional gated loop units. The dynamic word vector pre-trained by ALBERT is used to replace the traditional static word vector, and the BiGRU neural network model is used for training, which preserves the semantic association between long-distance words in the patent text to the maximum extent. Roudsari et al[23]. Studied the effect of applying the DistilBERT pre-training model and fine-tuned it to complete the important task of multi-label patent classification. To sum up, domestic and foreign scholars' patent text classification methods based on deep learning mainly focus on the fusion application of deep learning models, the adjustment of pre-training models or the introduction of attention mechanisms to extract relevant features. Although these methods can achieve the goal of improving the classification effect, they are all cumulative applications of existing methods, which not only affect the model classification efficiency but also difficult to improve the patent classification effect.

In previous studies, the focus has been on improving the models, while ignoring the specific structure of the data itself, making it difficult to improve the accuracy of patent text classification. Based on this, a generic feature extraction method is considered for patent text feature extraction to improve the accuracy of automatic patent text classification, mainly considering the intrinsic connection between words and classification as the basis for feature extraction, and introducing the extracted features into the deep learning model to fuse with the features extracted by the model to improve the model classification effect.

III. PROPOSED WORK

Through the analysis of previous text classification studies, it is found that LSTM has more advantages than CNN in text classification because of its contextual characteristics[24], which has also been confirmed in many long text classification studies. However, in the classification of Chinese patent texts, there is an exception in that the accuracy of CNN is equal to or slightly higher than LSTM. Through the analysis of patent data and experimental reproduction, it is found that patent documents belong to a special format in long texts, and are insensitive to the context, but sensitive to the words divided by social production practices such as industry and industry. Therefore, in the neural network feature extraction stage of deep learning, the parameters extracted by traditional feature extraction methods in patent texts have the low feature density required for classification, and the existence of multi-word synonyms in feature words increases the difficulty of classification, which makes the full connection layer learning feature inefficient and feature learning incomplete in deep learning, ultimately leading to low accuracy of Chinese patent text classification. In order to solve the problem that the feature extraction ability of Chinese patent text of neural network is insufficient, this paper proposes a feature extraction method based on a vocabulary network to strengthen the feature extraction ability of Chinese patent text of deep learning network, so as to improve the accuracy of Chinese patent text classification.

A. Deep Learning Model based on Vocabulary Network
Feature Extraction

This paper proposes a feature extraction method of a deep learning model based on a vocabulary network. This method combines the extracted features with the extracted features of the original model and expands the features on the basis of retaining the features of the original model to improve the classification accuracy. Fig. 1 is a schematic diagram of the LNFE method of deep learning model fusion, which includes four parts: text vector representation, feature extraction, feature

fusion and text classification prediction. In the figure, P_i is patent data, W_i is patent text information after word segmentation, and F_i is the feature information extracted from text. In the training or prediction process, the patent text is first segmented and processed separately. In the deep learning network feature extraction, the data is extracted in the original feature extraction layer and LNFE, respectively, and then the extracted features are spliced and fused, and then the appropriate parameters are trained in the full connection layer to classify the subsequent data.

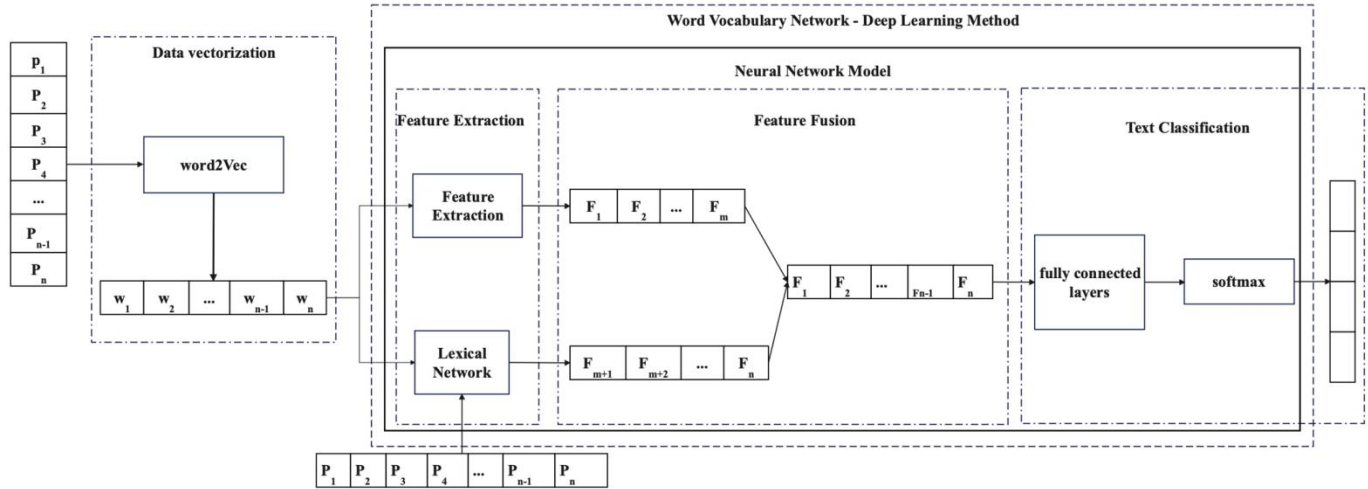


Fig. 1. Schematic diagram of vocabulary network and model fusion

B. Vocabulary Network Generation

The lexical network is a key step in feature extraction, which includes synonym clustering of words in the patent, the internal relationship between words and classification and other information. In the subsequent feature extraction process, the characteristic words of patent text can be accurately extracted using the lexical network as a reference. Fig. 2 is a schematic diagram of the generation of a vocabulary network.

First, the word vector (Word Embedding) representation of each word in all texts is obtained by unsupervised learning of Word2vec, and the word vector similarity is calculated by using the cosine formula of Formula 1 to cluster and merge synonyms in words, thus avoiding the problem of slow model convergence caused by synonyms in the feature extraction stage.

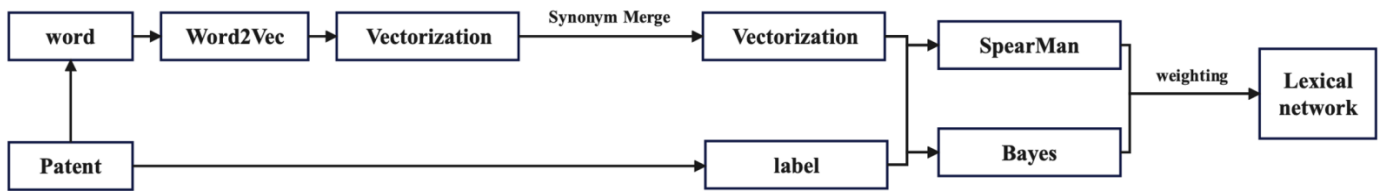


Fig. 2. Schematic diagram of vocabulary network generation

$$\text{Sim}(\vec{v}, \vec{w}) = \sum_i^N v_i w_i \quad (1)$$

Where \vec{v} and \vec{w} are word vectors trained by Word2vec respectively, Sim is the similarity between vector \vec{v} and \vec{w} , v_i and w_i is the value of the vector \vec{v} and \vec{w} at different latitudes.

Each group of word vectors after merging has corresponding label labels. According to Formula 2 and Formula 3, the prior probabilities of words and classifications are obtained from the two dimensions of probability and linearity through Label information and word vector

information, and each word vector is given a classification feature weight by weighting.

$$\rho = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (2)$$

$$P(B_i|A) = \frac{P(A|B_i)P(B_i)}{\sum_{i=1}^n P(A|B_i)P(B_i)} \quad (3)$$

Where the ρ Is the correlation between word vector and classification, x_i is the frequency of word change in patent No. i , \bar{x} appears in this category \bar{x} Average frequency of. y_i is x_i

Classification of y in i word patent, \bar{y} is the average of the categories in the patents with x_i words. In Formula (3), $P(B_i|A)$ represents the probability of being classified as B_i when the word A appears, $P(A|B_i)$ is the probability of occurrence of A in B_i classification, $P(B_i)$ is the probability of occurrence of B_i classification.

C. Classification Model Design

Based on the characteristics of existing models, this paper selects CNN, LSTM and Bert models to combine with LNFE as the new deep learning model, and designs LNFE, CNN, LSTM and Bert classical models as the contrast to judge the classification effect of the new deep learning model.

- LNFE-CNN model

LNFE can use patent text feature extraction to form a vocabulary network. CNN cannot extract the context, semantic information of the patent text. The LNFE-CNN model combines the characteristics of a vocabulary network and convolutional neural network to improve the training speed. Fig. 3 is the schematic diagram of the LNFE-CNN model.

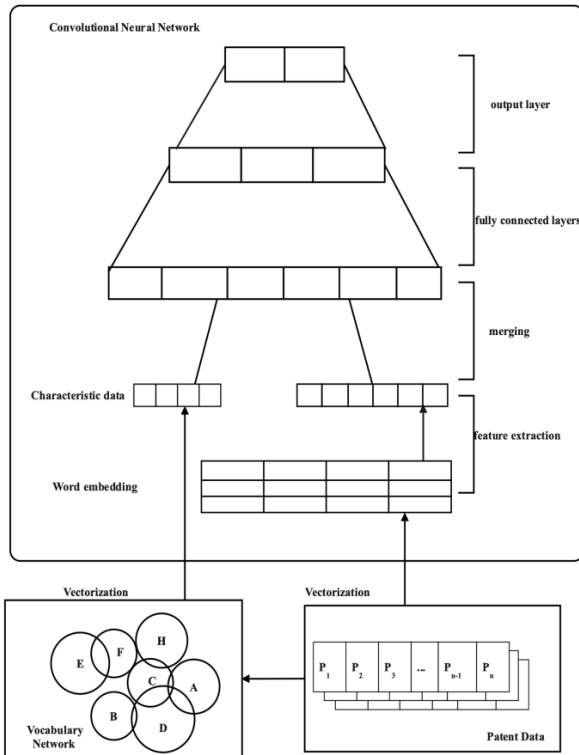


Fig. 3. Schematic diagram of the LNFE-CNN model

- LNFE-LSTM model

LNFE-LSTM model short-term memory is a special RNN network, which is mainly used to solve the problem of gradient disappearance and gradient explosion during long sequence training. LSTM mainly has three gate structures: forgetting gate, input gate and output gate. The forgetting gate, mainly decides to give up the information and leave important information. The input gate, mainly determines which information is added, and the output gate determines which

information can be output after judgment. The supplementary calculation formula of the LNFE-LSTM model is as follows:

Forgotten Gate

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

Input gate

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (5)$$

$$\tilde{C}_t = \tan h(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (6)$$

$$C_t = f_t \cdot C_{t-i} + i_t \cdot \tilde{C}_t \quad (7)$$

Output gate

$$O_t = \sigma(W_O \cdot [h_{t-1}, x_t] + b_O) \quad (8)$$

$$h_t = O_t \cdot \tan h(C_t) \quad (9)$$

In this paper, we propose an LNFE-LSTM model, which can accurately extract patent features through LNFE, combine LSTM text classification to deal with long text classification problems, and has the characteristics of context semantic features.

- LNFE-Bert model

The Bert model perfectly considers the bidirectional semantic features, while retaining the maximum meaning of the text according to the context word order features. This paper accurately extracts the patented features through LNFE, combines Bert (Bidirectional Encoder Representations from Transformers) [25] with the bidirectional transformers encoder, trains the model through the masked language model and next sentence prediction, Have strong sentence expression ability. The supplementary calculation formula of the LNFE-Bert model is as follows:

$$(\theta, \theta_1, \theta_2) = L_1(\theta, \theta_1) + L_2(\theta, \theta_2) \quad (10)$$

$$L_1(\theta, \theta_1) = -\sum_{i=1}^M \log_p(m = m_i | \theta, \theta_1), m_i \in [1, 2, \dots, |V|] \quad (11)$$

$$L_2(\theta, \theta_2) = -\sum_{j=1}^M \log_p(n = n_j | \theta, \theta_2), n_j \in [IsNext, NotNext] \quad (12)$$

D. Evaluation Indicators

In order to evaluate the classification effect of the LNFE method combined with deep learning, this score uses Precision, Recall and F1 values as evaluation indicators, and judges the feature extraction ability of the LNFE method by comparing various indicators of LNFE, CNN, LNFE CNN, LSTM, LNFE LSTM, Bert, and LNFE Bert models. Formula (13-15) is the calculation formula of the P value, R-value and F1. Generally, the higher the P value is, the higher the prediction accuracy is which proves that the stronger the prediction ability of the model is, the higher the R-value is, the better the model is, and the better the classification effect is. Since the single expression of the P value and R-value may lead to incomplete measurement standards due to the uneven distribution of data, F1 value can better evaluate the model by means of a weighted average.

$$P = \frac{\text{Predicted number of correct samples}}{\text{All predicted results}} \quad (13)$$

$$R = \frac{\text{Predicted number of correct samples}}{\text{All results in the sample}} \quad (14)$$

$$F_1 = \frac{2 \cdot P \cdot R}{P + R} \quad (15)$$

IV. MODEL EVALUATION AND RESULT ANALYSIS

A. Experimental Platform and Parameter Selection

The processor of the experimental platform used in this paper is Dual Core Intel Core i7, the memory is 8 GB (2133 MHz) LPDDR3, and the GPU is Nvidia 2080Ti. The operating environment is Python 3.9, the network framework is PyTorch, and the training models are CNN, LSTM, BERT, LNFE, LNFE-CNN, LNFE-LSTM, and LNFBERT. Among them, CNN, LSTM and BERT are general comparison networks in current patent classification research, and LNFE is a neural network that only uses the LNFE method as a feature extraction method. LNFE-CNN, LNFE-LSTM and LNFE-BERT are neural networks constructed by combining the extracted features of LNFE with the original features. Through analysis, it is found that the LNFE method has a high correlation with the classification in the first 40 words on average in the extracted feature times, so the LNFE method intercepts 40 words for training. In the training process, Table I is selected as the parameter of the neural network for training.

TABLE I. NEURAL NETWORK PARAMETER SETTING

net	learn rate	bach size	word size	word vector
CNN	5.00E-4	64	240	200
LSTM	5.00E-4	64	240	200
BERT	3.00E-5	32	240	200
LNFE	1.00E-3	32	40	240
LNFE-CNN	1.00E-3	64	240+40	200
LNFE-LSTM	1.00E-3	64	240+40	200
LNFE-BERT	5.00E-5	32	240+40	200

B. Data Source and Preprocessing

In this experiment, the "part" in the main IPC classification number is selected as the classification basis. The classification data is from the incoPat patent database. The obtained patent information includes the title, abstract, IPC main classification number, independent claims and other information. The uniform distribution of data in the learning features of the model is conducive to improving the learning effect of the model. Therefore, patent data are randomly selected from the eight IPC ministerial classifications of patents, and 125064 patents are screened through data denoising. In order to evaluate the effectiveness of the model, cross-validation method is adopted. The paper experiment extracts 80% of the collected data as the training set, 10% as the verification set, and 10% as the test set. The data distribution of the patent data set is shown in Fig. 4.

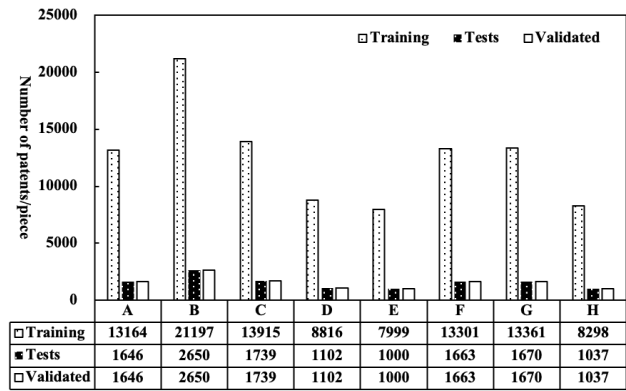


Fig. 4. Data distribution

C. Patent Text and Word Vector

The title and abstract of a patent are a summary of patent documents. Independent claims fully represent technical solutions and necessary technical features. This information can be used as a feature source of patent classification. To classify patents using deep learning, we first need to import the text to a word vector into the neural network, so we need to use Word2vec to convert Chinese patent text information into a word vector and then import it into the neural network for learning. This paper combines several neural networks, including Word2vec, CNN, LSTM, Bert, and so on. The vocabulary length of the training text is fixed during the training process. Table II shows the statistics of the vocabulary length of patent texts. The vocabulary queue of most patent texts is between 200 and 400, with an average vocabulary length of 241. In this paper, 240 words are used as the text cutting position, and the missing parts are filled with 0 vectors.

TABLE II. MODEL PARAMETER SETTINGS

Data	Sample length below 200 words	200 words - 400 words	Sample length above 400
Training	12309	62819	20192
Validation	712	13890	1201
Test	819	13210	1192

D. Vocabulary Network Generation

In the process of vocabulary network generation, 125064 patent documents generated 272141-word vectors, and 241524-word vectors remained after removing stop words and merging synonyms. By calculating the Spearman correlation and Bayes probability weighting of words and tags, 30080 words with strong correlation characteristics were obtained and classified, and their word network is shown in Fig. 5. Each point in the vocabulary network represents a word. The position of the point is the mapping of the high latitude of the word vector in the low dimensional. The shorter the distance between the points, the closer is the part of speech. The colors of the dots represent different parts of the patent classification related to the word. It can be found from the number of related words that a large number of words have a strong relationship with some categories. From the distribution of words, the meaning of words is also related to classification, which also confirms the above statement that "Chinese patent texts are sensitive to words classified by social production practices such as industries and industries".

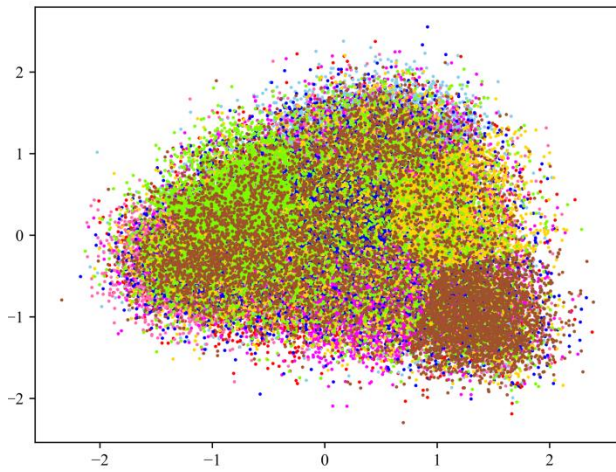


Fig. 5. Vocabulary network diagram

E. Model Training

The deep learning model used for classification is usually trained at the full connection layer after feature extraction. Finally, the data are classified through SOFTMAX. After repeated training, the accuracy of classification is improved. In order to verify whether the LNFE method has the ability of

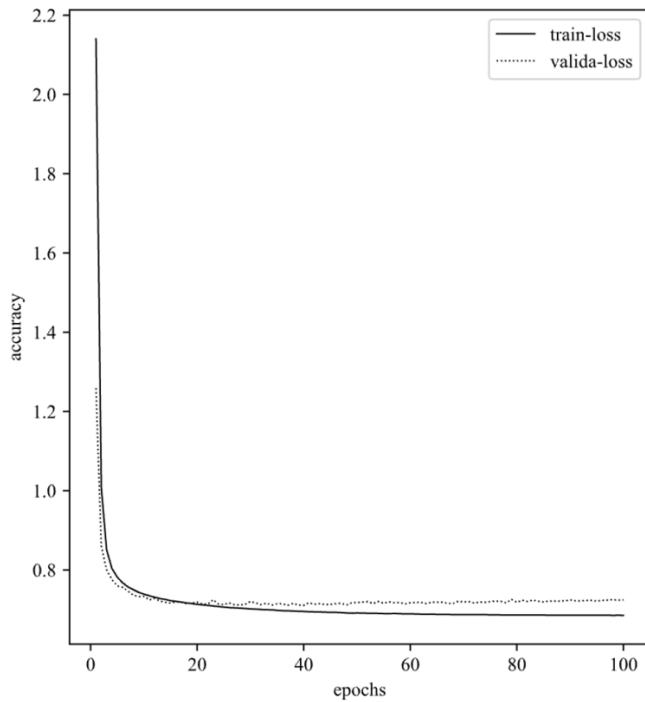


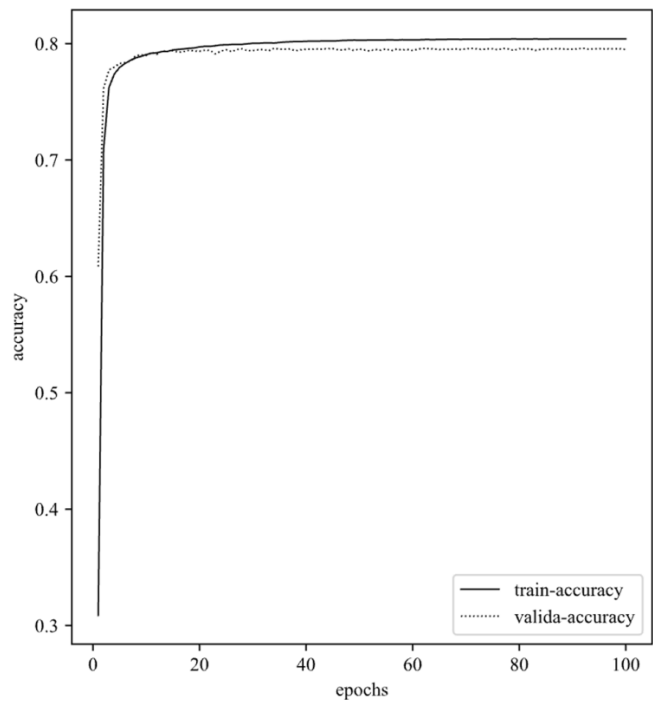
Fig. 6. LNFE training process

Word2vec+LNFE method is slightly higher than other networks; it is lower than traditional neural networks combined with the LNFE method. This is because the LNFE method only extracts features in probability and linear dimensions, while

feature extraction, the LNFE method is used as the feature extraction layer of the neural network to test the classification results. First, after removing the stop words and vectorization of the Chinese patent text, the similarity analysis of the word meaning with the words in the vocabulary network is conducted, and the words in the word segmentation results of the patent text are replaced with highly similar words in the vocabulary network to merge the multi-word consent words. According to the weighted results of words and classification, the word vector of 40 words with the highest weight is selected as the feature, which is sent to the network layer for training. The training process is shown in Fig. 6. After several iterations of training, the accuracy of the model can be found to improve rapidly. This result confirms that the LNFE method indeed extracts classified feature data.

F. Accuracy Verification

After the LNFE method is confirmed to have the feature extraction capability, this section verifies the feature extraction effect of LNFE by comparing it with other networks. Table III shows the comparison results between the LNFE method and other networks. During the validation process, it is found that traditional neural networks have improved after combining the LNFE method, while Bert has limited improvement ability to Bert due to its excellent feature searching ability. Although the



traditional networks consider more dimensions. Because the LNFE method has the characteristic of being Embeddable for neural networks, it has practical value.

TABLE III. CHINESE PATENT TEXT CLASSIFICATION RESULTS

Number	Model	P/%	R/%	F1/%
1	Word2vec + LNFE	81.16	73.24	77.00
2	Word2vec + LSTM+ Attention	78.83	65.23	72.20
3	Word2vec + LNFE-LSTM + Attention	81.02	69.12	74.60
4	Word2vec + CNN	79.32	66.43	72.89
5	Word2vec + LNFE-CNN	81.57	74.51	77.88
6	Bert	82.31	76.13	79.10
7	LNFE-Bert	82.73	78.31	80.27

G. Data Analysis

The classification results after CNN and Bert classify separately and combined LNFE method is shown in Fig. 7. Because the text features found by the LNFE method are more explicit, LNFE CNN has a faster convergence speed than CNN. At the same time, LNFE CNN may learn more text features than CNN, so the accuracy of LNFE CNN is slightly higher than CNN. The LNFE method has the prior probability of

words in the patent text so that LNFE Bert can find the complex functions of features and classification earlier than the Bert model in the training process, that is, the LNFE Bert model converges earlier than the Bert model. However, Bert's model pays more attention to the feature extraction of text. After a lot of training, the accuracy of the Bert model and LNFE Bert is almost the same. Therefore, it can be determined that the LNFE method can improve the learning ability and classification accuracy of traditional deep learning models.

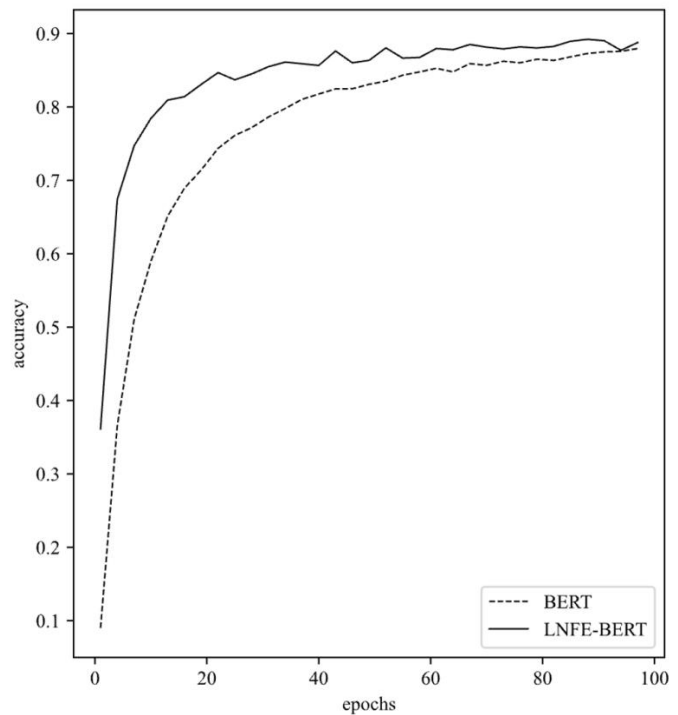
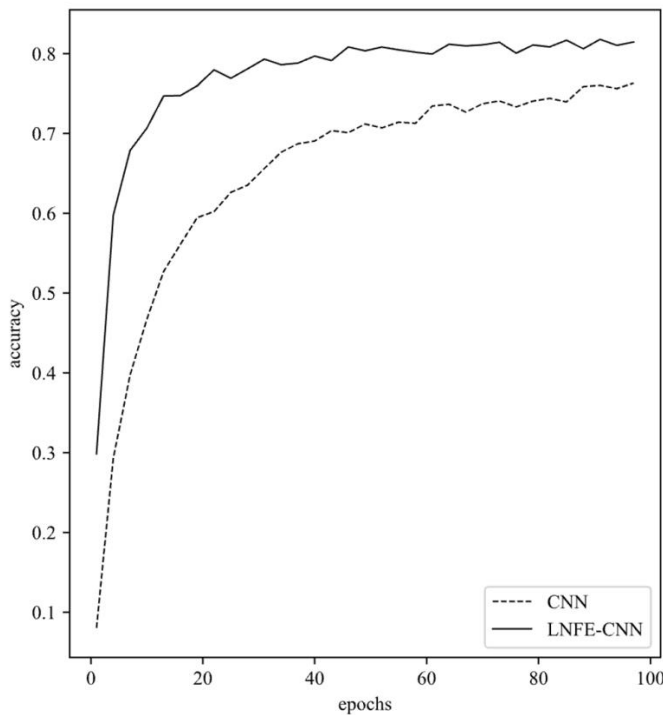


Fig. 7. LSTM and LNFE-LSTM training process chart

V. CONCLUSION

When comparing many text classification materials, it is found that the accuracy of the classification results of Chinese patent texts is lower than that of other texts. Through repetition experiments and analysis and verification, it is concluded that patent documents are insensitive to context due to special formats, and the traditional deep learning model has insufficient ability to extract features of Chinese patent texts. These two reasons lead to slow convergence speed and low accuracy of the model. In order to solve this problem, this

paper proposes an LNFE feature extraction method, which uses Spearman and Bayes to obtain the prior probabilities of vocabulary and classification from the probability and linear dimensions respectively, constructs the network relationship between vocabulary and classification in a weighted way, and uses the features extracted from the vocabulary network to fuse with the features extracted from the deep learning model for classification. The method differs from the traditional method of improving the accuracy of the model by altering the neural network, by extracting the patent text features in depth through the prior probability of the text, thus allowing the combination

of various neural networks to enhance the performance and accuracy of the network in patent text classification. The experimental results demonstrate that the method speeds up the convergence of the model to a certain extent and improves the classification accuracy of the model. However, in this paper, only a ministerial lexical network is constructed to validate the method for Chinese patent text classification. In future work, we will try the method for more fine-grained patent classification in the case of feature extraction, and try the method in other text classification to verify whether the method is universal.

In future research, we will enrich the construction methods of patent text features based on how to make more subtle potential features of patent text, so as to improve the classification performance of the model. At the same time, it will also further explore the performance of the model applied to other tasks in the field, and consider how to deal with multi-classification problems to provide better help for researchers.

ACKNOWLEDGMENT

This article was supported by the Special Project for Scientific and Technological Cooperation of Jiangxi Province Grant Number [20212BDH80021].

REFERENCES

- [1] H. Sasaki and I. Sakata, "Identifying potential technological spin-offs using hierarchical information in international patent classification," *Technovation*, vol. 100, p. 102192, Feb. 2021.
- [2] Z. Liu, H. Kan, T. Zhang, and Y. Li, "DUKMSVM: A Framework of Deep Uniform Kernel Mapping Support Vector Machine for Short Text Classification," *Appl. Sci.-Basel*, vol. 10, no. 7, p. 2348, Apr. 2020
- [3] F. Sun and S. Lian, "nu-Improved nonparallel support vector machine," *Sci Rep*, vol. 12, no. 1, p. 17855, Oct. 2022.
- [4] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *Journal of Information Science*, vol. 44, no. 1, pp. 48–59, Feb. 2018.
- [5] H. Gao, X. Zeng, and C. Yao, "Application of improved distributed naïve Bayesian algorithms in text classification," *J. Supercomput.*, vol. 75, no. 9, pp. 5831–5847, Sep. 2019.
- [6] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1503–1509, Jan. 2012.
- [7] F. Barigou, "Improving K-Nearest Neighbor Efficiency for Text Categorization," *Neural Netw. World*, vol. 26, no. 1, pp. 45–65, 2016.
- [8] U. A. K. Santoshi, "Text Classification Using Machine Learning Techniques," *Specialusis Ugdymas*, vol. 1, no. 43, Art. no. 43, Sep. 2022.
- [9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
- [10] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, Austin, Texas, Jan. 2015, pp. 2267–2273.
- [11] P. Liu, X. Qiu, and X. Huang, "Recurrent Neural Network for Text Classification with Multi-Task Learning," May 2016.
- [12] L. Yao, C. Mao, and Y. Luo, "Graph Convolutional Networks for Text Classification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, Art. no. 01, Jul. 2019.
- [13] J. Li, G. Huang, J. Chen, and Y. Wang, "Dual CNN for Relation Extraction with Knowledge-Based Attention and Word Embeddings," *Computational Intelligence and Neuroscience*, vol. 2019, p. e6789520, Jul. 2019.
- [14] S. O. Alhumoud and A. A. Al Wazrah, "Arabic sentiment analysis using recurrent neural networks: a review," *Artif Intell Rev*, vol. 55, no. 1, pp. 707–748, Jan. 2022.
- [15] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," arXiv, May 19, 2014.
- [16] X. She and D. Zhang, "Text Classification Based on Hybrid CNN-LSTM Hybrid Model," in *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, Dec. 2018, vol. 02, pp. 185–189.
- [17] C. Yang, Y. Guo, X. Li, and B. Chen, "A Novel Method Using Local Feature to Enhance GCN for Text Classification," in *2021 11th International Conference on Intelligent Control and Information Processing (ICICIP)*, Dec. 2021, pp. 59–65.
- [18] S. Silvestri, F. Gargiulo, M. Ciampi, and G. De Pietro, "Exploit Multilingual Language Model at Scale for ICD-10 Clinical Text Classification," in *2020 IEEE Symposium on Computers and Communications (ISCC)*, Jul. 2020, pp. 1–7.
- [19] F. Chen, Z. Yuan, and Y. Huang, "Multi-source data fusion for aspect-level sentiment classification," *Knowledge-Based Systems*, vol. 187, p. 104831, Jan. 2020.
- [20] Z. Lu, P. Du, and J.-Y. Nie, "VGCN-BERT: Augmenting BERT with Graph Embedding for Text Classification," in *Advances in Information Retrieval*, Cham, 2020, pp. 369–382.
- [21] S. Zheng and M. Yang, "A New Method of Improving BERT for Text Classification," in *Intelligence Science and Big Data Engineering. Big Data and Machine Learning*, Cham, 2019, pp. 442–452.
- [22] W. E. N. Chaodong, Z. Cheng, R. E. N. Junwei, and Z. Yan, "Patent text classification based on ALBERT and bidirectional gated recurrent unit," *Journal of Computer Applications*, vol. 41, no. 2, p. 407, Feb. 2021.
- [23] A. H. Roudsari, J. Afshar, C. C. Lee, and W. Lee, "Multi-label Patent Classification using Attention-Aware Deep Learning Model," in *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Feb. 2020, pp. 558–559.
- [24] Y. Liu, P. Li, and X. Hu, "Combining context-relevant features with multi-stage attention network for short text classification," *Computer Speech & Language*, vol. 71, p. 101268, Jan. 2022.
- [25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv, May 24, 2019.