Multi-label Text Classification Based on BERT and Label Attention Mechanism

Xinghong Chen*
College of Computer Science and Technology
Wuhan University of Science and Technology
Wuhan, China
202013703054@wust.edu.cn

Yi Yin
College of Computer Science and Technology
Wuhan University of Science and Technology
Wuhan, China
yinyi@wust.edu.cn

Tao Feng
School of Business and Economics
Loughborough University
London, UK
samfung97@163.com

2023 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC) | 978-1-6654-5703-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/IPEC57296.2023.00073

***Abstract*—In natural language processing, multi-label text classification is a crucial task. Recently, many methods had introduced information related to labels, which had improved the classification effect of methods. This paper proposed a model that utilizes BERT and a label attention mechanism to effectively leverage the semantic information present in labels. Through fine-tuning of BERT, the textual and label data were transformed into vector representations, and then the label attention mechanism was employed to extract text features, which are more relevant to labels. Finally, a corresponding classifier was constructed to complete the classification task. Experiments show that compared with the baselines mentioned in the paper, the proposed method had an improvement on both AAPD and RCV1-v2 datasets, which proved the effectiveness of the proposed method in the multi-label text classification task.**

Keywords—multi-label text classification, natural language processing, attention mechanism, label semantics, BERT

I. INTRODUCTION

Multi-label text classification (MLTC) is applicable in various domains and has a broad spectrum of use cases, especially in the area of tag recommendation, sentiment analysis, and information retrieval. The goal of MLTC is to process the given text information so that it can be associated with one or more label instances. MLTC involves summarizing and abstracting text information from multiple perspectives, making the text information effectively utilized.

Currently, there are already many methods to solve MLTC problems, including Binary Relevance (BR)[1], Text-CNN[2], Sequence Generation Model for Multi-label Classification (SGM)[3], Label Embedding Attention Model (LEAM)[4] and et al. The BR method is an early approach used to solve MLTC problems, which treated each label as a separate binary classification problem, so that, MLTC problems can be solved via existing binary classification problems. Text-CNN is the first method to add convolutional neural networks (CNN) into text classification. Text-CNN used maximum pooling layers or average pooling layers to extract text features and used binary cross entropy loss function to train the model. However, it lacked the usage of label information. SGM is a multi-label classification method based on sequence to sequence model (seq2seq), which treated the MLTC problem as a sequence generation problem, and applied a novel decoder structure to solve it. LEAM regards MLTC as a joint embedding problem of text and labels. When given a text sequence, words with high correlation with labels will account for higher weight than words with weak relevance, to better capture text characteristics, reflecting the importance of label semantic information for multi-label classification of text.

This paper introduces a new method called BERT-LAM, which aims to enhance the effectiveness of text classification by combining the BERT model with the Label Attention Mechanism (LAM). The BERT model is known for its ability to capture contextual semantic information in text, while LAM is a technique that computes word and label embeddings jointly to identify text features that are most relevant to the labels. By combining BERT and LAM, the proposed method is expected to take advantage of the semantic information in text and highlight the most important information that contributes to label prediction.

The rest of the paper is structured into several sections. In Section 2, the related work in the field will be introduced. Section 3 will provide a detailed description of the proposed BERT-LAM model, including how it combines BERT and LAM to improve the efficiency of text classification. Section 4 will present the experimental details and results of BERT-LAM, including the dataset used, evaluation indicators, and comparative experimental results with other methods. Additionally, an ablation study will be conducted to assess the effectiveness of individual components of BERT-LAM. Finally, in Section 5, the paper will conclude with a summary of the work and potential future research directions in the field of text classification.

II. RELATED WORK

A. The definition of multi-label text classification

The MLTC task can be interpreted as a mapping between a given text and multiple labels through a specific classifier[5]. The mathematical relationship D={(x_i, y_i)|1≤i≤m} can be used to represent a MLTC task. The mapping f:X to Y can be obtained via the calculation of a designed classification model, which makes x_i correlated to y_i, where x_i represents a text instance within the text instances set X, and y_i represents a subset of the label instances set.

Data preprocessing is a fundamental step in natural language processing (NLP). Via converting text information into structured data, text information can be processed by computer. Generally, text processing has a fixed process, including word segmentation, stem extraction, part-of-speech recovery, etc.

Usually, text information is unstructured and cannot be processed by computers. Thus, it is necessary to vectorize the

978-1-6654-5703-3/23/$31.00 ©2023 IEEE
DOI 10.1109/IPEC57296.2023.00073

preprocessed text information and convert the text data into digital data, so that the computer can recognize the text information. A good text vector representation can greatly improve computational efficiency. Text vectorization methods can be divided into two types: discrete representation methods and distributed representation methods. One-hot coding method[6] and bag of word model (BOW)[7] are typical discrete representation methods. While, Distributed representation methods include co-occurrence matric[8], Word to Vector model (Word2Vec)[9], Glove[10], and so on.

Feature dimension reduction is also called feature extraction. The features of the text vector obtained by vectorization are sparse and have higher dimensions. After feature extraction, the redundant feature information can be removed, and effective feature information will be retained.

The classifier model can be obtained by feeding the preprocessed data into the classifier for training. Finally, according to the output of the validation and test set, the model is judged by the F1 score and other indicators.

B. Model structure of BERT

BERT is known as a popular pre-training model currently. Inspired by the bidirectional structure of the Elmo model and the special structure of the Transformer encoding part, BERT consisted of the encoding part of the Transformer with a bidirectional structure. Due to the special structure of BERT, it can calculate in a parallel method, which makes BERT have a high speed for calculation and also have the ability to stack with more neural networks.

One of the key differences between Transformer-based encoders and traditional Long Short Term Memory Networks (LSTM) is that Transformer encoders use a multi-head attention mechanism layer and a feedforward neural network. This allows the model to directly interact with front and back information in the same layer. This unique structure of BERT is what gives it strong context extraction abilities. Another innovative aspect of BERT is its input layer. The input vector of the BERT model is comprised of three components: a position embedding vector that represents the sentence, a segment embedding vector that identifies a sentence as belonging to the upper or lower sentence, and a token embedding vector that represents the meaning of the sentence. By adding these three vectors together, BERT is able to effectively capture the context of words. Fig. 1 displays the architecture of the BERT.

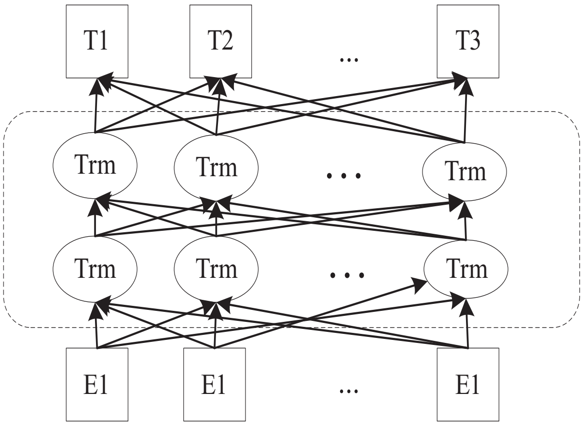

Fig. 1. Model structure of BERT

III. Methods

In this part, this paper will have an overview of the structure of the BERT-LAM model. As shown in Fig. 2, there are three main parts of BERT-LAM model, including the embedding layer, label attention mechanism, and classifier layer. Firstly, the word and label embeddings were got via BERT. Different from traditional multi-label text classification models, a label attentive model was added to make the words that are relevant to the labels weighted higher. Finally, BERT-LAM used full connection layers and the sigmoid function to get the label-predicting sequence.

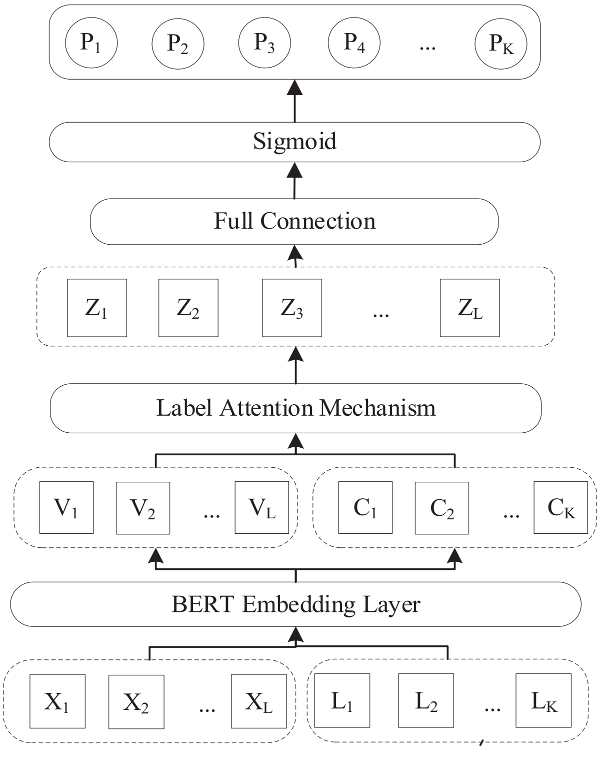

Fig. 2. The overview of BERT-LAM

A. Embedding layer

The embedding layer is designed to obtain the word and label embeddings through the BERT model, which can better capture the semantic and contextual information of the words. After the text information and labels are embedded through the BERT model, they can be mapped into the same dimensional space, making it easier to perform joint calculations between the text embeddings and label embeddings. As shown in the structure of the BERT-LAM, X_i represents the word vector of the i-th word in the text, and the length of text information is L. L_i represents the label vector of the i-th label, and there are a total of K labels. After feature extraction by the BERT model, the text context semantic vector representation and the label vector representation with contextual semantics are obtained. The word vector representation of the text is denoted as V=[V1, V2⋯VL], with a length of L, and the word vector representation of the labels is denoted as C=[C1, C2⋯CK], with a length of K.

$$\boldsymbol{Vi} = BERT(\boldsymbol{Xi}, \boldsymbol{si}, \boldsymbol{pi}) \quad (1)$$

$$\boldsymbol{Cj} = BERT(\boldsymbol{Lj}, \boldsymbol{sj}, \boldsymbol{pj}) \quad (2)$$

Formulas (1) and (2) show the way to obtain the word vector representation and the label vector representation. Here, s_i and s_j represent the values at corresponding positions in the segment vectors, which are used to record the position of the sentence where the word is located. p_i and p_j represent the values at corresponding positions in the position vectors, which are used to record the relative position between words. Necessary parameters are passed into the BERT model and after computation, the text and label word vector representations V and C with contextual semantic information are obtained.

B. Label attention mechainism

The label attention layer consists of four calculation steps. Firstly, cosine similarity is used to measure the compatibility between words and labels. The word vector representation is multiplied by the label vector representation, and the result is divided by the product of their second norms. Each value in the matrix G represents the correlation between the text vector and the label vector. The calculation is shown in formula (3).

$$\boldsymbol{G} = (\boldsymbol{V} \cdot \boldsymbol{C^T}) \oslash \left(||\boldsymbol{V}||_2 \cdot ||\boldsymbol{C^T}||_2\right) \tag{3}$$

To further obtain the correlation coefficients between labels and text, the maximum pooling calculation method is used to calculate the coefficient with the maximum correlation between the l-th word and the labels. Finally, the attention coefficient α between the word vector and the label vector is obtained with the normalized exponential function softmax. As shown in formulas (4) and (5), G_l represents the similarity vector that shows the similarity relationship between the l-th word and all labels in the similarity matrix G. M_l is obtained by maximum pooling calculation, which represents the coefficient with the maximum similarity between the l-th word and all labels. α is a vector in a vector space of dimension L, where each value represents the attention coefficient that the text vector generates for the classification result.

$$\boldsymbol{M_l} = max - pooling(\boldsymbol{G_l}) \tag{4}$$

$$\alpha_i = \frac{e^{\boldsymbol{Mi}}}{\sum_1^L e^{\boldsymbol{Mj}}} \tag{5}$$

After obtaining the label attention scores, the attention coefficients are allocated to the vector representation of each text feature using the label attention mechanism, which is used as the input of the multi-label classifier. The label attention mechanism can effectively represent the explicit relationship between text features and labels and strengthen the text features that are highly related to the labels. After weighing the label attention coefficients and the text vectors, the text features combined with label semantic information Z can be obtained. The calculation of Z is shown in formula (6).

$$\boldsymbol{Z} = \sum_L \alpha_i \cdot \boldsymbol{V_i} \tag{6}$$

C. Multi-label classifier

The last part of the model is a multi-label classifier layer, which uses a feedforward neural network to map the text features Z, which have been processed through the attention mechanism, to a space with the same dimension as the label vectors. Finally, BERT-LAM used the sigmoid function to obtain the predicted results for each label, as shown in equation (7). This calculation method allows the predicted values for labels to be independently distributed while mapping the predicted probabilities to between 0 and 1.

$$\boldsymbol{p_i} = sigmoid(f(\boldsymbol{Z}_i)) \tag{7}$$

IV. Experiments and analysis

The performance testing experiments of the BERT-LAM model were mainly conducted on two publicly available multi-label classification datasets, AAPD and RCV1-v2. This section will introduce the datasets, evaluation methods, and baseline models used in the comparative experiments first. Finally, this paper will present the experimental results and analyze them.

A. Introduction of datasets

AAPD: The Institute of Big Data Research at Peking University has provided a dataset called AAPD that consists of 55,840 abstracts from the field of computer science, along with their respective topics. Each abstract can be associated with multiple topics, and there are a total of 54 topic words available. The goal of multi-label text classification is to predict which academic topics are related to each abstract based on its content.

RCV1-v2: The RCV1-v2 dataset, provided by Lewis et al., is a publicly available English dataset consisting of over 800,000 manually classified news articles. The dataset was provided to researchers by Reuters Limited. Each news article in the dataset may be assigned to one or more topics, with a total of 108 different topics represented. This dataset has been widely used in research on text classification due to its large size and diversity of topics.

B. Evaluation methods

This paper mainly used binary cross entropy loss (BCE-Loss) and micro-F1 to evaluate the generalization ability of the model.

BCE-Loss also referred to as symmetric loss, is a metric used to evaluate the effectiveness of classification models. It calculates the degree of difference between the predicted values and the true values, and its value increases as the deviation between these values grows larger.

$$\text{Loss} = -\boldsymbol{Wn}[y \cdot ln(1 - p(x) + (1 - y) \cdot \ln(p(x))] \tag{8}$$

In this formula, p(x) is the predicted value of the model, y is the actual value of the training set, and **Wn** is the parameter of the model. Therefore, when the predicted value is the same as the actual value, the loss value is zero, and if the actual value is opposite to the predicted value, the loss value will be a large number. It can be seen that when the model performs very well, the loss value will approach zero infinitely.

Micro-F1 is represented by the weighted average of recall and precision. Recall and precision are calculated by true positive samples (TP), false positive samples (FP), and false negative samples (TN).

$$\text{P} = \frac{\sum_1^N TP}{\sum_1^N TP + \sum_1^N FP} \tag{9}$$

$$\text{R} = \frac{\sum_1^N TP}{\sum_1^N TP + \sum_1^N FN} \tag{10}$$

$$\text{F1} = \frac{2 \cdot P \cdot R}{P + R} \tag{11}$$

C. Comparative experiment

This paper carried out comparative experiments to evaluate the efficacy of the proposed method. The study compared it against several widely used multi-label text classification approaches, including BR, SGM, CNN-RNN[11], and MAGNET[12].

In this paper, the "BERT-base-uncased" pre-trained model was utilized, which consists of 12 Transformers encoding layers, a hidden layer dimension of 768, and 12 attention heads. As the length of texts varies, padding and truncation were employed to handle the text inputs. The AAPD dataset experiment used a text length of 128, while the RCV1-v2 dataset used a length of 126 based on the average text length of each dataset. To accelerate the training process, the Adam optimizer was used with an initial learning rate of 5×10^{-5}. Additionally, a learning rate warm-up strategy was adopted to gradually increase the learning rate during training for better performance. To prevent overfitting, dropout regularization with a probability of 0.1 was implemented. Table I shows the results of the comparative experiments.

TABLE I. COMPARATIVE EXPERIMENTAL RESULTS OF BERT-LAM AND BASELINE MODELS

Methods	*AAPD*			*RCV1-v2*		
	P	*R*	*F1*	*P*	*R*	*F1*
BR	0.644	0.648	0.646	0.904	0.816	0.858
SGM	0.746	0.659	0.699	0.887	0.850	0.869
SGM+GE	0.748	0.675	0.710	0.897	0.860	0.878
MAGNET	0.728	0.667	0.696	0.922	0.833	0.875
CNN-RNN	0.718	0.618	0.664	0.889	0.825	0.856
BERT-LAM	0.794	0.652	0.716	0.892	0.885	0.888

From the table, it can be seen that the SGM+GM model currently performs relatively well on the AAPD and RCV1-v2 datasets, with F1 evaluation metrics reaching 0.71 and 0.878 respectively, which is already a very impressive classification result. The proposed method of fusing BERT and label attention mechanism has improved the F1 evaluation metrics on both datasets by 0.6 percentage points and 1.0 percentage points, respectively, compared to SGM+GM. The improvement of BERT-LAM is notably superior compared to other methods. Compared with traditional binary correlation methods, the improvement of the proposed method on the AAPD dataset has reached 6 percentage points, which is also quite significant. In summary, the proposed method in this paper has effectively improved the solution for the text multi-label classification task and fully demonstrates the significant effect of introducing label information for the text multi-label classification task.

D. Ablation experiment

To confirm the effectiveness of the label attention mechanism, this paper conducted a set of ablation experiments. The ablation experiment used "LAM" as the ablation variable item and "NoLAM" as a representation of directly inputting the text vector into BERT and obtaining the classification prediction result through a fully connected layer and a sigmoid function. By comparing the performance of the "BERT-LAM" model and the "NoLAM" model, the results are shown in Table II.

TABLE II. ABLATION STUDY FOR BERT-LAM

Methods	*AAPD*			*RCV1-v2*		
	P	*R*	*F1*	*R*	*P*	*F1*
NoLAM	0.823	0.594	0.690	0.904	0.843	0.875
BERT-LAM	0.794	0.652	0.716	0.892	0.885	0.888

The results of the ablation experiment revealed that incorporating the label attention mechanism in the proposed BERT-LAM model led to a significant improvement in the performance of multi-label text classification. This finding suggests that the label attention mechanism is an effective approach for guiding the model to focus on text information that is most relevant to the labels.

E. Conclusion

In the research, this paper developed a solution to the problem of multi-label text classification by combining BERT, a powerful language model, with the label attention mechanism. By leveraging BERT to learn the features of both the text and labels and incorporating label attention to prioritize relevant text for each label, BERT-LAM achieved significant improvements in classification performance. The ablation experiment confirmed the effectiveness of the label attention mechanism in multi-label text classification.

Moving forward, enhancing the attention mechanism is necessary by exploring different attention types and refining the classifier model to better account for label relevance. These advancements will further optimize the efficiency of text classification.

ACKNOWLEDGMENT

This work was partly supported by Storage analysis and research of text-oriented big data, Information Storage System Ministry of Education Key Laboratory Open Fund.

REFERENCES

[1] Wu G, Zheng R, Tian Y, et al. Joint ranking SVM and binary relevance with robust low-rank learning for multi-label classification. Neural Networks, 2020, 122: 24-39.

[2] Guo B, Zhang C, Liu J, et al. Improving text classification with weighted word embeddings via a multi-channel TextCNN model. Neurocomputing, 2019, 363: 366-374.

[3] Yang P, Sun X, Li W, et al. SGM: sequence generation model for multi-label classification. arXiv preprint arXiv:1806.04822, 2018.

[4] Wang G, Li C, Wang W, et al. Joint embedding of words and labels for text classification. arXiv preprint arXiv:1805.04174, 2018.

[5] You R, Zhang Z, Wang Z, et al. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. Advances in Neural Information Processing Systems, 2019, 32.

[6] Rodríguez P, Bautista M A, Gonzalez J, et al. Beyond one-hot encoding: Lower dimensional target embedding. Image and Vision Computing, 2018, 75: 21-31.

[7] HaCohen-Kerner Y, Miller D, Yigal Y. The influence of preprocessing on text classification using a bag-of-words representation. PloS one, 2020, 15(5): e0232525.

[8] Nataraj L, Mohammed T M, Chandrasekaran S, et al. Detecting GAN generated fake images using co-occurrence matrices. arXiv preprint arXiv:1903.06836, 2019.

[9] Jatnika D, Bijaksana M A, Suryani A A. Word2vec model analysis for semantic similarities in english words. Procedia Computer Science, 2019, 157: 160-167.

[10] Sakketou F, Ampazis N. A constrained optimization algorithm for learning GloVe embeddings with semantic lexicons. Knowledge-Based Systems, 2020, 195: 105628.

[11] Wang R, Ridley R, Qu W, et al. A novel reasoning mechanism for multi-label text classification. Information Processing & Management, 2021, 58(2): 102441.

[12] Pal A, Selvakumar M, Sankarasubbu M. Multi-label text classification using attention-based graph neural network. arXiv preprint arXiv:2003.11644, 2020.