Proceedings of the 3rd International Conference on Computing Innovation and Applied Physics
DOI: 10.54254/2753-8818/31/20241133

Convergence of polarized self-attention with consistent rank Chinese text classification

Yetong Jin[1,2], Linfu Sun[1,2], Songlin He[1,2,3]

[1]School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, China
[2]Sichuan Provincial Key Laboratory of Manufacturing Industry Chain Collaboration and Information Support Technology, Southwest Jiaotong University, Chengdu, China

[3]sohe@swjtu.edu.cn

Abstract. Utilizing the powerful feature extraction capabilities of deep learning, a text classification algorithm with multi-dimensional and high-domain adaptability is designed in this study. This method enhances the model's understanding of topics and content by incorporating the Polarized Self-Attention (PSA) module, which strengthens the spatial structure and semantic features of textual information. The loss function is redesigned to assign smaller losses to misclassifications of neighboring categories, allowing the model to optimize classification accuracy while learning hierarchical structural information between categories. Finally, experimental verification is conducted on a publicly available news dataset, demonstrating improved results in text classification achieved by the proposed algorithm.

Keywords: Text Classification; Attention Mechanism; Loss Function

1. Introduction

Text classification, as a crucial information processing technique, plays a vital role in accurately and rapidly determining the category of textual information, making it a key task in natural language processing. Traditional methods for handling text classification tasks often involve two independent stages: feature extraction and classification. These methods primarily rely on frequency-based feature extraction techniques, making it challenging to uncover semantic information within the text. With the advancement of machine learning, models such as decision trees, Naive Bayes classifiers, support vector machines, and K-nearest neighbors have been widely applied in the field of text classification. However, these machine learning algorithms struggle to extract textual features and only serve as classifiers, resulting in limited text mining capabilities and room for improvement in classification accuracy.

In recent years, the development of deep learning technology has brought significant improvements to text classification. Convolutional neural networks capture local semantic information in the text, effectively identifying features at different positions through the sliding window mechanism of convolutional kernels [1]. Recurrent neural networks, with their cyclic structure, excel in capturing temporal information and long-range dependencies in the text, enhancing the model's understanding of context and emotional variations [2]. The introduction of attention mechanisms directs more attention to vocabulary or phrases crucial for the classification task [3]. Combining these methods, the application

DOI: 10.54254/2753-8818/31/20241133

of deep learning technology in text classification allows for a better understanding of the semantic structure of text, thereby achieving superior performance.

Addressing the aforementioned issues, this paper proposes a new classification model, PSA-CR (Polarized Self-Attention and Consistent Rank).

The main contributions of this paper are as follows: (1) Introducing a novel text classification model, PSA-CR, which captures semantic relationships between texts from multiple dimensions, enhancing the model's understanding of text semantics. (2) Incorporating the PSA module to selectively enhance the weight of spatial information, reducing unnecessary information interference through a parallel fusion of channel and spatial branch information. (3) Utilizing a consistent rank framework to align text feature distributions, reducing differences between different texts and ensuring classifier consistency. (4) Comparing with other advanced models, evaluating performance in terms of accuracy, precision, recall, and F1 score.

2. Related Work

2.1. Text Classification Tasks

In recent years, the rise of deep learning technology has led to new developments in the field of text classification. The introduction of CNN and RNN has improved performance by learning abstract features but still requires manual feature engineering. In 2018, Google's release of the BERT model marked a significant breakthrough in text classification research. BERT, based on the Transformer bidirectional encoder, pre-trains on a large-scale unlabeled text data, learning rich language representations and encoding contextual information bidirectionally into word embeddings [4]. Due to BERT's strong text representation capabilities, it has been widely applied in text classification tasks. For instance, Syaiful et al. used BERT as word embeddings, employed LSTM for aspect extraction, and utilized CNN for sentiment extraction, achieving a 2.04% increase in accuracy [5].

2.2. Attention Mechanisms and Variants

RNN and CNN, due to the existence of hidden vectors, result in models with a certain lack of interpretability. In response, Bahdanau and others introduced the Attention mechanism from image processing problems into natural language processing. When faced with a large amount of information to process, the attention mechanism filters key information through importance-weighted vectors, enhancing the efficiency of neural networks. Inspired by this, Wang et al. proposed an attention-based LSTM emotion classification model, achieving good results on public datasets [6]. Chen et al. enriched the attention model for semantic information in short texts by combining prior knowledge and introduced attention to short texts and concepts to measure the importance of concepts from these two aspects [7].

3. PSA-CR Classification Model

The PSA-CR text classification model proposed in this paper is illustrated in Figure 1, consisting of the word embedding layer, feature layer, and output layer. The specific details are outlined below.

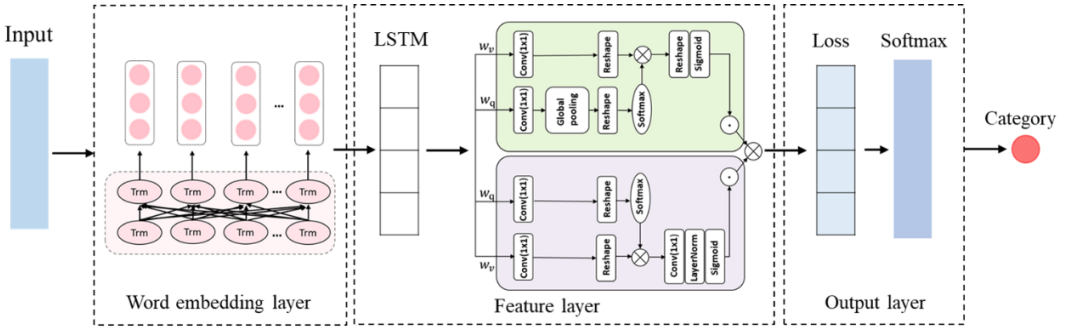

Figure 1. Model Architecture.

DOI: 10.54254/2753-8818/31/20241133

3.1. Word Embedding Layer

In this study, a pre-trained BERT model is employed as the word embedding layer. BERT is a deep bidirectional pre-trained language representation model based on the Transformer architecture. Compared to traditional word embedding methods, the BERT model excels in capturing semantic and contextual information within sentences, thereby enhancing the accuracy of text classification tasks. During the pre-training phase, the BERT model learns word and sentence representations through tasks such as word masking and sentence relationship prediction. Through this pre-training process, the BERT model gains a profound understanding of the complex semantics and syntactic structures present in the text.

Initially, the input text sequence undergoes tokenization, converting each word into its corresponding word vector. Subsequently, these word vectors are fed into the BERT model to obtain contextually relevant representations for each word. The distinctive feature of the BERT model, as compared to other pre-trained models, lies in its utilization of a bidirectional Transformer structure, leveraging the Encoder module for feature representation.

3.2. Feature Layer

The character vectors generated by the BERT model are used as the embedding layer for feature extraction, followed by the utilization of an LSTM model to extract text features. The essence of the attention mechanism lies in allocating different weights to various words in the text, allowing the model to selectively focus on more important word features based on these weight values.

In text classification tasks, the model's understanding of context and the key text window have a significant impact on the classification results. For a sentence to be classified, the classification target only occupies a limited portion, while the rest consists of words that may interfere with the classification. During the deep learning process, these interfering words contribute redundant information through operations like convolution, causing interference with the text classification results. Therefore, it is crucial to selectively enhance the weight of spatial information and reduce unnecessary interference. To achieve this goal, the PSA attention mechanism module is embedded into the feature layer in this study. The PSA module has two characteristics:

(1) In text classification tasks, many models enhance robustness and computational efficiency by reducing the resolution of features. However, this may lead to the loss of some detail information in the nonlinear parts of the text (referring to various complex grammatical and semantic structures in the text). Research indicates that the feature resolution of text is critical for text classification tasks. The polarized filtering mechanism of the PSA module aims to completely fold one dimension of text features while preserving high-resolution information in another dimension. When folding channel dimension features, it retains high-resolution information about the semantics or positions of the text, assisting in improving the accuracy of text classification tasks when dealing with text elements with different semantics or structures.

(2) To broaden the model's attention range for the smallest features in the text, the PSA module employs a softmax normalization process to enhance information, followed by applying a sigmoid function for projection mapping. In text processing, this process helps extend the model's attention range, effectively focusing on key elements in the text, thereby enhancing the performance of text classification.

The PSA module utilizes channel and spatial branches, merging them in a parallel manner.

The computation formula for the channel-only attention mechanism is as follows:

$$A^{ch}(X) = F_{SG}\left[W_{(z|\theta_1)\left(\delta_1(W_v(X))\right)} \times F_{SM}\left(\delta_2\left(W_q(X)\right)\right)\right] \tag{1}$$

Where X represents the input sequence, $A^{ch}(X) \in R^{sequence\_length \times 1}$; W_z, W_v, and W_q are 1×1 convolutional layers used to process textual features in the channel; δ_1 and δ_2 are two dimension-reduction operations, further reducing the dimension of features; F_{SG} represents the sigmoid function, F_{SM} represents the softmax function, and $\times$ denotes matrix dot product operation.

The computation formula for the spatial-only attention mechanism is:

DOI: 10.54254/2753-8818/31/20241133

$$A^{sp}(X) = F_{SG}\left[\sigma_3\left(F_{SM}\left(\delta_1\left(F_{GP}\left(W_q(X)\right)\right)\right) \times \left(W_v(X)\right)\right)\right] \tag{2}$$

Here, $A^{ch}(X) \in R^{1 \times sequence\_length}$, δ_3 is an up-sampling operation, restoring the feature dimension to match the length of the text sequence, F_{GP} is global pooling from the left, used to compress information across the entire sequence.

The parallel fusion of the two branches is obtained as follows:

$$PSA_P(X) = A^{ch}(X) \odot^{ch} X + A^{sp}(X) \odot^{sp} X \tag{3}$$

When introducing the PSA attention module into text classification tasks, the structure of the channel branch and the spatial branch provides a powerful method to better capture key information and features in textual data. The channel branch and spatial branch handle different aspects of the text, fully considering the multidimensional nature of the text, enhancing the accuracy and robustness of the task.

3.3. Output Layer

In text classification tasks, the challenge is to map textual data to discrete categories. Traditional classification methods directly associate text features with categories. However, in certain applications, there is an inherent ranking relationship between categories, which traditional multiclass classification methods fail to capture.

To address this issue, this model integrates a consistent rank framework, considering not only the classification loss but also the ranking relationship between predicted and true categories. It assigns smaller losses to misclassifications of neighboring categories, better reflecting the relative importance of categories.

Let $D = \{X_i, y_i\}_{i=1}^{N}$ represent the training dataset consisting of N training examples, where $x_i \in X$ represents the i-th training example, and $y_i \in Y = \{r_1, r_2, \ldots r_k\}$ represents the corresponding rank. First, extend the rank y_i to K-1 binary labels $y_i^{(1)}, \ldots y_i^{(K-1)}$, where $y_i^{(k)} \in \{0,1\}$ indicates whether y_i exceeds rank r_k. If $y_i > r_k$, the function value is 1; otherwise, it is 0. Using W to denote the weight parameters of the neural network, the output is represented as $g(x_i, W)$, and all nodes in the final output layer share the same weights.

$$\delta(z) = \frac{1}{(1 + \exp(-z))} \tag{4}$$

The sigmoid function, in Equation (4), transforms the output of the neural network into probability values, measuring the likelihood of a sample being assigned to a specific rank or exceeding a certain rank. The construction of the loss function consists of two parts: the first part pertains to correctly classified items, where $\lambda(k)$ is a weight parameter controlling the importance of different ranks. It penalizes misclassifications and encourages correct classifications. The second part pertains to misclassified items, penalizing the model's output when incorrectly classified.

$$L(W, b) = -\sum_{i=1}^{N}\sum_{k=1}^{K=1}\lambda(k)\begin{bmatrix}\log\left(\delta(g(x_i, W) + b_k)\right) y_i^{(k)} \\ +\log\left(1 - \delta(g(x_i, W) + b_k)\right)\left(1 - y_i^{(k)}\right)\end{bmatrix} \tag{5}$$

By introducing the consistent rank framework to improve the loss function, the model gains a better understanding and learning capability of the associations in the text. This enhancement is particularly effective when there is a ranking relationship between categories, leading to improved accuracy and domain adaptability.

4. Experiment

This section will first introduce the dataset, experimental environment, parameter settings, and Baseline experimental methods. Experiments were conducted on a public dataset to compare the text classification accuracy of the PSA-CR model proposed in this paper with the Baseline model.

DOI: 10.54254/2753-8818/31/20241133

4.1. Dataset

This study selected a portion of news text data from Tsinghua University's THUNews website, totaling 200,000 articles. Subsequently, 180,000 articles were randomly assigned to the training set, while 10,000 articles each were allocated to the test and validation sets. The news texts were categorized into 10 classes. Through ChatGPT data augmentation, the training set was expanded to 180,354 articles. The meanings and quantities of the data for each category after augmentation are summarized in Table 1.

Table 1. THUNews Data Information.

Category Name	ID	Data Quantity
Finance	0	16050
Reality	1	18043
Stocks	2	19051
Education	3	18030
Science	4	18030
Scoiety	5	18060
Politics	6	17030
Sports	7	18030
Game	8	18030
Entertainment	9	18970

4.2. Experimental Setup

The experiments in this paper were conducted on a server with the following basic environment: Ubuntu 18.04 operating system, Intel(R) Xeon(R) Platinum 8255C CPU, RTX 3080 Ti, Python environment version 3.8, and PyTorch version 1.11.0. Performance evaluation metrics include accuracy (A), recall (R), precision (P), and F_1 score.

4.3. Baseline Models

To validate the effectiveness of PSA-CR in text classification, this paper compares it with several popular and representative methods, as detailed below:

TextCNN: Captures local features in the input text through convolutional operations and extracts the most prominent features using a pooling layer. During this process, multiple convolutional kernels can capture different features. These features are concatenated and input into a fully connected layer, ultimately classified through a Softmax layer.

TextRNN: Mixes text input to Word2Vec to obtain text features, then uses TextRNN as a classifier. TextRNN obtains forward and backward hidden vectors through forward LSTM and backward LSTM modules, concatenates them, and finally classifies through a Softmax layer.

Transformer: Accepts word or character embeddings as input, where the encoder part is typically used to extract features from the input text.

Bert: Accepts word, character, or subword-level embeddings, utilizing bidirectional contextual information to represent contextual textual information.

Bert-CNN: Takes the output of the Bert model as input, then uses convolutional layers to capture local features. These features, combined with Bert's global information, are subsequently passed to a classifier for text classification tasks.

Bert-CNN-LSTM: Accepts the output of the Bert model, applies CNN layers to capture local features. Next, an LSTM module is used to obtain temporal information, combining the outputs of CNN and LSTM, and finally passing them through a classifier for text classification.

4.4. Experimental Results and Analysis

The experimental results are presented in Table 2. The results indicate a significant improvement of the proposed PSA-CR model compared to other methods.

DOI: 10.54254/2753-8818/31/20241133

Table 2. Experimental Results.

Model Name	A	C	R	F_1
TextCNN	89.63	89.70	89.63	89.61
TextRNN	88.44	88.61	88.44	88.42
Transformer	85.28	85.72	85.28	85.32
Bert	91.65	91.69	91.65	91.65
Bert-CNN	93.76	93.79	93.76	93.77
Bert-CNN-LSTM	93.95	93.96	93.95	93.95
Ours	**95.19**	**95.24**	**95.19**	**95.19**

The data in the table indicates that the PSA-CR model outperforms the deep learning models mentioned above on the news dataset. In comparison to TextCNN and TextRNN models, which use Word2Vec to obtain text features, Word2Vec has limited understanding of text context compared to the Bert model, as it assigns a fixed vector to each word. While Bert-CNN and Bert-CNN-LSTM models combine convolutional layers and temporal information to focus on global text features, they still only consider the extraction of local features during feature extraction, resulting in limited generalization ability. In contrast, the PSA-CR model can effectively focus on key content in the text, balancing contextual semantic understanding, and incorporating a robust error-punishment mechanism to enhance the overall classification performance of the model.

5. Conclusion

This paper proposes a fusion model, PSA-CR, combining polarized self-attention mechanisms and optimized loss functions for text classification tasks. Initially, the Bert model is used as the word embedding layer to transform discrete vocabulary in text data into continuous vector representations. Then, LSTM is employed to concatenate the PSA attention mechanism module for text feature extraction. Finally, an optimized loss function guides the classifier in optimizing adjustments to complete the classification task. The model not only integrates multi-dimensional features of text sentences but also enhances the model's adaptability to cross-domain classification tasks. Experimental results on a public dataset, compared with popular models, demonstrate that PSA-CR exhibits excellent classification performance.

This work is supported by the New Interdisciplinary Cultivation Fund (YH15001124322133, YH1500112432319) with Southwest Jiaotong University, Sichuan, China.

References

[1] LIU P F, QIU X P, HUANG X J. Recurrent neural networkfor text classification[C]//Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016: 2873-2879.

[2] KIM Y. Convolutional neural networks for sentence classification[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014: 1746-1751.

[3] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[4] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Bidirectional Encoder Representations from Transformers. arXiv preprint arXiv:1810.04805.

[5] Syaiful Imron, Setiawan, E. I., Joan Santoso, & Mauridhi Hery Purnomo. (2023). Aspect Based Sentiment Analysis Marketplace Product Reviews Using BERT, LSTM, and CNN.

[6] Wang Y, Huang M, Zhu X, et al. Attention-based LSTM for aspect-level sentiment classification[C]//Proceedings of the 2016 conference on empirical methods in natural language processing. 2016: 606-615.

[7] Chen J, Hu Y, Liu J, et al. Deep short text classification with knowledge powered attention[C]//Proceedings of the AAAI conference on artificial intelligence. 2019, 33(01): 6252-6259.