Collaborative filtering algorithm based on big data mining

Wang Gang
Nanjing University of Posts and Teleconmmunications
Nanjing,China
ID:1023041105

***Abstract*—In the task of text classification, due to the inaccuracy of Chinese text classification relation and the inability of static word vector to interpret text well. This paper proposes a weighted fusion model based on Ernie-cnn and attention mechanism. In this model, a pre-trained Ernie (Enhanced Representation through Knowledge Integration) model is used to obtain the language Representation of input text encoding, and then the important features of sentences are extracted by convolution, attention weighting and pooling. Finally, the extracted features are mapped to the final classification prediction results through the full connection layer. This process takes full advantage of the semantic representation capabilities of the Ernie model and the local feature extraction capabilities of the convolutional neural network, and by fusing key features of the attention mechanism that influence the final outcome. The experimental results on two open datasets of THUCNews and Baike 2018qa and one self-built datasets of industrial supplies show that the accuracy is 94.62% , 86.34% and 93.73% respectively on three datasets, all are better than the existing TEXTCNN, Bert and other models.**

***Keywords*—Ernie; attention mechanism; weighted fusion, text classification**

I. INTRODUCTION

In today's era of information explosion, massive amounts of text data are rapidly emerging. The efficient processing and accurate classification of these data are crucial for promoting the development of information technology and artificial intelligence. Especially in the context of the popularization of online shopping, the amount of product data is huge and the variety is rich. In order to facilitate users to browse web pages, search and improve the user experience, it is crucial to classify the title information of industrial products.

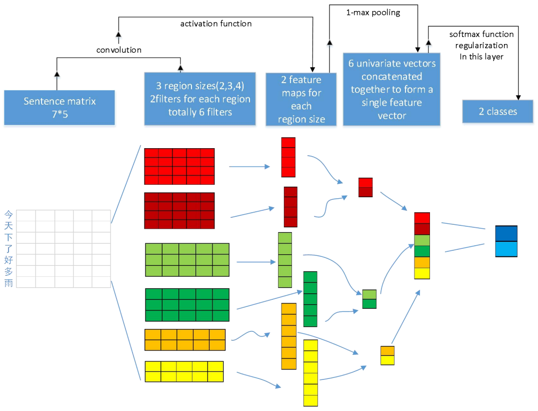
Traditional manual classification methods can no longer meet the needs of text classification. In this case, text classification using techniques such as natural language processing , data mining, and machine learning has become a widely adopted method. Especially with the rise of deep learning, it has demonstrated powerful capabilities in text classification. The application of deep learning in industrial commodity text classification has significant advantages compared to traditional machine learning methods. For example, it can directly process raw text data, avoiding the tedious feature engineering process. Meanwhile, deep learning models can learn complex semantic and contextual information, enabling better processing of large-scale and complex industrial commodity text data.

This study aims to explore and apply the advantages of deep learning methods in industrial commodity text classification tasks. We will propose an efficient and accurate industrial product text classification method by integrating technologies from natural language processing, deep neural networks, and representation learning. By constructing training datasets and models that adapt to the characteristics of the industrial field, we aim to improve the efficiency and accuracy of industrial commodity text classification.

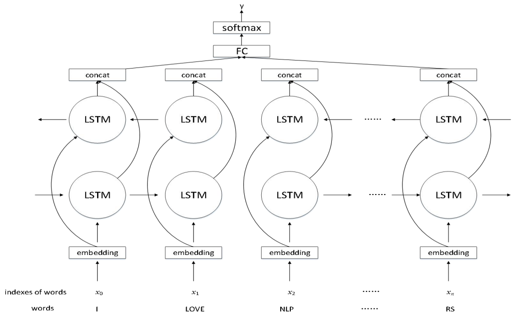
II. RELATED WORK

In order to obtain the optimal experimental results, this article conducted experiments on multiple models and compared the results. The following is an introduction to the models used in this article:

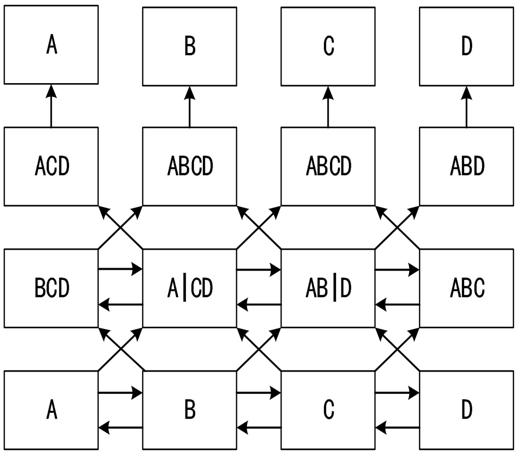
TextCNN is a fundamental and important model in text classification. Convolutional neural networks (CNN) use convolutional layers, pooling layers, and fully connected layers to form a model in text classification. The model structure is shown in Figure 3.1. The main process is as follows: Firstly, the text is transformed into a set of word vector feature matrices composed of words, which are used as inputs to the model. Then, the feature matrix is input into the convolutional layer for one-dimensional convolution operation, and local features are extracted using a convolutional filter to obtain a new set of feature matrices. The number of convolution kernels is determined by the model, and each convolution kernel slides on the feature matrix and performs convolution operations. Next, input the new feature matrix into the pooling layer for downsampling to reduce the dimensionality of features and retain important information. Finally, the convolved and pooled features are input into the fully connected layer for classification, and the results are obtained through the fully connected layer. This process utilizes convolutional layers to extract local features of text, and further processes and classifies them through pooling and fully connected layers.



TextRNN is a text classification model based on Recurrent Neural Network (RNN). It can capture contextual information and semantic relationships in text through sequence modeling. RNN can sequentially read text sequence data and has a certain level of memory ability. The RNN model for this experiment selected the bidirectional long short-term memory network BiLSTM . LSTM is a variant of RNN, which controls the processing of data information by controlling the three gate structures of forget gate, input gate, and output gate, as well as the state of cell units, thus avoiding the problem of gradient vanishing. BiLSTM has a larger receptive field and bi-directional extraction of semantic association information to obtain high-level feature representations. When modeling, a set of vectors are used as inputs for forward and backward LSTM:

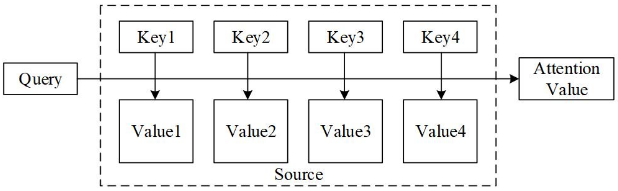


The BERT model is a language model built on a bidirectional Transformer, which can discover the interrelationships between words. This bidirectional process is shown in the following figureThe following figure shows the core components that make up taste:



Transformer is composed of 6 encoders and 6 decoders stacked together. The Transformer receives a vector sequence and outputs the processed data sequence. After being processed by 6 encoders, it is fed into 6 decoders for decoding. The core of the encoder and decoder is the attention mechanism, which can understand the overall meaning of a sentence based on the key points in the sentence. By calculating the attention distribution of the Key and integrating it into the Value, the attention value can be calculated. The principle and formula are as follows:

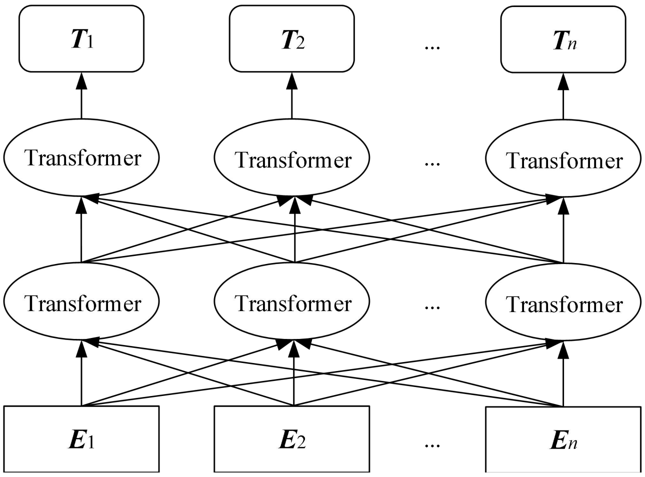
$$Attention(Q,K,V) = softmax\left(\frac{Q\cdot K^T}{\sqrt{d_k}}\right) \tag{4}$$



Where is the input word vector matrix and is the dimension of the input vector. In the Bert calculation process, the Transformer encoder directly connects any two words in a sentence through a one-step calculation, and weighted the representations of all words. To shorten the distance between long-distance dependencies and improve the effective utilization of features.

In order to better obtain semantic information in Chinese texts and make text classification more accurate, this paper introduces the ERNIE pre training model to formulate training strategies for texts and better extract feature information from texts. At the same time, in order to extract more important feature information, this paper incorporates CNN convolutional neural networks and self attention mechanisms. Self attention only focuses on the input of its own information. After pooling before convolution, the self attention mechanism is fused into feature vectors extracted by different sized convolution kernels, fully utilizing the self attention mechanism to enhance the model.

Dynamic word vector generation is an advantage of ERNIE pre trained models compared to static word vectors. Compared to static word vectors, ERNIE's dynamic word vectors more accurately capture contextual information, effectively avoiding the occurrence of polysemy. Ernie's model architecture is similar to Bert, both based on improvements to bidirectional converters, as shown in the following figure:



The training methods of the ERNIE model and BERT model are different from traditional word vector models. They used Mask Language Model (MLM) to complete training. MLM randomly masks some content in a sentence and predicts the position of masked text using unmasked text. In this way, the model can learn the feature information of each position and the semantic information of the context.

However, due to the differences between Chinese and English, the ERNIE model has been improved on the basis of the BERT model. The BERT model masks words on a word by word basis, but in Chinese, individual words and their constituent words often have different meanings, and the frequency of word occurrence is relatively high. If masking training is only conducted on a word by word basis, it will lose the feature information between words and between words, and it will not be able to learn the information and knowledge expressed by the word in the sentence well. Therefore, the ERNIE model introduces a masking training strategy based on words and entities.

III. Solutions

The overall architecture of the model proposed in this article first converts the sorted Chinese text data into input vectors and inputs them into the ERNIE pre trained model. The model generates dynamic word vectors through ERNIE and concatenates them with one-dimensional entity position vectors to mark the position of entities in sentences, providing more effective information for the model. Next, the concatenated word vector matrix is used to extract feature vectors at different levels through convolutional layers with different kernel sizes of 2, 3, and 4. Subsequently, these feature vectors are input into the attention layer, and weights are reassigned by calculating the attention probability distribution to obtain the weighted feature vectors. Finally, the feature vectors output by the attention layer are averaged and pooled through a pooling layer. The dimensionality reduced vectors are then processed through a fully connected layer and a softmax classifier to generate the final relationship extraction result.

The previously processed word vector matrix is the input to this layer, and new local features are obtained through convolution operations.

$$c_i = f\left(W \cdot E'_{i:i+h-1} + b\right)$$

f represents a nonlinear function，b ∈ R is a bias term，W ∈ $R^{h\ k}$ is a convolutional kernel，h represents the length of the site selection range，Change the h can get different N-grams，k reprents the word vector dimension of each word，it is 769 here.So we get N − h + 1 new feature，corresponding feature mapping C_m:

$$C_m = [c_1, c_2, \cdots, c_{N-h+1}]$$

The self attention mechanism used in this paper is different from the traditional attention mechanism. The traditional attention mechanism needs to introduce information other than the input data to calculate the weight, while the self attention focuses on the input vector itself, and learns to update parameters by analyzing its own information. By combining convolutional neural networks and self attention mechanisms, it is possible to better analyze the semantic features of text. By continuously iterating and updating, the weight of more important features in relation extraction tasks is increased, thereby filtering and reducing the impact of irrelevant features on relation extraction. The weight matrix of the self attention mechanism is composed of normalized values calculated by the softmax function.Calculate as follows:

$$A = softmax(u \tanh\ (W_a C^T + b))$$

The role of the pooling layer is to reduce dimensionality by integrating feature vectors, thereby reducing the number of parameters in subsequent fully connected layers, improving the running speed of the model, and preventing overfitting. This article adopts maximum pooling, as shown in the following formula:

$$P = max\_pooling(M)$$

Then input the integrated feature information into the fully connected layer to obtain the final output F_c:

$$F_c = \sigma\left(W_f P + b_f\right)$$

IV. Experimental Introduction

The dataset crawled in this experiment is industrial products, collected from the three major e-commerce platforms of JD, Suning, and Tmall, with a total of 643963 pieces of data. After preprocessing, the data amount is 449271 pieces, and there are 17 types of labels, including cleaning products, HVAC lighting, instruments and meters, mechanical accessories, chemicals, tools, welding and fastening, experimental supplies, labor protection, safety and fire protection, storage and packaging, furniture installation, industrial control and distribution The label category information for home appliance installation, lighter and smoking equipment, nursing and protective equipment, and agricultural machinery and tools is shown in the table below：

label	Category	Datas
0	Cleaning supplies	33175
1	HVAC lighting	7963
2	Instrumentation	34038
3	Machinery Parts	23181
4	Chemical	27325
5	Tool	29371
6	Welding fastening	18422
7	Laboratory supplies	14441
8	Labor protection	38676
9	Safety and fire protection	29149
10	Storage packaging	26607
11	Furniture installation	20952
12	Industrial control distribution	29651
13	Home appliance installation	25142
14	Lighter smoking set	17687
15	Nursing protective equipment	37656
16	Agricultural machinery and tools	35862

The above data is divided into training set, validation set, and test set in a 6:2:2 ratio.

In order to evaluate and validate the performance and generalization ability of the model, additional experiments were conducted on two publicly available Chinese datasets, THUCNews and Baike2018qa. The detailed information of the three datasets is shown in the table below.

Data Set	Label Category	Data
THUCNews	10	200000
Baike2018qa	13	379649
ss	17	449271

Data preprocessing mainly includes the following steps:

(1) Data cleaning. Missing value removal: Remove the occurrence of empty text from the crawled data samples.

Repetitive value removal: When removing duplicates, it is best to use multiple columns as references and not judge duplicates based solely on one dimension.

Unuseful text removal: Remove numbers, etc.

(2) Mechanical compression to remove words: Remove duplicate characters from text information, such as "danger" in "dangerous goods signs, warning signs, and safety signs". Short sentence filtering: Remove text with a length of less than 4 characters.

(3) Jieba segmentation: Perform segmentation analysis on these texts, analyzing specific categories of keywords.

To verify the practicality of the model, this article uses accuracy Acc, accuracy P, recall R, and F1_ Score is used to evaluate the effectiveness of the model , and accuracy is the most basic evaluation indicator in text problems. The formula definition is shown in Figure 11; Accuracy is the ratio of the amount of correctly predicted data to the amount of correctly predicted data, measuring precision. The formula is defined as 12; Recall rate is the ratio of the amount of data predicted to be correct to the actual amount of data classified to be correct, measuring recall rate. The formula is defined as 13; F1_ Score is the harmonic mean of precision and recall, and the formula is defined as follows:

$$Acc = \frac{TN + TP}{TN + TP + FN + FP}$$

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

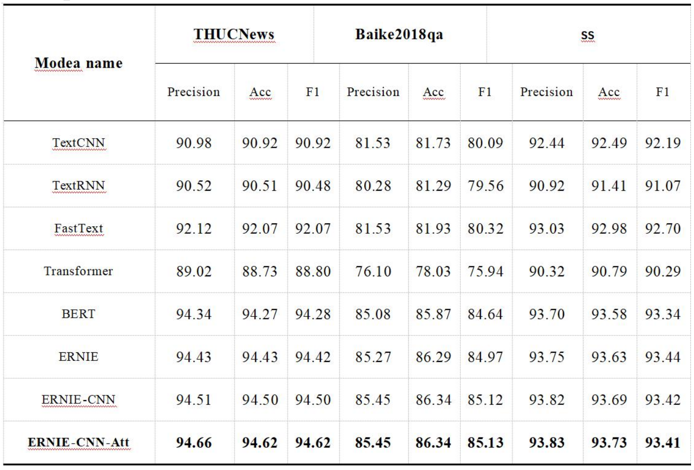
$$F1 = \frac{2 \times P \times R}{P + R}$$

TP represents the number of positive classes predicted as positive classes; TN represents the number of predicted negative classes as negative classes; FP represents the number of negative classes predicted as positive classes; FN represents the number of positive classes predicted as negative classes.

Firstly, conduct experimental comparisons of the above models on a self built dataset, comparing six evaluation indicators: loss rate, accuracy, precision, recall, F1 value, and time. The experimental comparison results are shown in the table below:

Model name	Loss	Acc	Precision	recall	F1	Time
TextCNN	0.23	92.49%	92.44%	92.02%	92.19%	0:04:00
TextRNN	0.25	91.41%	90.92%	91.26%	91.07%	0:02:06
FastText	0.2	92.98%	93.03%	92.52%	92.70%	0:14:11
Transformer	0.34	90.79%	90.32%	90.40%	90.29%	0:08:12
BERT	0.2	93.58%	93.70%	93.08%	93.34%	1:01:30
ERNIE	0.18	93.63%	93.75%	93.23%	93.44%	0:59:00
ERNIE-CNN	0.18	93.69%	93.82%	93.31%	93.42%	1:01:30
ERNIE-CNN-Att	**0.17**	**93.73%**	**93.83%**	**93.15%**	**93.41%**	**1:08:22**

To verify the effectiveness of the model, comparative experiments were conducted on three different datasets, and the experimental results are shown in the table below:



Modea name	THUCNews			Baike2018qa			ss		
	Precision	Acc	F1	Precision	Acc	F1	Precision	Acc	F1
TextCNN	90.98	90.92	90.92	81.53	81.73	80.09	92.44	92.49	92.19
TextRNN	90.52	90.51	90.48	80.28	81.29	79.56	90.92	91.41	91.07
FastText	92.12	92.07	92.07	81.53	81.93	80.32	93.03	92.98	92.70
Transformer	89.02	88.73	88.80	76.10	78.03	75.94	90.32	90.79	90.29
BERT	94.34	94.27	94.28	85.08	85.87	84.64	93.70	93.58	93.34
ERNIE	94.43	94.43	94.42	85.27	86.29	84.97	93.75	93.63	93.44
ERNIE-CNN	94.51	94.50	94.50	85.45	86.34	85.12	93.82	93.69	93.42
ERNIE-CNN-Att	**94.66**	**94.62**	**94.62**	**85.45**	**86.34**	**85.13**	**93.83**	**93.73**	**93.41**

The above experimental results indicate that the ERNIE model and BERT model achieved good results, but the Transformer model based on self attention mechanism also achieved the worst results in this experiment. TextCNN and TextRNN take less time compared to other models, but their classification performance ranks last among many models. FastText belongs to the middle level, with time consumption and accuracy ranking in the middle. The effect of the ERNIE-CNN Attention model on the Baike2018qa dataset and the self -developed dataset is not significant compared to the ERNIE-CNN model. This may be due to the fact that the Baike2018qa dataset is a question answering dataset with unclear feature effects, and the self-developed experimental dataset is industrial product title text. The quality of the dataset is average, and the attention feature extraction is not accurate. On the news text THUCNews dataset with obvious features, the accuracy increased by 0.15, the accuracy increased by 0.12, and the F1 value increased by 0.12.

V. CONCLUSION

This article analyzes the experimental results of six models, TextCNN, TextRNN, FastText, Transformer, BERT, and ERNIE, on a self built dataset. It is found that the effect of ERNIE is better than other models. This is because the ERNIE word vector is trained by masking phrases based on the characteristics of Chinese, so the ERNIE model is chosen to improve. On the basis of the ERNIE model, the CNN model's ability to extract features from input data was utilized, and the emphasis was placed on exploring the feature expression effect of the ERNIE model combined with attention mechanism on Chinese text. The improved model ERNIE-CNN Attention proposed in this paper was obtained. Although the model has improved in accuracy, recall, and F1 value, the experimental improvement is not significant due to the similar model architecture of ERNIE and BERT, which are both built on the basis of bidirectional Transformers. The model can be seen to be effective on the high-quality public dataset THUCNews. Therefore, optimizing the proposed model while processing the dataset to highlight its features will be the focus of future research in this paper.

REFERENCES

[1] Chowdhary K R. Natural language processing[J]. Fundamentals of artificial intelligence, 2020:603-649.

[2] Kowsari K, Jafari Meimandi K, Heidarysafa M, et al. Text classification algorithms: A survey[J]. Information, 2019, 10(4): 150.

[3] Minaee S, Kalchbrenner N, Cambria E, et al. Deep learning--based text classification: a comprehensive review[J]. ACM Computing Surveys (CSUR), 2021, 54(3): 1-40.

[4] Kim Y. Convolutional neural networks for sentence classification[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar. Stroudsburg, PA, USA: Association for Computational Lin-guistics, 2014: 1746-1751.

[5] Mnih V, Heess N, Graves A. Recurrent models of visual attention[J]. Advances in Neural Information Processing Systems,2014,27.

[6] Peters M E, Neumann M, Iyyer M, et al. Deep Contextualized Word Representations[C]. NAACL 2018, 2018.

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

[8] Kenton J D M W C, Toutanova L K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. Universal Language Model Fine-Tuning for Text Classification, 2018, 278.

[9] Sun Y, Wang S, Li Y, et al. Ernie: Enhanced representation through knowledge integration[J]. arXiv preprint arXiv:1904.09223, 2019.

[10] Chen Y. Convolutional neural network for sentence classification[D]. University of Waterloo, 2015.

[11] Liu P F, Qiu X P, Huang X J. Recurrent neural network for text classification with multi-task learning [C]// Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016: 2873–2879.

[12] Chen J, Weihua L I, Chen J I, et al. Bi-directional long short-term memory neural networks for Chinese word segmentation[J]. Journal of Chinese Information Processing, 2018, 32(2): 29-37.

[13] Joulin A, Grave E, Bojanowski P, et al. FastText. zip: Compressing text classification models[J]International Conference on Learning Representations, 2016.