Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 2, No. 3, 2022

Improved Text Matching Model Based on BERT

Qingyu Li, Yujun Zhang *

School of Computer and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China
* **Corresponding author**: Yujun Zhang

Abstract: Text matching is a basic and important task in natural language understanding, this paper proposes a new model BBMC for the problem of insufficient feature extraction ability of existing text matching models, which integrates BiLSTM and multi-scale CNN on the basis of BERT. First, the word embedding representation of the text is obtained by the BERT, and then the semantic features of the text are further extracted by the double-layer BiLSTM, followed by the multi-scale CNN model, the key local features are extracted, and finally the linear and SoftMax function are used to classify. Experimental results on the LCQMC dataset show that the BBMC has been improved to a certain extent compared with other methods, and the accuracy on the test set can be best achieved 88.01%.

Keywords: Bert; BiLSTM; CNN; Text Matching; NLP.

1. Introduction

As a core problem in natural language understanding , text matching is to take two texts as inputs and predict their relationship categories or correlation scores by understanding their respective semantics . Usually, this task is treated as a binary task. First, rich text feature information is obtained through various methods, and then the extracted features are classified. Many tasks in natural language processing can be regarded as text matching tasks, so it is particularly important to study efficient text matching methods. Traditional machine learning methods need to rely on manual access to the shallow features of the text, which can not well represent the semantic information of the text.

With the rapid development of deep learning, a large number of text matching models based on deep learning have emerged, which can be roughly divided into three types: representation based method, interaction based method and pretraining language model based method. The representation based method obtains the semantic representation of text in various ways for matching calculation. In 2013, Huang et al. proposed the DSSM model , which is the pioneering work of deep learning in text matching tasks. Later, with the popularity of CNN (Convolutional Neural Network, CNN) and LSTM (Long Short Term Memory, LSTM), etc., Shen et al. introduced CNN into the DSSM model to solve the problem of losing context information in DSSM . Palangi et al. introduced LSTM into DSSM to capture long-term context and proposed LSTM-DSSM model . This kind of model only obtains the semantic representation of the text, but does not take into account the interaction information between texts.

The interaction based method realizes text interaction through various attention mechanisms to obtain context information. Hu et al. use one-dimensional convolution to focus on adjacent word vectors for two pieces of text respectively, then combine the two tensors obtained after convolution, and finally propose the ARC-II model using the classification method of multilayer perceptron. Yin et al. introduced the attention mechanism on the basis of CNN, used the attention weight to fuse the output of the convolution layer, and finally proposed the ABCNN model . Pang et al. used dot product operation on word vectors of two texts to interact, and then proposed MatchPyramid model by using convolution and pooling to extract features. Chen et al. proposed a simple and efficient semantic matching network ESIM by using bi-directional recurrent neural network BiLSTM to obtain context representation and introducing attention mechanism to strengthen the interaction between texts. In order to solve the problem of insufficient interaction between texts, Wang et al. proposed a multi angle matching model BiMPM, which uses a variety of information interaction methods to improve the degree of text interaction. Although the above model can make use of the semantic information and interactive information between texts, the traditional static text semantic acquisition methods such as word2vec and Glove are still used in the semantic coding stage, which can not express the semantic information of text well.

The method based on pretraining language model refers to the unsupervised way to train the language model in large corpus in advance, and then load the pretrained model weights to finetune the specific downstream tasks. In 2018, the BERT (Bidirectional Encoder Representations from Transformers) model based on bidirectional Transformer came out, refreshing the list of many tasks in NLP field. Therefore, the method based on the pretraining language model has become a research hotspot in recent years. So since 2018, many researchers have improved the BERT model [16, 17, 18]. Cui et al. changed the pretraining method on the basis of BERT model and trained the Chinese-wwm-bert model for Chinese tasks on a large Chinese corpus [19, 20]. This model has achieved good results in Chinese natural language processing tasks, including text matching tasks. However, the text semantic information obtained by fine-tuning specific text matching tasks using BERT model alone is insufficient, Therefore, based on this problem, this paper proposes a method based on pretraining BERT model fusion of BiLSTM and multi-scale CNN to remodel the text matching task, further extract global semantic information and local semantic information to improve the accuracy of text matching.

2. Our Method

The model proposed in this paper mainly obtains the word vector representation of the text with context semantic

information through BERT, then further obtains the semantic information of the text through the BiLSTM model, and then the key information of different scales obtained through CNN with different size convolution cores, and finally classifies through the classification layer. The overall architecture is shown in the following, see Figure 1.

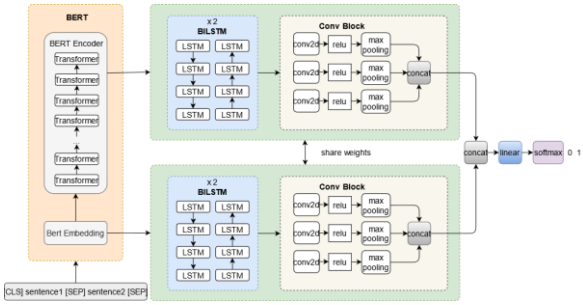

Figure 1. Overall architecture Diagram

(1) BERT

According to the characteristics of the input layer of the BERT model, this paper first separates the two sentences with [CLS] and [SEP], and then gets the word embedding of the text through the BERT Embedding layer. It consists of three parts, namely Token Embedding, Segment Embedding, and Position Embedding. The calculation formula is as follows:

$$E = E_{token} + E_{seg} + E_{pos} \tag{1}$$

Input the word embedding E obtained from BERT Embedding layer into the BERT coding layer, and then the sum of hidden layer vectors output by the last four layers of transformer in BERT coding layer is selected as the semantic feature H extracted from BERT model. The calculation formula is as follows:

$$H = \sum_{i=9}^{12} hidden\_state_i \tag{2}$$

Where i represents the number of layers of the transformer, hidden_states represents the hidden layer vector obtained through the ith transformer. After calculating H, input E and H into the subsequent layers to further extract the text feature information.

(2) BiLSTM Blocks

This module consists of two layers of BiLSTM. The architecture of BiLSTM is shown in the following, see Figure 2.

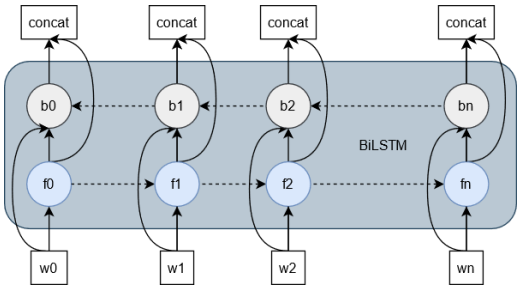

Figure 2. BiLSTM

where f represents forward LSTM and b represents backward LSTM. The calculation formula is as follows:

$$\begin{aligned} f_i &= \overrightarrow{\mathrm{LSTM}}(f_{i-1}, w_i) \\ b_i &= \overleftarrow{LSTM}(b_{i+1}, w_i) \\ t_i &= f_i \oplus b_i \\ T &= BiLSTM(W) = \sum_{i=1}^{n} t_i \end{aligned} \tag{3}$$

Where $\overrightarrow{LSTM}$ represents forward LSTM unit, $\overleftarrow{LSTM}$ represents backward LSTM unit, $\oplus$ represents splicing operation, W_i represents The ith word, f_i represents the output of W_i through forward LSTM, b_i represents the output of W_i through backward LSTM, t_i represents the output of W_i through BiLSTM, T represents the output of the whole sentence W after passing through BiLSTM.

In this paper, the word embedding E and semantic feature H obtained from the BERT layer are input into this layer respectively to obtain the overall semantic information TE and TH, according to the above BiLSTM calculation formula, TE and TH formulas are obtained as follows:

$$\begin{aligned} T_{E1} &= BiLSTM(E) \\ T_E &= BiLSTM(T_{E1}) \\ T_{H1} &= BiLSTM(H) \\ T_H &= BiLSTM(T_{H1}) \end{aligned} \tag{4}$$

Where T_{E1} represents the feature obtained after passing through the first layer of BiLSTM, and then input it into the second layer of BiLSTM to obtain T_E, and the Calculation of T_H is the same as above.

(3) Conv Block

This layer is a multi-scale convolution block. CNN with different convolution kernel sizes is used to further process the semantic features extracted by BiLSTM to obtain the local key information of the text. Since the dimensions of the semantic features TE and TH extracted from the BiLSTM layer are all [batch_size, seq_len, 768], and the length of the phase mostly 2 to 4 characters, this layer selects two-dimensional convolutions with convolution kernels of 2 * 768, 3 * 768, and 4 * 768 respectively to extract local key features, and then uses the relu activation function to modify them, and further compresses the features by using the maximum pooling method to remove redundant information. Finally, the three convolution features are spliced to obtain M, and the calculation formula is as follows:

$$\begin{aligned} M_{E1} &= mp(relu(conv(T_E, 2*768))) \\ M_{H1} &= mp(relu(conv(T_H, 2*768))) \\ M_{E2} &= mp(relu(conv(T_E, 3*768))) \\ M_{H2} &= mp(relu(conv(T_H, 3*768))) \\ M_{E3} &= mp(relu(conv(T_E, 4*768))) \\ M_{H3} &= mp(relu(conv(T_H, 4*768))) \\ M_E &= concat(M_{E1}, M_{E2}, M_{E3}) \\ M_H &= concat(M_{H1}, M_{H2}, M_{H3}) \\ M &= concat(M_E, M_H) \end{aligned} \tag{5}$$

Where conv represents the convolution operation, and the two parameters represent the input and convolution kernel size respectively, relu represents the activation function, mp represents the max pooling, and concat represents the splicing operation.

(4) Classification Layer

The classification layer classifies the local semantic information M obtained from the Conv Block on the upper layer through a layer of fully connected neural network, and then gets the matching or mismatching results through softmax function. The calculation formula is as follows:

$$y = soft\max(linear(M)) \tag{6}$$

3. Experiment

In this paper, the proposed model is tested in the text

matching dataset LCQMC , and compared with other model methods. ACC and F1-score are used to evaluate the effectiveness of the model.

3.1. Dataset

LCQMC is a large-scale open corpus for text matching. The dataset contains 260068 pieces of data in total, including 238766 training sets, 8802 validation sets, and 12500 test sets. Each data is composed of two sentences and a label. The label is divided into 0 and 1, where 0 represents the semantic mismatch of two sentences, and 1 represents the semantic match of two sentences. Data examples are shown in the following, see Table 1.

Table 1. Example Of LCQMC

Sentence1	Sentence2	Label
这腰带是什么牌子	护腰带是什么牌子	0
货到付款的网站是哪个	什么购物网站是货到付款的	1

3.2. Experimental Environment

The training, validating and testing of all models in this paper are based on the deep learning framework pytorch. The specific experimental environment configuration is shown in the following, see Table 2.

Table 2. Experimental Environment Configuration

Software And Hardware	Configuration
operating system	Ubuntu 20.04.4 LTS
Development tools and languages	Pycharm and Python
Deep learning framework	Pytorch 1.11.0
GPU	NVIDIA GeForce RTX 3090 24G
CPU	Intel® Core™ i9-9900K CPU @ 3.60GHz × 16
Memory	64G

3.3. Experimental Parameters

The BERT model selected in this paper is the Chinese-wwm-ext model pretrained by Cui et al. on a large Chinese corpus. The max length of input data is 128 and the hidden size of BERT is 768, the hidden size of BiLSTM is 334, the convolution kernel of multi-scale convolution is 2 * 768, 3 * 768 and 4 * 768 respectively, the random seed is 47, batch size is 64, learning rate is 2e-5, dropout is 0.2 and the epoch is 3.

3.4. Evaluation criteria

In order to verify the effectiveness of the proposed model, the accuracy Acc and F1-score are used to evaluate the model. The calculation formula of Acc and F1-score is as follows:

$$Acc = \frac{(T_P + T_N)}{(T_P + T_N + F_P + F_N)} \tag{7}$$

$$P = \frac{Tp}{Tp+Fp}, R = \frac{Tp}{Tp+FN}, F1 - score = \frac{2*P*R}{P+R} \tag{8}$$

T_P represents the positive sample predicted by the model as a positive class, T_N represents the negative sample predicted by the model as a negative class, F_P represents the negative sample predicted by the model as a positive class, and F_N represents the positive sample predicted by the model as a negative class.

3.5. Evaluation criteria

In this paper, all experiments are carried out under the same experimental environment as far as possible. First, the baseline model BERT is implemented, and the output of its different layers are used as the semantic features of the text, and experiment on the LCQMC dataset. Then, the BERT model is spliced into BiLSTM and Conv Block respectively to process the features. Finally, our method is tested. The experimental results are shown in following, see Table3.

Table 3. Experiment Result

Model	ACC (%)	F1-score (%)
BERT-cls	86.84	87.82
BERT-pooler	86.83	87.76
BERT-last-avg	86.59	87.65
BERT-first-last-avg	87.87	88.54
BERT-last-four-avg	86.47	87.50
BERT +BiLSTM	86.56	87.63
BERT + Conv Block	88.17	88.66
BERT + BiLSTM + Conv Block	87.87	88.55
Diff(BERT-wwm)	87.80	--
Our method BBMC	88.01	88.72

From the experimental results, it can be seen that compared with the BERT model alone, the ACC and F1 score of the model proposed in this paper are improved. And the ACC is increased by 1.45% and 0.14% respectively compared with the method of splicing the BiLSTM module and the BiLSTM + conv Block module. The F1-score is increased by 1.09% and 0.17% respectively. The ACC is 0.16% lower and the F1-score is 0.06% higher than the method of splicing the Conv Block module. And the ACC has improved compared with the Diff (BERT-wwm) model proposed in paper. This shows that the model proposed in this paper has improved in general.

4. Conclusion

In order to cover the shortage of only using the BERT model cannot well express the text semantic features, this paper uses the hidden layer vectors of different layers output by the BERT model and uses BiLSTM and multi-scale convolution network to further extract rich semantic features. Through training, validating and testing on LCQMC dataset, it is proved that the fusion of BiLSTM and multi-scale convolution can improve the effect of the model, which is sufficient to show that it is feasible to fuse BiLSTM and multi-scale convolution on the basis of pretrained BERT model to finetune.

References

[1] Pang Liang, Lan Yanyan, Xu Jun, Guo Jiafeng, Wan Shengxian, Cheng Xueqi. Overview of Deep Text Matching [J]. Journal of Computer Science, 2017,40 (04): 985-1003.

[2] Yang R, Zhang J, Gao X, Ji F, Chen H. Simple and effective text matching with richer alignment features[J]. arXiv preprint arXiv:1908.00300, 2019.

[3] Huang P S, He X, Gao J, Deng L, Acero A, Heck L. Learning deep structured semantic models for web search using clickthrough data[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013: 2333-2338.

[4] Shen Y, He X, Gao J, Deng L, Mesnil G. A latent semantic model with convolutional-pooling structure for information retrieval[C]//Proceedings of the 23rd ACM international conference on conference on information and knowledge management. 2014: 101-110.

[5] Palangi H, Deng L, Shen Y, Gao J, He X, Chen J, et al. Semantic modelling with long-short-term memory for information retrieval[J]. arXiv preprint arXiv:1412.6629, 2014.

[6] Hu B, Lu Z, Li H, Chen Q. Convolutional neural network architectures for matching natural language sentences[J]. Advances in neural information processing systems, 2014, 27.

[7] Yin W, Schütze H, Xiang B, Zhou B. Abcnn: Attention-based convolutional neural network for modeling sentence pairs[J]. Transactions of the Association for Computational Linguistics, 2016, 4: 259-272.

[8] Pang L, Lan Y, Guo J, Xu J, Wan S, Cheng X. Text matching as image recognition[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2016, 30(1).

[9] Chen Q, Zhu X, Ling Z, Wei S, Jiang H, Inkpen D. Enhanced LSTM for natural language inference[J]. arXiv preprint arXiv:1609.06038, 2016.

[10] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM networks[C]//Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005. IEEE, 4: 2047-2052.

[11] Wang Z, Hamza W, Florian R. Bilateral multi-perspective matching for natural language sentences[J]. arXiv preprint arXiv:1702.03814, 2017.

[12] Mikolov T, Sutskever I, Chen K, Corrado G, Dean J. Distributed representations of words and phrases and their compositionality[J]. Advances in neural information processing systems, 2013, 26.

[13] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

[14] Devlin J, Chang M W, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

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

[16] Li B, Zhou H, He J, et al. On the Sentence Embeddings from Pre-trained Language Models[C]// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020.

[17] Xia T, Wang Y, Tian Y, Chang Y. Using Prior Knowledge to Guide BERT's Attention in Semantic Textual Matching Tasks[J]. 2021.

[18] Meng Jinxu, Shan Hongtao, Wan Junjie, Jia Renxiang. BSLA: Improved Text Similarity Model of Siamese LSTM [J]. Computer Engineering and Application. 2021.

[19] Cui Y, W Che, T Liu, B Qin, Z Yang, S Wang and G Hu, “Pretraining with whole word masking for chinese bert,” arXiv preprint arXiv:1906.08101, 2019.

[20] Cui Y, Che W, Liu T, Qin B, Yang Z. Pre-training with whole word masking for chinese bert[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021, 29: 3504-3514.

[21] Liu X, Chen Q, Deng C, Zeng H, Chen J, Li D, et al. Lcqmc: A large-scale chinese question matching corpus[C]//Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1952-1962.

[22] Zhang Wenhui, Wang Meiling, Hou Zhirong. A short text matching model incorporating contextual semantic differences [J/OL]. Journal of Peking University (Natural Science Edition) https://doi.org/10.13209/j.0479-8023.2022.071.