A Novel Discrimination Structure for Assessing Text Semantic Similarity

Peng Ding[1], Dan Liu[1], Zhiyuan Zhang[2, 3*], Jie Hu[1], Ning Liu[1]

[1] China Telecom Research Institute, China
[2] School of Electronic and Information Engineering, Beijing Jiaotong University, China
[3] Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, China
dingpeng6@chinatelecom.cn, liudan11@chinatelecom.cn, zhangzhiyuan@bjtu.edu.cn, hujie1@chinatelecom.cn, sdqzliuning@163.com

Abstract

Discrimination of semantic textual similarity refers to comparing the similarity between two or more entities (including words, short texts and documents) through certain strategies to obtain a specific quantitative similarity value. Traditional research put more experience into the similarity calculation of the original text content, using the matching degree or distance of characters or words as the yardstick to judge whether the text pairs are similar. However, there are still some problems to be solved in the following aspects: the key points of sentence meaning and word semantics, which play important role in the semantic expression of natural language, are not well integrated into the similarity discrimination, and the interactive features between texts are not fully utilized. To solve the above problems, this paper proposes a novel discrimination structure based on the Siamese Network model and the idea of text matching. In this structure, we introduce sentence meaning key information and word semantic information to realize the extraction of word interaction feature information, and then we realize the text vector representation by using Siamese BiLSTM. The experimental results showed that the accuracy of the proposed model is higher than that of the basic models.

Keywords: Semantic textual similarity, Siamese BiLSTM

1 Introduction

Discriminating semantic similarity is the basis task of natural language processing, such as information recommendation, text generation, text classification. Text similarity calculation refers to comparing the similarity between two or more entities (including words, short text and documents) through certain strategies to obtain a specific quantitative similarity value. In previous studies, people put more experience into the similarity judgment of the original text content, using the matching degree or distance of characters or words as the yardstick to judge whether the text pairs are similar. In this period, the text similarity discrimination mainly use direct text similarity discrimination method [1-3]. Direct text similarity detection method calculates the characters or words in the text in isolation, ignoring the dependency between characters or words and their true or latent meaning in the current text, so it is unable to distinguish the similarity of different texts from the semantic level. For example, for two sentences: “he is a minor” and “he is not over 18 years old this year”, the actual meaning is the same, but the semantic similarity can not be calculated through the superficial text similarity measurement method. Therefore, we need to understand the semantic information of the text more deeply, rather than just stay on the surface meaning of the text.

The emergence of neural network opens up a new way for computer to understand natural language. Bengio proposed the famous word embedding technology [7], which can express the text as multi-dimensional vector in the process of training neural network, so it becomes the link between computer and natural language. Mikolov developed the famous word2vector [8], which was widely used in word embedding training in the following years. With the advent of Glove, the training of word embedding tends to be more mature. Then the neural network based text semantic similarity discrimination method [4-6, 9-10] is widely developed. The word vector obtained by neural network takes into account the position relationship between words, so it contains more semantic information and does not have the problem of dimension disaster. The representative achievement is Long Short Term Memory, LSTM model [11] to solve the problem of long text dependence faced by CNN, it derived a bidirectional long short term memory (BiLSTM) model [12-14] to capture richer context information. Then, the idea of Siamese network [15-16] is proposed by researchers. Different from the single input idea of traditional network, Siamese network model uses two networks with shared weights to learn the text representation of text pairs, and then measures their spatial similarity by Manhattan distance, Euclidean distance or cosine similarity. With the attention mechanism proposed in reference [17] and successfully applied to machine translation tasks, attention mechanism has gradually attracted the attention of researchers, and has become an important role in natural language processing model [18-20].

In the process of analyzing the existing text semantic similarity discrimination methods based on neural network, we found two shortcomings: one is that in the model input stage, the traditional neural network model usually treats every word in the text without any difference, which is not consistent with the thinking mode of human understanding the text. Meanwhile, directly using the pre-training word vector to express words is convenient for model learning, but it ignores the semantic information behind the words, which is

*Corresponding Author: Zhiyuan Zhang; E-mail: zhangzhiyuan@bjtu.edu.cn
DOI: 10.53106/160792642022072304006

especially unfavorable for the computer to understand the ideographic characters (such as Chinese). Secondly, in the learning process of the model, the vector representation of text is usually considered in isolation, and little attention is paid to the interactive information between texts. It is very disadvantageous for learning models dealing with paired samples, such as text semantic similarity discrimination model.

Even though the research on the similarity discrimination of English texts has been relatively mature, if it is directly transplanted into Chinese texts, the effect of similarity discrimination is still not ideal. The reason is that the similarity discrimination model designed for English text does not take into account the characteristics and difficulties of Chinese text, so its performance in Chinese text is not satisfactory. The characteristics and difficulties of Chinese text are summarized as: (1) Chinese has the characteristics of large character set. There are 48000 Chinese characters, but only 26 letters in English. (2) Chinese lacks morphological changes [21], the expressions of past tense and future tense in English are completed through the morphological change of verbs, but there is no similar morphological change in Chinese. (3) The granularity of Chinese word segmentation has different effects on semantic segmentation [22]. At present, there is no recognized standard on how to deal with Chinese granularity on semantic segmentation. (4) The structure of Chinese text usually does not have a fixed form, but Chinese sentences depend on the order of words, which implicitly conveys semantic information. Therefore, a more adaptive learning model is needed to extract more semantic features of Chinese.

Based on the above problems, this paper, on the basis of previous studies, research on the semantic similarity discrimination method for sentence level of Chinese texts. We propose A novel semantic matching structure for discriminating semantic similarity with mining semantic expression characteristics. The structure firstly introduce sentence meaning key information and word semantic information (which are collectively referred to as natural language semantic expression feature information) and realizes the extraction of word interaction feature information. Secondly, the indirect interactive matching method of Siamese BiLSTM introduces differential attention after the hidden layer output, which realize the interaction between context feature vectors. The multiple matching methods used to help the model fully learn the interactive feature information between texts.

2 Methodology

2.1 Motivation

At present, the text similarity discrimination method usually starts from two ideas. One is based on Siamese network model [15, 23], which uses two deep neural networks with shared weights to express the text to be discriminated as the text vector in the same vector space, and then to discriminate the similarity. The other one is based on the text matching [24-26], which usually matches the text in small units (such as words or context feature vectors), and then aggregates the matching results in a certain way to obtain the similarity discrimination results. Siamese network models pay more attention to the internal relationship between text pairs through sharing weights to reduce the dimension of the sample to a lower dimension. For the matching models, they will pay more attention to the local matching of sentences, and usually they not only use sentence information, but also uses more fine-grained information such as words and phrases.

However, both of the two ideas have shortcomings. Compared with matching models, Siamese network models can not learn enough interaction characters and semantic information while encoding. However, Siamese network model's parameters are shared, which means that the model needs fewer parameters during training, so it is not easy to have the phenomenon of over fitting. The disadvantage of matching models is that they ignore the focus of sentence meaning and semantic information of the word itself. Therefore, motivated by the Siamese network model and the idea of text matching, an innovative structure of text similarity analysis is proposed.

2.2 Discriminating Semantic Similarity with Mining Semantic Expression Characteristics

2.2.1 Extraction of Sentence Key Information and Word Semantic Features

In order to make better use of the word semantic expression characteristics of natural language (The efficient expression of words is the basis for the further semantic similarity discrimination), we use the sentence meaning word recognition and word semantic interaction to match the text semantic. By using sentence meaning word recognition and word meaning matching technology, sentence meaning key information and word semantic information are introduced into the model respectively.

We define the meaning words of a sentence as verbs, nouns, pronouns, adjectives and adverbs in the sentence. The words of the above parts of speech usually contain more abundant semantic information, which will closely related to the sentence meaning. Then we use the part of speech tagging method to recognize the sentence meaning word sequence. Based on the research on the weight of words with different parts of speech in the sentences in reference [27], we set the contribution weights of sentence meaning words and other words to recognize sentence meaning. After normalization, the weight sequence of semantic words and other words in the original sentence can be obtained. The weight shows the importance of each category of words in the original sentence for the meaning of the sentence. We call this part as the sentence key word recognition based on part of speech tagging. Then, we will use the famous tool in reference [28] for measuring the word semantic matching of words by calculating the similarity between words. To a certain extent, word similarity reflects the similarity between the texts. Because the more similar the words in the two texts, the more likely they are to be similar. If the words in the two texts are more different, the more likely they are not to be similar. In addition, the calculation of word similarity involves two texts, then the calculation of word similarity itself is a good way of measuring text interaction.

With the help of word similarity calculation method proposed in reference [29], this paper constructs a word similarity matrix and introduces early text interaction information for similarity discrimination model from the perspective of word semantic interaction. The calculation formula of word similarity is shown as:

$$Sim(c_1,c_2) = \max_{i=1...n,j=1...m} sim(P_{1i},P_{2j}) \quad (1)$$

where c_1 and c_1 are the words from different sentences， P_{1i} is n meanings (also called senses) contained in sentence 1 and P_{2j} is *m* semantic meanings contained in sentence 2 [12]. The similarity of words depends on the similarity of the meanings they contain. The calculation of meaning similarity finally comes down to the follow:

$$Sim(S_1,S_2) = \frac{a}{d+a} \quad (2)$$

where S_1 and S_2 are two sememes [28], sememe is the semantic unit of different meanings. d is the length of the path to in the sememe hierarchy [30-31]. a an adjustable parameter. After word segmentation, we will get the word sequence of the text pair. Then the word similarity is calculated by formula (3-6), and finally we get the word semantic similarity matrix *SIM* of the two texts. We call this part as word semantic features extraction based on similarity. On the basis of sentence key word recognition and word semantic matching, combined with pre-training word vector, the final word vector can be calculated as the follow structure in Figure 1.

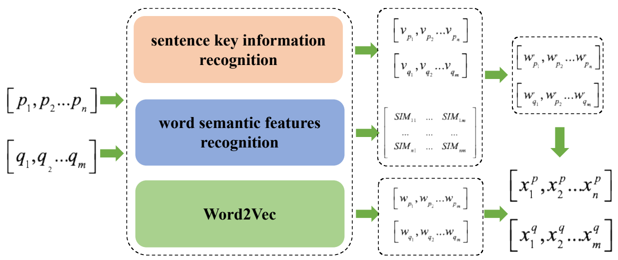


Figure 1. Structure for extracting sentence key information and word semantic features (the calculation of word embedding)

In the figure, $[p_1,p_2,...,p_n]$ and $[q_1,q_2,...,q_m]$ are the inputs, they are the word sequences after two sentences segmentation, $[w_{p_1},w_{p_2},...,w_{p_n}]$, $[w_{q_1},w_{q_2},...,w_{q_m}]$ are the corresponding pre training word sequences after Word2Vec embedding, $[v_{p_1},v_{p_2},...,v_{p_n}]$, $[v_{q_1},v_{q_2},...,v_{q_m}]$ are the word category weight sequences of the two sentences respectively, SIM stands for the word semantic similarity matrix, which can be computed by formula (1) and (2). Then, the text matching word vectors $[w'_{p_1},w'_{p_2},...,w'_{p_n}]$ and $[w'_{q_1},w'_{q_2},...,w'_{q_m}]$ are computed as:

$$w'_{p_i} = \sum_{i=1}^{m}(v_{qj}e^{(SIM_{ij})} / \sum_{k=1}^{m} e^{(SIM_{ik})}) \odot w_{qj} \quad (3)$$

$$w'_{q_j} = \sum_{i=1}^{n}(v_{pi}e^{(SIM_{ij})} / \sum_{k=1}^{n} e^{(SIM_{kj})}) \odot w_{qi} \quad (4)$$

After getting word vector sequences, we let them participate in the representation of the final word embedding, then we use text matching word vectors and pre-training word vectors to calculate the final word vectors $[x_1^p,x_2^p...,x_n^p]$ and $[x_1^q,x_2^q...,x_m^q]$:

$$x_i^p = contat(w_{p_i},w'_{p_i}) \quad (5)$$

$$x_i^q = contat(w_{q_i},w'_{q_i}) \quad (6)$$

2.2.2 Extraction of Text Semantic Features

Recent research [32-34] have proved that the Siamese Network pays more attention to the relationship between the two texts when dealing with the NLP tasks. Taking the task of similarity discrimination as an example, the Siamese Network model is more likely to learn a similarity measurement method. This way is more consistent with the human way of thinking to judge whether the text 1 is similar to text 2 or not. Because when people judge whether two texts are similar or not, they do not first remember the characteristics of a single text and then determine its similarity, but usually by comparing and observing the similarities and differences between the two texts. The information contained in the text pair is more likely to enter people's brain in the way of double inputs. In addition to the above advantages, the idea of sharing weights in Siamese Network model also reduces the training burden of the model.

Therefore, in order to capture the context information as much as possible, we choose to use Siamese BiLSTM Network as the basic network model of text semantic similarity discrimination model to handle the text semantic features of generating text vector. BiLSTM Network model's hidden layer include a forward LSTM and a reverse LSTM, which are used to capture the context feature information of the text. As a result of this two-way learning mechanism, compared with ordinary LSTM, the context feature information captured by BiLSTM will be more abundant.

Theoretically, the deep neural network model can transfer the dependent information step by step through the hidden layer when handling NLP tasks. Then the context feature vector output from the last hidden layer should fully learn the context feature information of the text, so it can be used as the vector representation of the text. This method ignores the information contained in other hidden layers and lacks the interaction between text pairs.

Therefore, the final text vector representation loses part of the semantic information of the original text. We design a differential attention mechanism in the hidden layer of Siamese BiLSTM for the outputs to realize the interactive matching and the extraction of text semantic features of different text context feature vectors. The model of proposed Siamese BiLSTM is shown in Figure 2.

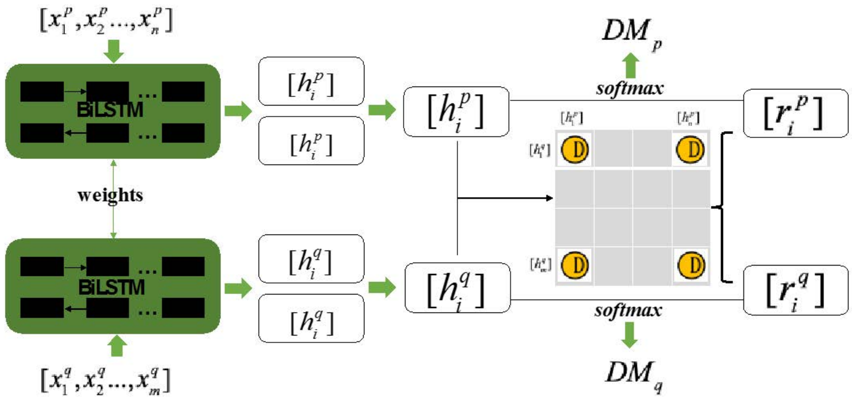


Figure 2. Structure for text vector generation using BiLSTM and differential attention

$[x_1^p, x_2^p ..., x_n^p]$ and $[x_1^q, x_2^q ..., x_m^q]$ are the word vectors corresponding to the input word sequences $[p_1, p_2, ..., p_n]$ and $[q_1, q_2, ..., q_m]$, respectively, as shown in figure 1(the final output). For the four hidden layer vectors $[\overrightarrow{h_i^p}]$, $[\overleftarrow{h_i^p}]$ and $[\overrightarrow{h_i^q}]$, $[\overleftarrow{h_i^q}]$, which are the outputs by Siamese BiLSTM network, we make the following calculations. Firstly, the hidden layer vectors of different directions are stitched together for each sentence in the same time step by using the following formulas:

$$h_i^p = contat(\overrightarrow{h_i^p}, \overleftarrow{h_i^p}) \tag{7}$$

$$h_i^q = contat(\overrightarrow{h_i^q}, \overleftarrow{h_i^q}) \tag{8}$$

Then we have the hidden layer vector sequences. The hidden layer vector of each time step in the sequence contains the context feature information of the current sentence.

The function shown in equation (9) is used as the scoring function to measure the distance between the hidden layer vectors of two sentences. The distance can be understood as the difference between hidden layer vectors from different sentences.

$$FD_{ij} = \underset{i=1...n, j=1...m}{absum} (h_i^p - h_i^q) \tag{9}$$

where **absum** (•) is a custom function, it stands for the operation of finding the sum of absolute values along a certain dimension of a vector. After getting the difference matrix $FD_{n' m}$, the attention value of each hidden layer vector of the two sentences is calculated by the following formula:

$$r_i^p = sum(FD[i]) / m \tag{10}$$

$$r_j^p = sum(FD[:, j]) / n \tag{11}$$

where $FD[i]$ and $FD[:, j]$ are the i-th row and j-th column of the difference matrix $FD_{n' m}$, respectively. They are the text semantic attention value, then we have the text semantic attention vector $[r_i^p]$ and $[r_i^q]$. After normalization by softmax, the final sentence vector can be obtained by weighted sum of the hidden layer vectors of the original sentence, which will be calculated by:

$$DM_p = \sum_{i=1}^{n} (e^{r_i^p} / \sum_{k=1}^{n} e^{r_k^p}) \circ h_i^p \tag{12}$$

$$DM_q = \sum_{j=1}^{m} (e^{r_j^q} / \sum_{k=1}^{m} e^{r_k^q}) \circ h_j^q \tag{13}$$

According to the idea of interactive matching: the global similarity depends on the local similarity, then the global difference also depends on the local difference.

2.3 Model Structure

According to the method proposed in the previous two sections, we design a multiple interactive matching model-we call it MIMM-for discriminating semantic similarity with mining semantic expression characteristics. The structure is listed below.

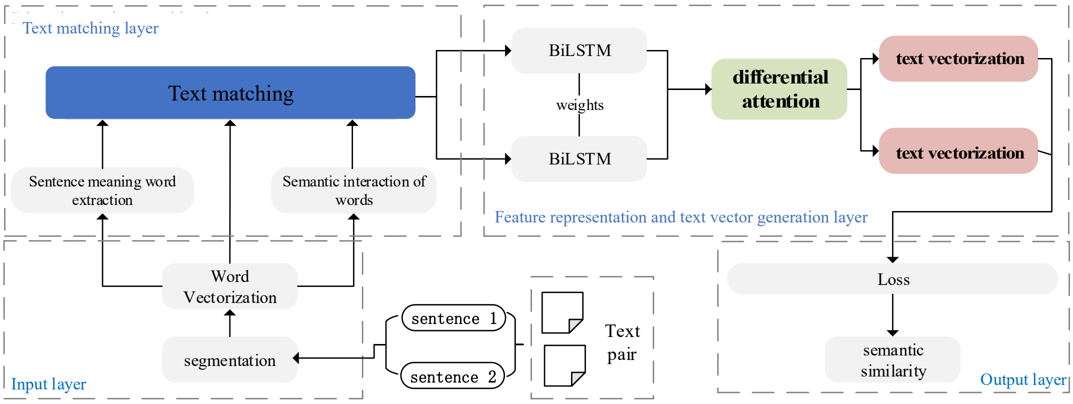


Figure 3. The model structure

As shown in Figure 3, The model is mainly composed of input layer, text matching layer, feature representation, text vector generation layer and output layer. The input of the model is the text pair in the data set, and the output is the result of text similarity. The structure and workflow of each layer are described in detail as follows.

- **Input layer**

The task of input layer is to transform characters into the form that can be processed by computer, namely vectorization. There are three steps: (1) Useless content elimination: Remove spaces, punctuation marks, or other content unrelated to the meaning of the sentence. (2) Segmentation: In this paper, the segmentation model provided by LTP platform is used to process the segmentation of data sets. (3) Vectorization: In this paper, the pre-trained word2vec model is used to initialize the word vectors.

- **Text matching layer**

The text matching layer is designed according to the matching method described in section 2, which is used to capture the sentence meaning key information, word semantic information and word level interaction feature information of sample pairs.

After the word segmentation at input layer, we will collect two word sequences of sequence P and Q. Using the part of speech tagging model, sentence key words and other words of the two sentences are identified respectively. Then according to the category (sentence key words and other words) weight, the weight sequence of two sentences can be determined using the method proposed in section 2.

The text matching vector sequences of sentence P and sentence Q are calculated by formula (3) and formula (4), separately. Finally, by combining the text matching sequences with the original pre-training word sequences according to formula (5) and (6), the final word vector sequence of sentence A and B can be obtained as the output sequence of Figure 1 as $[x_1^p, x_2^p \ldots, x_n^p]$ and $[x_1^q, x_2^q \ldots, x_m^q]$. $[x_1^p, x_2^p \ldots, x_n^p]$ and $[x_1^q, x_2^q \ldots, x_m^q]$ are the vectorized text representations of sentences A and B (the outputs of this layer), and they are the inputs of context feature representation layer.

- **Feature Representation and Text Vector Generation Layer**

Then the vectorized text representations are input into the Siamese BiLSTM network to get the hidden layer vector sequence of sentence pairs from four directions, we define them as $[\overrightarrow{h_i^p}]$, $[\overleftarrow{h_i^p}]$ and $[\overrightarrow{h_i^q}]$, $[\overleftarrow{h_i^q}]$, then the final context feature vector sequences $[h_i^p]$ and $[h_i^q]$ are obtained by splicing. Then the mechanism of matching attention is activated.

First, for the context feature vector sequences $[h_i^p]$ and $[h_i^q]$, the difference values between the context eigenvectors from different sequences are calculated according to formula (9), we get the difference matrix $FD_{n' m}$. The difference attention value was calculated according to formula (10) and (11), then we have the difference attention vector by equations (10) and (11). Finally, the sentence vectors representation DM_p and DM_q are calculated according to equations (12) and (13).

- **Output layer**

For the output layer, the comparative loss function is used to calculate the loss. It was proposed in [35], which was used in dimensionality reduction, it can make the original similar samples still similar in the extracted feature space, while the originally dissimilar samples are still not similar in the extracted feature space. The formula is shown as:

$$L = \frac{1}{2N}\sum_{n=1}^{N} yd^2 + (1-y)\max(margin - d, 0)^2 \quad \textbf{(14)}$$

where N is the number of samples, y is the label of whether the two samples are similar, $y = 1$ means similar between samples, $y = 0$ means dissimilar between samples. $margin$ is the threshold, d is the distance value between text vectors, which can be obtained by Euclidean distance.

The loss function can ensure that the loss value of dissimilar samples decreases with the increase of sample distance, while the loss value of similar samples increases with the increase of sample distance. It is more suitable for the training of natural language processing model based on Siamese network.

We use Euclidean distance to measure the similarity between sentence vector and sentence vector (expressed by D). When $D \le 0.5$, the two sentences are similar and the predictive tag value is 1; when $d > 0.5$, the two texts are not similar and the predictive tag value is 0.

3 Experiments and Results Analysis

3.1 Dataset

The model proposed in this paper is mainly used to judge the semantic similarity of Chinese texts at sentence level. In order to evaluate the performance of the model, the AFQMC dataset is selected for experimental test. The dataset is a binary dataset (similar, not similar). This dataset contains 34334 samples. Each sample is composed of two sentences and a tag. The tags (0, 1) indicate the similarity or dissimilarity between sentences. 1 means the two sentences of a sample are similar, and 0 is opposite. The data sample after preprocessing is shown in Table 1.

Table 1. Samples of dataset

Sentence 1	Sentence 2	Tag
Which company can use Hua Bei	What are the companies that use Hua Bei	1
I paid the electricity bill with Hua Bei	My electricity bill paid from Hua Bei is staggered	0

3.2 Evaluation Indicator

In this paper, accuracy and F1 score are selected as the evaluation indicators for evaluating the model performance

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{15}$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{16}$$

where TP is True Positive, TN is True Negative, FP is False Positive, and FN is False Negative.

3.3 Evaluation of Model Validity

- **Comparison**

We use the following methods (please see Table 2) mentioned in this paper as the comparisons to evaluate the validity of the proposed model.

Table 2. Comparison methods

Model	Abbreviation	Section
Siamese BiLSTM network	Sia-BiLSTM	2.2.2
Extraction of sentence key information and word semantic features	M1	2.2.1
Extraction of text semantic features	M2	2.2.2
our model	M3	2.3

- **Results**

According to the different purposes of model setting in the control group, the above four models were set as three control groups. The results are shown in Table 3.

During the experiment, the model parameters in the same group are always consistent to prevent other factors from affecting the experimental results. As shown in Table 3, group A, compared with the basic network model Sia-BiLSTM, the accuracy and F1 score of M1 model are significantly improved (Accuracy improved from 0.6954 to 0.7383, F1 improved from 0.6987 to 0.7396) after incorporating text matching method based on sentence meaning word recognition and word semantic interaction. This is because the text matching method makes the model pay more attention to the key information of sentence meaning, the semantic information of words and the interactive feature information between them. However, the basic model only uses the pre-trained word vectors as the input, without focusing on each word in the sentence, and lacks the interaction of sentence pairs, so the learning ability is limited.

Similarly, we can see from Table 3, group B, that when the text matching method is added to the basic model Sia-BiLSTM structure, the accuracy and F1 score of M2 on the dataset are improved (Accuracy improved from 0.6954 to 0.7109, F1 improved from 0.6987 to 0.7114). This is because M2 takes the difference between sentence and context feature vectors as attention value, and considers the output of each hidden layer of the model. The attention vector is used to retain the information of the text itself.

As can be seen from Table 3, group C, the performance of the above two models (M1, M2) on the dataset is not as good as the proposed model. It can be seen that the two text matching methods can complement each other, which make the accuracy and F1 score of M3 on the dataset are improved. Therefore, it can be proved that M3 is a reasonable and effective model for text semantic similarity discrimination. It can be found that the improvement of our model's accuracy is not directly equal to the sum of M1 and M2. This is because the two text matching models are overlapped in extracting related text feature information.

Table 3. Results of comparison groups

Group	Models	Accuracy	F1
A	Sia-BiLSTM	0.6954	0.6987
	M1	**0.7383**	**0.7396**
B	Sia-BiLSTM	0.6954	0.6987
	M2	**0.7109**	**0.7114**
C	M1	0.7383	0.7396
	M2	0.7109	0.7114
	M3	**0.7435**	**0.7497**

3.4 Evaluation of Model Accuracy

- **Comparison**

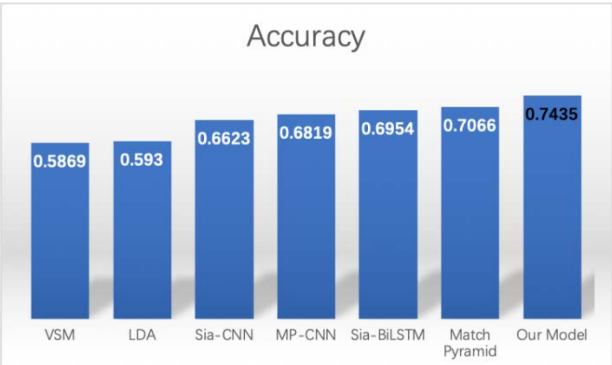
In order to further verify that the model proposed in this paper is a better model for text semantic similarity discrimination, this paper selects some common similarity discrimination models and conducts comparative experiments. The parameters of the contrast groups were consistent with the above experimental settings.

According to the main characteristics of the model, this paper use the following seven models to evaluate the model accuracy, the models can be classified as four comparison groups.

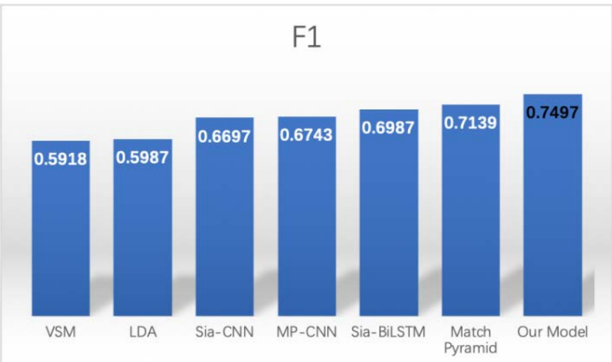
- ✓ LDA and VSM are traditional text semantic similarity discrimination methods, which do not belong to neural network model.
- ✓ Sia-CNN and Sia-BiLSTM are text semantic similarity discrimination models built under the framework of Siamese Networks.
- ✓ MP-CNN and Match Pyramid are text semantic similarity discrimination models based on the idea of interactive matching by extracting word and text semantic features.
- ✓ Our model is based on the Siamese Network framework and introduces the idea of text matching. It is a comprehensive neural network model combining Siamese Network model and text matching by extracting sentence key information, word nad text semantic features.

- **Results**

The performance of the models in contrast group on the AFQMC dataset is shown in Figure 4.



(a) The result of accuracy



(b) The result of F1 score

Figure 4. Experimental comparison

The accuracy of the proposed model on AFQMC dataset is 0.7435, and the F1 score is 0.7497. In terms of accuracy, our model is 8.12% and 4.81% higher than Sia-CNN and Sia-BiLSTM, respectively, and 6.16% and 3.69% higher than MP-CNN and Match Pyramid, respectively. These two sets of comparative results show that the combination of Siamese network structure and text matching idea is reasonable and effective. Because Siamese network models have the characteristics of dual input and their loss function designed to measure the output difference, these characteristics make it very suitable for the task of text semantic similarity discrimination. The introduction of text matching idea can inject interactive feature information between texts into Siamese Network model, which makes up for the defect that Siamese Network model ignores interactive features between texts. The combination structure can make the model gain in performance.

4 Conclusion

The neural network model for the task of text semantic similarity discrimination usually ignores the characteristics of natural language semantic expression and does not extract the interactive features of text. This paper proposes a novel discrimination structure for assessing text semantic similarity, which combines the characteristics of natural language semantic expression. It is an effective neural network based structure for text semantic similarity discrimination. The control group experiments on AFQMC dataset contains three control models, which proves the effectiveness of the proposed combination structure. The contrast group experiments includes six existing models. By comparing their performance on the same data set, it is proved that the proposed model structure is a more accurate model for text semantic similarity discrimination.

Although this paper proposes a discrimination model structure for handle the text similarity, it only stays at the basic work level and proposes an improved model analysis framework. Many fine-grained research work still needs to be addressed, such as the work of more granular text processing, the work of new word recognition. But maybe the work of this paper can give some inspiration to the future research, and can help other researchers in the future work based on the model structure of this paper (or the improved structure based on this paper), integrate more efficient analysis model, and improve the efficiency of text similarity discrimination.

Acknowledgement

This work has been supported by the Fundamental Research Funds for the Central Universities (grant number: 2022JBMC014) and Beijing Nova Program (Z211100002121120) from Beijing Municipal Science & Technology Commission.

References

[1] F. Tian, P. Gao, L. Li, W. Zhang, H. Liang, Y. Qian, R. Zhao, Recognizing and regulating e-learners' emotions based on interactive Chinese texts in e-learning systems, *Knowledge-Based Systems*, Vol. 55, pp. 148-164, January, 2014.

[2] A. Selvarasa, N. Thirunavukkarasu, N. Rajendran, C. Yogalingam, S. Ranathunga, G. Dias, Short Tamil Sentence Similarity Calculation using Knowledge-Based and Corpus-Based Similarity Measures, *3rd International Moratuwa Engineering Research Conference*, Moratuwa, Sri Lanka, 2017, pp. 443-448.

[3] Z. Liu, W. Zhang, J. Sun, H. N. H. Cheng, X. Peng, S. Liu, Emotion and Associated Topic Detection for Course Comments in a MOOC Platform, *5th International Conference on Educational Innovation through Technology*, Tainan, Taiwan, 2016, pp. 15-19.

[4] B. Sathiya, T. V. Geetha, A review on semantic similarity measures for ontology, *Journal of Intelligent and Fuzzy Systems,* Vol. 36, No. 4, pp. 3045-3059, April, 2019.

[5] J. Singh, A. K. Singh, Semrank: A Semantic Similarity-Based Tweets Ranking Approach, *International Journal of Cognitive Informatics and Natural Intelligence*, Vol. 15, No. 3, pp. 74-96, July-September, 2021.

[6] Y. Jiang, A unified framework for semantic similarity computation of concepts, *Multimedia Tools and Applications*, Vol. 80, No. 21-23, pp. 32335-32378, September, 2021.

[7] Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin, A Neural Probabilistic Language Model, *Journal of Machine Learning Research*, Vol. 3, pp. 1137-1155, March, 2003.

[8] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, *1st International Conference on Learning Representations*, Scottsdale, Arizona, USA, 2013, pp. 1-12.

[9] J. Pennington, R. Socher, C. Manning, Glove: Global Vectors for Word Representation, *Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, USA, 2014, pp. 1532-1543.

[10] Y. Zhang, Z. Zhang, D. Miao, J. Wang, Three-way Enhanced Convolutional Neural Networks for Sentence-level Sentiment Classification, *Information Sciences*, Vol. 477, pp. 55-64, March, 2019.

[11] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, R. Ward, Deep Sentence Embedding Using Long Short-Term Memory Networks: Analysis and Application to Information Retrieval, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 24, No. 4, pp. 694-707, April, 2016.

[12] Y.-A. Chung, J. Glass, Speech2Vec: A Sequence-to-Sequence Framework for Learning Word Embeddings from Speech, *19th Annual Conference of the International-Speech-Communication-Association*, Hyderabad, India, 2018, pp. 811-815.

[13] C. Liu, L. Zhu, A two-stage approach for predicting the remaining useful life of tools using bidirectional long short-term memory, *Measurement*, Vol. 164, Article No. 108029, November, 2020.

[14] S. Kamath, C. Shivanagoudar, K. G. Karibasappa, Part of Speech Tagging Using Bi-LSTM-CRF and Performance Evaluation Based on Tagging Accuracy, *4th International Conference on Computing and Network Communications*, Chennai, India, 2020, pp. 299-310.

[15] Y. Zhou, J. Li, B. Du, J. Chang, Z. Ding, T. Qin, Learning adaptive updating siamese network for visual tracking, *Multimedia Tools and Applications*, Vol. 80, No. 19, pp. 29849-29873, August, 2021.

[16] G. A. Rao, P. V. V. Kishore, Sign Language Recognition System Simulated for Video Captured with Smart Phone Front Camera, *International Journal of Electrical and Computer Engineering*, Vol. 6, No. 5, pp. 2176-2187, October, 2016.

[17] T. Wu, S. Tang, R. Zhang, J. Cao, Y. Zhang, CGNet: A Light-Weight Context Guided Network for Semantic Segmentation, *IEEE Transactions on Image Processing*, Vol. 30, pp. 1169-1179, December 2020.

[18] Y. Liu, L. Ji, R. Huang, T. Ming, C. Gao, J. Zhang, An attention-gated convolutional neural network for sentence classification, *Intelligent Data Analysis*, Vol. 23, No. 5, pp. 1091-1107, October, 2019.

[19] M. Shi, Y. Tang, J. Liu, TA-BLSTM: Tag Attention-based Bidirectional Long Short-Term Memory for Service Recommendation in Mashup Creation, *International Joint Conference on Neural Networks*, Budapest, Hungary, 2019, pp. 1-8.

[20] N. T. Tran, V. Luong, N. L.-T. Nguyen, M. Nghiem, Effective attention-based neural architectures for sentence compression with bidirectional long short-term memory, *the Seventh Symposium on Information and Communication Technology*, Ho Chi Minh City, Vietnam, 2016, pp. 123-130.

[21] Y. Chen, S. Nan, Q. Tian, H. Cai, H. Duan, X. Lu, Automatic RadLex coding of Chinese structured radiology reports based on text similarity ensemble, *BMC Medical Informatics and Decision Making*, Vol. 21, No. Suppl 9, Article No. 247, November, 2021.

[22] X. Wang, S. Ju, S. Wu, A Survey of Chinese Text Similarity Computation, *4th Asia Information Retrieval Symposium*, Harbin, China, 2008, pp. 592-597.

[23] Y. Benajiba, J. Sun, Y. Zhang, L. Jiang, Z. Weng, O. Biran, Siamese Networks for Semantic Pattern Similarity, *IEEE 13th International Conference on Semantic Computing*, Newport Beach, CA, USA, 2019, pp. 191-194.

[24] H. Liu, N. Lin, Z. Chen, K. Li, S. Jiang, Learning Deep Matching-Aware Network for Text Recommendation using Clickthrough Data, *International Conference on Asian Language Processing*, Shanghai, China, 2019, pp. 96-101.

[25] C. Xu, Z. Lin, H. Wang, S. Wu, Multi-Level Compare-Aggregate Model for Text Matching, 2019 *International Joint Conference on Neural Networks*, Budapest, Hungary, 2019, pp. 1-7.

[26] Z. Wang, W. Hamza, R. Florian, Bilateral multi-perspective matching for natural language sentences, *the 26th International Joint Conference on Artificial Intelligence*, Melbourne, Australia, 2017, pp. 4144-4150.

[27] H. Wu, A. Wumaier, K. Abiderexiti, L. L. Wang, T. Yibulayin, Method of computing Chinese sentence similarity based on part-of-speech feature, *Computer Engineering and Design*, Vol. 41, No. 1, pp. 159-164, 2020.

[28] Y. Wang, X. Feng, Y. Zhang, H. Chen, L. Xing, An improved semantic similarity algorithm based on HowNet and CiLin, *MATEC Web of Conferences*, Vol. 309, Article No. 03004, March, 2020.

[29] Q. Zhou, G. Yue, Y. Meng, Incorporating HowNet-Based Semantic Relatedness Into Chinese Word Sense Disambiguation, *20th Chinese Lexical Semantics Workshop*, Beijing, China, 2019, pp. 359-370.

[30] Y. Niu, R. Xie, Z. Liu, M. Sun, Improved Word Representation Learning with Sememes, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 2017, pp. 2049-2058.

[31] J. Hu, Y. Ouyang, C. Li, C. Wang, W. Rong, Z. Xiong, Hierarchical Lexicon Embedding Architecture for Chinese Named Entity Recognition, *30th International Conference on Artificial Neural Networks*, Bratislava, Slovakia, 2021, pp. 345-356.

[32] M. Liu, Y. Lu, S. Long, J. Bai, W. Lian, An Attention-based CNN-BiLSTM Hybrid Neural Network Enhanced with Features of Discrete Wavelet Transformation for Fetal Acidosis Classification, *Expert Systems with Applications*, Vol. 186, Article No. 115714, December, 2021.

[33] P. Khurana, P. Agarwal, G. Shroff, L. Vig, A. Srinivasan, Hybrid BiLSTM-Siamese network for FAQ Assistance, *the 2017 Acm Conference on Information and Knowledge Management*, Singapore, 2017, pp. 537-545.

[34] M. Nicosia, A. Moschitti, Accurate Sentence Matching with Hybrid Siamese Networks, *the 2017 Acm Conference on Information and Knowledge Management*, Singapore, 2017, pp. 2235-2238.

[35] R. Hadsell, S. Chopra, Y. Lecun, Dimensionality Reduction by Learning an Invariant Mapping, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, 2006, pp. 1735-1742.

Biographies

Peng Ding is the team leader of the Blockchain and Metaverse Research Center, Big Data And Artificial Intelligence Institute, China Telecom Research. He received his Ph.D degree in Instrument Science and Technology from BeiHang University (BUAA), China, in 2007. Since 2013, he has worked at China Telecom Research. His research interests include artificial intelligence, edge computing and digital twin.

Dan Liu received her M.D. degree in communication and information systems from Beijing Jiaotong University (BJTU), China, in 2019. She is currently working as an engineer at China Telecom Research. Her research interests include edge computing, edge intelligence, digital twin.

Zhiyuan Zhang received his PHD degree in Beijing Jiaotong University. Currently, he is associate professor in the School of Electronics and Information Engineering at Beijing Jiaotong University. His current research fields include recommender system, machine learning, data mining and artificial intelligence.

Jie Hu received the M.S. degree in Software Engineering from Beijing University of Posts and Telecommunications, Beijing, China in 2017. Her research interests include Information Retrieval, Multi-hop Reasoning and Natural Language Processing.

Ning Liu received his PHD degree in Beijing Jiaotong University. His research interest includes sentiment analysis, deep learning and NLP techniques.