Chinese Sentence Semantic Matching With Multi-Granularity Based on Siamese Neural Network

Xuan Wen
School of Computer Science
Hubei University of Technology
Hubei, China
1595159972@qq.com

Jianxia Chen
School of Computer Science
Hubei University of Technology
Hubei, China
chenjianxiawh@gmail.com

Yu Chen
School of Computer Science
Hubei University of Technology
Hubei, China
1148848330@qq.com

Shirui Sheng
School of Computer Science
Hubei University of Technology
Hubei, China
1606472332@qq.com

***Abstract*—Sentence semantic matching is one of critical research in various NLP tasks such as natural language inference, paraphrase identification, and question answering, in which similarity of input sentences has always been a key aspect to determine the semantic relations of sentences. One of the most popular models is to utilize single word granularity to address the semantic similarity. However, it is not appropriate for Chinese sentences semantic matching. This is because there are various meanings following various granularities such as characters or word segmentation in a Chinese sentence. In addition, it is difficult for the sentence semantic matching due to its own short contents and sparse features. Inspired by Siamese Neural Network, an artificial neural network that uses the same weights while working in tandem on two different input vectors to compute comparable output vectors, this paper proposes a Multi-Granularity Fusion neural network, which enables preserving semantic features from both the character-granularity and the word-granularity in Chinese sentences. The paper evaluates the proposed architecture on highly competitive benchmark datasets related to sentence matching. Experimental results show that the proposed architecture, which retains both characters and words features of sentences, and achieves state-of-the-art performances for most of the tasks.**

Index Terms: Siamese Network; Multi -Granularity; Chinese Sentence; Semantic Matching

I. INTRODUCTION

Semantic matching plays a critical role in many NLP (natural language processing) tasks, such as question and answer (QA) [1], machine translation (MT) [2], information retrieval [3], etc. As a widely existing text representation over the Internet, sentence semantic matching has been gradually showing its strong research values. How to effectively excavate and analyze the sentence semantic has become a research hotspot in the field of NLP.

Usually, the most serious issues of sentences semantic matching are resulted from their short contents and sparse features. To solve this problem, traditional methods often mine information according to the original text, such as using semantic dictionary HowNet [4] or introducing topic model LDA [5] to assist sentences semantic matching. However, these methods often carry out semantic matching only by weighting the method of the original sentences. Obviously, the rich semantic information of the sentences in Chinese texts has not been fully utilized to improve the matching performance. This is because Chinese texts have its own particularity, the semantic matching of Chinese sentences is often affected by their word segmentations.

The common semantic matching methods usually calculated similarities from the aspect of the word granularity, however, ignored the extraction of semantic features of the sentence. With the rapid development of deep learning, many deep learning models based on the word granularity have been proposed in the research of sentence semantic matching, such as DeepMatch tree [6], Match-Pyramid [7], ARC-I [8], etc. However, these models do not obtain the rich characteristics of the sentence itself via only the single word granularity. It may even become worse only depending on a single word granularity calculation. Therefore, some researchers design a novel neural network model that can combine words with characters together to form a new sequence such as Lattice CNNs [9]. Although this network has achieved better results in QA (question and answer), such a simple combination may introduce the noise and even lose its original meaning.

Inspired by SNN (siamese neural network) [10], an artificial neural network that uses the same weights while working in tandem on two different input vectors to compute comparable output vectors, this paper proposes a novel architecture, named MGFSN which is an abbreviation for Multi-Granularity Fusion Siamese neural Network. The MGFSN enables preserving semantic features from both the character-granularity and the word-granularity in Chinese sentences. In particular, the proposed MGFSN architecture is composed of three components including the word embedding layer, the multi-granularity coding layer, and the semantic interaction layer. The paper evaluates the proposed architecture on a highly competitive benchmark LCQMC dataset related to sentence matching. Experimental results show that the proposed architecture achieves state-of-the-art performances for most of the tasks.

The rest of this paper is structured as follows. The paper introduces the related work about semantic matching of Chinese sentences in Section II, and describes the proposed

DOI reference number : 10.18293/SEKE2021-074

Multi-Granularity Fusion based on the Siamese Network model in Section III in detail. Section IV provides the experimental results and related analysis, Section V summarizes the contributions of the paper and the future work.

II. RELATED WORK

A. *Sentence similarity calculation*

Sentence similarity calculation is the basis of natural language understanding tasks. Sentence similarity refers to the degree of interchangeability of words between two sentences and the degree of consistency of word meaning [11], which is an index used to evaluate the sentences similarity. From the perspective of information theory, Lin et al. [12] believe that sentence similarity is related to the commonality among sentences. The greater the commonality, the smaller the difference and the higher the similarity. Therefore, the calculation of similarity of two sentences, including S_1 and S_2, shown in Eq. (1).

$$sim(S_1, S_2) = \frac{\log P(common(S_1, S_2))}{\log P(description(S_1, S_2))} \quad (1)$$

Since there are many infect factors in the sentence similarity calculation such as sentence structure, language, syntax, etc., there are various ways to research sentence similarity. The classification methods recognized by most scholars are string-based method, corpus-based method, and knowledge-based Library methods, knowledge-based methods and hybrid methods [13-15]. Among them, this paper utilizes a corpus-based method, specifically a neural network-based method. The neural network-based method takes a corpus to convert a sentence into a vector representation with semantic information as in input for learning. Compared with other methods, the biggest advantage of this method is that it can represent complex contexts.

B. *Convolutional Neural Network*

Convolutional neural network (CNN) is a classic deep learning algorithm. Its basic idea is to use parallel multi-level convolution to perform multi-layer representation of input data, extract feature information of data, and obtain better feature robustness. It was first used in computer vision (CV). With its mature application in the field of computer vision, people have also begun to apply it to text processing, such as the model proposed by Kim et al. in 2014 [16].

After the first success of Convolutional Neural Network in NLP field, more and more people apply CNN in NLP field. Kal et al. [17] proposed a Network model named DCNN (Dynamic Convolutional Neural Network), whose delicacy lies in the use of Dynamic pooling method which can process input of variable length. The network contains two types of layers, namely the one-dimensional convolutional layer and the dynamic k-max pooling layer. The structure of DCNN is shown in Fig. 1. The convolution layer of the network adopts the way of wide convolution, followed by the dynamic k-max pooling layer, which retains the first k maximum values with certain position information. Then, the pooled features are folded, mainly to consider some relation between two adjacent rows, and the model adopts the RAE model idea to extract features hierarchically. The advantage of DCNN is that it does not need any prior information input, nor does it need to construct very complex artificial features.

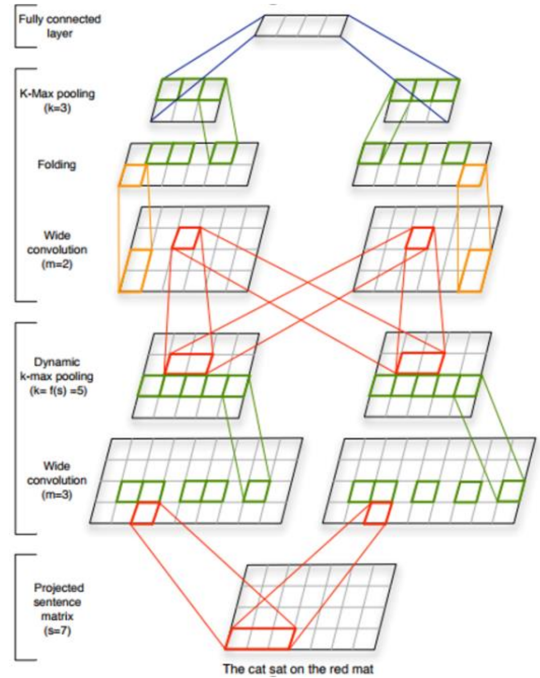

Figure 1. The Framework Of DCNN [17]

Wang et al. [18] proposed a network structure based on similar and dissimilar information, which considered the similarity and dissimilarity of sentences by decomposing and combining the semantics of words, and decomposed two sentences into similar matrix and dissimilarity matrix. He et al. [19] proposed a network structure with multiple perspectives and granularity, which fully excavates the characteristic information of the sentence and improves the performance of the model, but at the same time it makes the model more complex and time-consuming. Ma et al. [20] took into account the dependency information of the sentence and integrated the dependency information into the sentence.

Although the above methods have made some progress in the application of CNN, there is still a lack of consideration of sentence granularity and the problem of long time consumption. Therefore this paper proposes the corresponding sentence vector, extract the character granularity and the word granularity feature respectively.

C. *Siamese Neural Network*

Siamese neural network is a neural network architecture composed of two or more identical subnets, which is widely used in the task of determining the consistency of two kinds of data and measuring the relationship between things [21-23]. One of the architecture of Siamese network is shown in Fig. 2 .

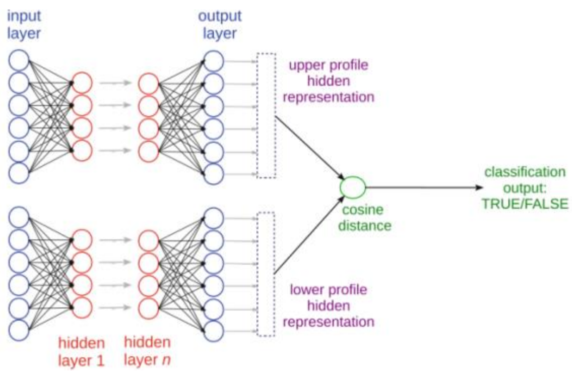

Figure 2. Siamese Network Architecture [24]

The parameters and weights are shared among the subnets, and the parameters are updated at the same time. The main idea is to use the network or function to map the input to the target space, and then use distance calculation formulas such as cosine distance or Euclidean distance to compare the similarity in the target space. If the mapping network or function is $G_W(X)$ and the parameter is W, the similarity measurement result is:

$$E_w(x_1, x_2) = f(G_w(x_1), G_w(x_2)) \tag{2}$$

Due to the sharing parameters between Siamese network subnets, the proposed model training requires fewer parameters, which means that less data is required to train the model to reduce the possibility of over-fitting.

III. MGFSN MODEL

A. Framework Overview

As shown in Fig. 3, the proposed MGFSN architecture is composed of the three components: (1) the word embedding layer, (2) the multi-granularity coding layer, and (3) the semantic interaction layer.

The paper denotes two input Chinese sentences as P = { P_{w1}, P_{w2}, ... , P_{wi}, ... , P_{c1}, P_{c2}, ... , P_{cj} } and Q = { Q_{w1}, Q_{w2}, ... , Q_{wi}, ... , Q_{c1}, Q_{c2}, ... , Q_{cj} }, where *wi* is the i^{th} word of the sentence *P/Q* , *i* is the word length of *P/Q*, *cj* is the j^{th} character of the sentence *P/Q* and *j* is the character length of *P/Q*.

B. Embedding Layer

To construct the appropriate sequence representation, the paper concatenates words embedding including both Chinese words segment representations and characters representations. Using jieba tool [25] to segment sentences, the paper obtains the sequence of the word granularity, and divides it directly through characters to obtain the sequence of the character granularity.

In the word embedding, each word is represented as a d-dimensional vector by using a pre-trained word embedding method such as Word2Vec [26]. In the MGFSN model, a word embedding vector can be divided into two types including word granularity and character granularity.

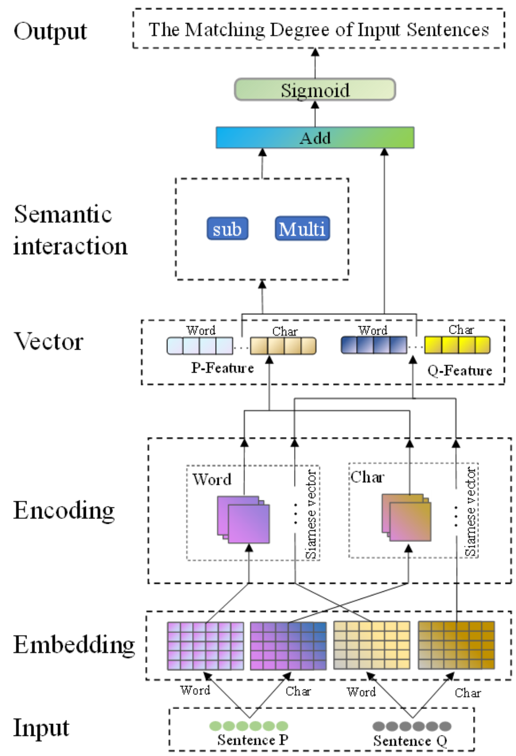

Figure 3. Model Architecture of Sentence Matching

C. Multi-Granularity Fusion Encoding Layer

As the most critical component in MGFSN model, multi-granularity fusion encoding layer, named MGFE layer in abbreviation, will extract features both of word granularity and char granularity via the SNN, in which no external resources are being introduced thus the semantic coding performance can be improved effectively.

As illustrated in Fig. 4, the proposed MGFE layer consists of two different Siamese network that have the same network structure with various weight training. Each Siamese network consists of two identical sub-networks in which *P*-e_w/*Q*-e_w and *P*-e_c/*Q*-e_c means the results of sentence P/Q passing through the embedding layer.

Firstly, for word vector, this paper utilizes self-attention as an attention mechanism,and chooses dot product to calculate the attention matrix in the equation (3-4):

$$f(m_t, m_s) = m_t^T m_s \tag{3}$$

$$a_t = Attention(m_t, m_s) = softmax(f(m_t, m_s))m_v \tag{4}$$

Where $m_t = m_s = m_v = Word\_vectotr$, *f(.)* is the matmul operation and *softmax* is normalized difference index function. This is done after the vectors in order to fully consider the semantic and grammatical connections between the different words in the sentences. And then extracts its features through two convolution neural networks in the equation (5-8):

$$c_1 = Conv(a_t) \tag{5}$$

$$m_1 = MaxPool(c_1) \tag{6}$$

$$c_2 = Conv(m_1) \tag{7}$$

$$m_2 = MaxPool(c_2) \tag{8}$$

Where a_t is the vector after Attention function, c_i is the the result of i^{th} convolution function and m_j is the result of j^{th} MaxPool function.

Meanwhile, for the character vectors, this paper adopts the same network structure with the same operation

Finally, both the character and word granularity are concatenated to obtain more semantic representation information.

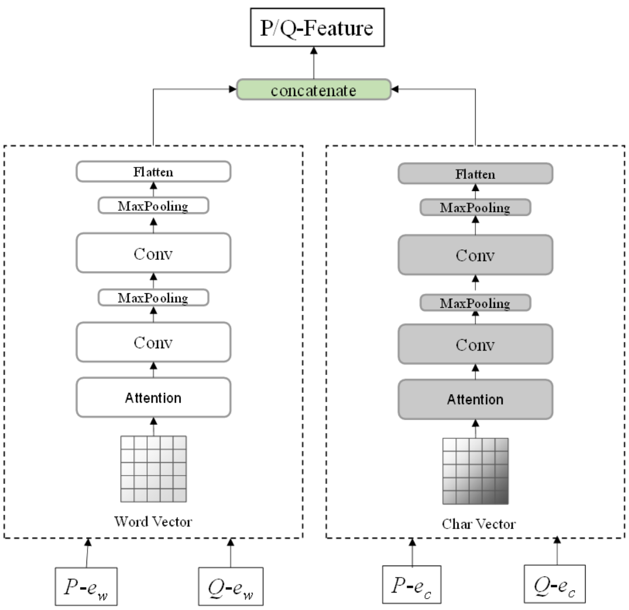

Figure 4. Multi-Granularity Fusion Encoding Layer

D. *Semantic Interaction And Matching Layer*

Semantic interaction and matching layer take multi-granularity fusion encoding layer output feature vector ($\vec{P}$, $\vec{Q}$) that combined word and char granularity as the input as shown in Fig. 5.

During the semantic interaction process, this paper utilizes various ways to compare the similarity of the semantic feature vectors for P and Q. The initial operations are described in the equation (9-11) as follows:

$$\vec{S} = |\vec{P} - \vec{Q}| \tag{9}$$

$$\vec{M} = \vec{P} \times \vec{Q} \tag{10}$$

$$\overrightarrow{Concatenate} = |\vec{S}, \vec{M}| \tag{11}$$

Where $\vec{P}$, $\vec{Q}$ is the output of multi-granularity fusion encoding layer, $\vec{S}$ is the absolute value of $\vec{P}$ minus $\vec{Q}$, $\vec{M}$ is the value of $\vec{P}$ multiply $\vec{Q}$ and $\overrightarrow{\text{Concatenate}}$ is the result of concatenating $\vec{S}$ and $\vec{M}$.

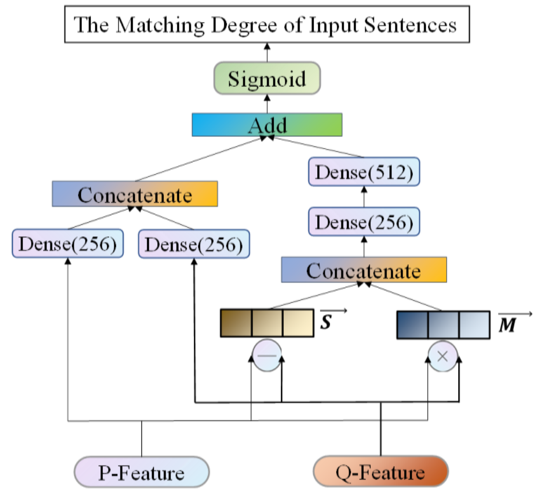

Figure 5. Semantic Interaction And Matching Layer

As shown in Fig. 5, *P-feature* and *Q-feature* are handled by Eq. (9) and Eq. (10) to obtain the vector S and M. Then the paper concatenates both vector *S* and vector *M* via Eq. (11). After that, the results of concatenating are extracted using two dense layers, whose dimensions are 256 and 512 respectively. At the same time, *P-feature* and *Q-feature* are extracted by two dense layers, whose dimensions are 256 respectively. Afterward, this paper adds the two vectors resulted from the above operation with the superposition effect to generates the final matching representation of input sentences, the matching degree, which will be transferred into Sigmoid function.

IV. EXPERIMENT AND ANALYSIS

A. *Data-set*

The data set used in this paper is LCQMC [27], which has a large-scale Chinese question matching corpus contains 260,068 problem pairs with manual annotations. In this paper, it is divided into three parts with the same proportion as in [27], that is, the training set containing 238,766 problem pairs, the development set containing 8,802 problem pairs, and the test set containing 12,500 problem pairs.

Illustrated in TABLE I, each data sample has three attributes: "sentence1", "sentence2" and "Label", sentence 1 and sentence 2 are text pairs. If Label is equal to 1 , it means that the semantics for sentence 1 and sentence 2 is similar, or 0, it means the semantics for sentence 1 and sentence 2 are not similar.

TABLE I. EXAMPLES IN LCQMC CORPUS.

Sentence Pairs	Semantic Match
Q1: 求一款网页游戏 EN: Ask for a web game Q2: 找一款网页游戏 EN: Find a web game	1
Q3:在家带小孩怎么赚钱 EN: How do you make money raising kids at home Q4:有什么工作适合在家带孩子做的 EN: What kind of jobs are suitable for stay at home parenting	0

B. Experimental enviornments

The proposed MGFSN model implements all experiments on a 2080Ti GPU with 11G explicit memory programming by Python based on the Keras and TensorFlow2.0 framework. The parameters are defined as follows.

TABLE II. PARAMETERS OF SIAMESE NETWORK WITH MULTI-GRANULARITY FUSION

Parameters		value
Embedding layer		300
CNN	***filters***	128
	kernel_size	3
	activation	Tanh
Dropout		0.3
Maxpooling		3
Batch size		512
Loss function		binary_crossentropy

The Adam method and its learning rate reduction mechanism have been utilized [28]. The learning rate is initially set to 0.0001. If the accuracy rate on the development set does not increase after 5 epochs, the learning rate will be reduced. In the optimization, epochs is 100, and batch size is 512. In particular, the paper establishes an stop mechanism, in which the training process will automatically stop and verify the performance of the model on the test set, if the accuracy rate on the development set is not improved after 10 epochs.

C. Baseline &Metric

Liu et al.[27] have implemented eight relevant and representative state-of-the-art methods in LCQMC. Those methods have been used as baselines for evaluating the models in this paper.

Unsupervised Methods: word mover distance (WMD), word overlap (Cwo), n-gram overlap (Cngram), edit distance (Dedt), and cosine similarity respectively (Scos) [27].

Supervised Methods: convolutional neural network (CNN), bidirectional long short term memory(BiLSTM), bilateral multi-Perspective matching (BiMPM)[27].

This paper evaluates the Accuracy, Precision, Recall, F1 of all methods. Before calculating, this paper defines: True Positive is abbreviated as TP, FP is abbreviated as False Positive, TN means True Negative, FN means False Negative.

So the calculation formulas are described in the equations (12-15) as follows:

$$\text{Accuracy}=(TP+TN)/(TP+TN+FP+FN) \quad (12)$$

$$\text{Precision}=TP/(TP+FP) \quad (13)$$

$$\text{Recall}=TP/(TP+FN) \quad (14)$$

$$\text{F1-score}=2*\text{Precision}*\text{Rcall}/(\text{Precision}+\text{Recall}) \quad (15)$$

High accuracy and F1-score indicate better performance of the model. Both of them are used in this paper.

D. Performance comparison

Compared to unsupervised methods as shown in TABLE III, WMDchar, WMDword, Cwo, Cngram, Dedt, Scos, the proposed model MGFSN improves the precision metric by 34.75% at the highest and 14.25% at the lowest, recall by 10.87% at the highest and 0.17% at the lowest, F1-score by 24.67% at the highest and 11.77% at the lowest and accuracy by 32.11% at the highest and 13.71% at the lowest.

In contrast to the unsupervised approach, the proposed MGFSN model is a supervised model. MGFSN model can use the error between the real label and the prediction to carry out backpropagation, so as to correct and optimize the massive parameters in the neural network. In additon, since MGFSN model uses multiple granularities, there are more features that are good for similarity judgment. Therefore, the MGFSN has made great progress compared with the unsupervised method.

Compared with the supervised and neural network approach as shown in TABLE III, CBOWchar, CBOWword, CNNchar, CNNword, BiLSTMchar, BiLSTMword, BiMPMchar, BiMPMword, MGFSN improves the precision metric by 14.75% at the highest and 3.55% at the lowest, recall by 6.67% at the highest and -4.43% at the lowest, F1-score by 11.37% at the highest and 0.17% at the lowest and accuracy by 13.81% at the highest and 1.01% at the lowest.

In contrast to the above supervised and neural network approach, MGFSN model not only uses multiple granularity to obtain richer features, but also can extract richer and deeper semantic features because of its deeper network structure.

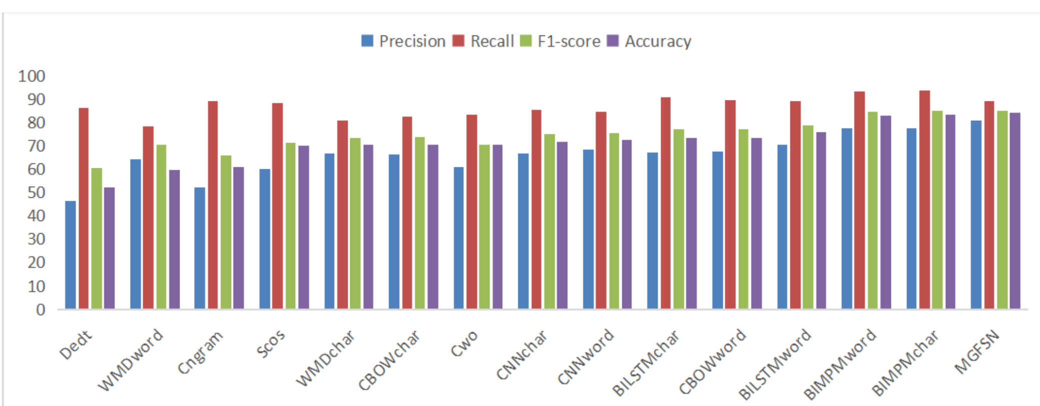

Figure 6. The Histogram of Experiments on LCQMC Sorted by Accuracy

TABLE III. EXPERIMENTS ON LCQMC

Methods	**Precision**	**Recall**	**F1**	**Accuracy**
WMDchar	67.0	81.2	73.4	70.6
WMDword	64.4	78.6	70.8	60.0
Cwo	61.1	83.6	70.6	70.7
Cngram	52.3	89.3	66.0	61.2
Dedt	46.5	86.4	60.5	52.3
Scos	60.1	88.7	71.6	70.3
CBOWchar	66.5	82.8	73.8	70.6
CBOWword	67.9	89.9	77.4	73.7
CNNchar	67.1	85.6	75.2	71.8
CNNword	68.4	84.6	75.7	72.8
BILSTMchar	67.4	91.0	77.5	73.5
BILSTMword	70.6	89.3	78.9	76.1
BIMPMchar	77.6	**93.9**	85.0	83.4
BIMPMword	77.7	93.5	84.9	83.3
MGFSN	**81.25**	89.47	**85.17**	**84.41**

V. CONCLUSION

Here is an explanation to a novel approach based on Siamese Network with Multi-Granularity Fusion for Chinese sentence semantic matching. The MGFSN model in this paper is based on Siamese architecture which reduces the parameters of the model, calculating not only the word granularity of Chinese but also the character granularity of Chinese. In particular, multi-granularity fusion is utilized to obtain more features for similarity matching.

Extensive experiments are carried out on the latest similarity matching benchmark LCQMC. Experimental results show that the proposed approach achieves excellent performances for most of the tasks. In the future, the paper would like to look for various features granularities such as clauses to enrich the features of the sentence, or take different other pre-trained contextual embeddings such as ELMo or BERT to improve performance of the approach.

REFERENCES

[1] Jiang, N.; de Marneffe, M.C. Do you know that Florence is packed with visitors? Evaluating state-of-the-art models of speaker commitment. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 4208–4213.

[2] Wang, Q.; Li, B.; Xiao, T.; Zhu, J.; Li, C.; Wong, D.F.; Chao, L.S. Learning deep transformer models for machine translation. arXiv 2019, arXiv:1906.01787.

[3] Li, H.; Xu, J. Semantic matching in search. Found. Trends Inf. Retr. 2014, 7, 343–469.

[4] HUANG X, ZHANG J, LIU Y, et al. Short text similarity algorithm based on term mapping with semantic[J]. Computer Engineering and Design, 2015 (6): 21. (Chinese)

[5] ZHANG Xiao-chuan, YU Lin-feng, ZHANG Yi-hao. Multi-feature Fusion for Short Text Similarity Calculation Based on LDA[J].Computer Science, 2018, 45(9): 266-270. (Chinese)

[6] Wang, M., Lu, Z., Li, H., Liu, Q.: Syntax-based deep matching of short texts.In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, pp. 1354–1361

[7] Pang, L., Lan, Y., Guo, J., Xu, J., Wan, S., Cheng, X.: Text matching as image recognition. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pp. 2793–2799 (2016)

[8] Hu, B.; Lu, Z.; Li, H.; Chen, Q. Convolutional neural network architectures for matching natural language sentences. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada,8–13 December 2014; pp. 2042–2050.

[9] Lai, Y., Feng, Y., Yu, X., Wang, Z., Xu, K., Zhao, D.: Lattice CNNS for matching based Chinese question answering 33, pp. 6634–6641 (2019)

[10] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1994). Signature verification using a" siamese" time delay neural network. In Advances in neural information processing systems,pp. 737-744

[11] Wang Chunliu. Yang Yonghui, Deng Sin, et al. A review of text similarity calculation methods.Information Science, 2019,37(03) : 158-168.

[12] Deking Lin. An Information Theoretic Definition of Similarity Semantic distance. WordNat Proceedings of the Fiftenth International Conference on Machine Learning,1998

[13] Goma WH,Fahmy A A. Short Answer Grading Using String Similarity And CoruBaed Similarity.Internationl. Journl of Advanced Computer Science and Application,2012.3(11):114-121

[14] KadupitiyaJ,Ranathunga S.Dis G,Short Sentence Similarity Calculation using Corpus-Based and Knowledge-Based Similarity Meaures// Proceeding of the 26th Intermationl Conference on Computational Linguistics. Oaka,2016.

[15] Goma W H.Fahmy AA. A Survey of Text Similarity Approaches .International Journal of Computer Applicatioms,2013.68(13):13-18.

[16] Kim Y. Convolutional ncural networks for sentence classifiation .arXiv preprint arXiv:1408.582,2014.

[17] Kalchbrenner, N. , E. Grefenstette , and P. Blunsom . "A Convolutional Neural Network for Modelling Sentences." Eprint Arxiv 1(2014).

[18] Z. Wang, H. Mi, A. Ittycheriah, Sentence Similarity Learning by Lexical Decomposition and Composition, in: Proceedings of the 26th International Conference on Computational Linguistics, COLING 2016 Technical Papers, 1340–1349, 2016.

[19] Hua, H. , K. Gimpel , and J. Lin . "Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks." Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing 2015.

[20] Ma, M. , et al. "Dependency-based Convolutional Neural Networks for Sentence Embedding." Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers) 2015.

[21] A.He, C. Luo, X. Tian and W. Zeng, "A Twofold Siamese Network for Real-Time Object Tracking," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 4834-4843, doi: 10.1109/CVPR.2018.00508.

[22] Zhao Yuan & Sun Jun.(2020).Siamese Network cooperating with Multi-head Attention for semantic sentence matching .Proceedings of 2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES 2020) ; pp.241-244

[23] Fe Rnando, S. , and M. Stevenson . "A Semantic Similarity Approach to Paraphrase Detection." Computational Linguistics Uk Annual Research Colloquium (2008).

[24] Chicco D. (2021) Siamese Neural Networks: An Overview. In: Cartwright H. (eds) Artificial Neural Networks. Methods in Molecular Biology, vol 2190. Humana, New York, NY. https://doi.org/10.1007/978-1-0716-0826-5_3

[25] Junyi S. jieba. https://github.com/fxsjy/jieba

[26] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. Efficient Estimation of Word Representation in Vector Space. arXiv preprint arXiv: 1301.3781. 2013

[27] Liu, X., et al.: LCQMC: a large-scale Chinese question matching corpus. In: Proceedings of the 27th International Conference on Computational Linguistics, pp.1952–1962 (2018)

[28] Kingma, D. , and J. Ba . "Adam: A Method for Stochastic Optimization." Computer Science (2014).