Learning Long-text Semantic Similarity with Multi-Granularity Semantic Embedding Based on Knowledge Enhancement

Deguang Peng, Bohui Hao, Xianlun Tang, Yingjie Chen and Jian Sun

April 13, 2020

“ACAIT2020+《华南理工大学学报（自然科学版）》(EI(JA))、CAAI Transactions on Intelligence Technology(EI(JA))、《智能系统学报》(CSCD、中文核心）”

Learning Long-text Semantic Similarity with Multi-Granularity Semantic Embedding Based on Knowledge Enhancement

Deguang Peng[1*], Bohui Hao[2], Xianlun Tang[2], Yingjie Chen[2], and Jian Sun[1]
[1] MegaLight AI LAB, Chongqing
[2] Chongqing University of Posts and Telecommunications, Chongqing 400065
Email:pengdeguang@163.com, 993257518@qq.com, tangxl@cqupt.edu.cn, potter1995@live.com, cq_jsun@163.com

Abstract

We propose a new method of semantic similarity calculation-"multi-granular semantic embedding model based on knowledge enhancement (MSE based knowledge)" to solve the similarity and relevance of long text semantic matching. The method firstly enhances semantics through the external knowledge base DBpedia, and simultaneously considers semantic attributes and relationships on the vector representation of key entities. Secondly, each long text is expressed as a multi-granularity vector: character vectors constructed based on one-dimensional convolution, word vectors constructed based on external knowledge sources and pre-trained word vectors, and sentence vectors constructed based on bidirectional LSTM. Furthermore, we use the Siamese network framework to calculate the final similarity. To get better results, we add the attention mechanism after the character vector representation to further weight the key characters. In the end, we evaluate the method on two popular data sets (LP50 and MSRP). Experimental results show that the method in this paper makes better use of long text knowledge and achieves higher accuracy with less time cost.

1 Introduction

Semantic similarity calculation of text is to quantify the strength of semantic relationship between text pairs, which plays an important role in NLP tasks, such as machine translation (QING Chunxiu, 2014), information retrieval (Liu Q, 2016), Q&A system (Zhu X, 2018), etc. Due to the complexity and abstractness of text, semantic similarity calculation still faces great challenges.

In recent years, researchers have used deep neural network models to perform semantic similarity calculations. These methods mainly use a Recurrent Neural Network or a Convolutional Neural

Network to capture the semantic information contained in the text itself, such as DSSM, based on the distributed representation of text. (Huang P.-S., 2013), CLSM (SHEN Y, 2014), and LSTM-RNN (PALANGI H, 2016) models, etc. Recently, lots of research methods have modeled text at various levels of granularity, such as MultiGranCNN model (YIN W, 2015) , the uRAE model (SOCHER R, 2011) , and the MF-LSTM model (WAN S, 2016) .

The research focus of these models is on one or two levels of granularity, and the semantic information of each granularity is simply combined, which leads to two main problems: (1) When humans understand text information, they do not simply and independently understand the meaning of words, words or sentences of the text. The previous method is a segmented understanding of the text, which cannot accurately reflect the meaning of the sentence. How to generate and combine semantic information with different granularities as a complete expression of text semantics and reduce the deviations in calculating the semantic similarity of texts still need to be explored. (2) When relying only on the text itself to calculate semantic similarity, there are problems of semantic similarity and relevance. If the two words in the text are semantically similar, they may be related. Otherwise, they will not work (HadjTaieb MA, 2014), that is, similarity is a special case of relevance (Miller GA, 1991) (Resnik P, 1995). For example, consider the two sentences S1 and S2 below, which are semantically similar.

*S1 ：**Michelle Obama** will travel to the **London Bowl** to watch the opening ceremony.*

*S2 ：**The First Lady of the United States** confirmed that next month will lead the US President to attend the opening ceremony of **the London Olympic Stadium**.*

In this simple scenario, if we use some of the previous methods. For example, when only using a distributed representation of text, certain concepts in a pair of texts, such as ***"Michelle Obama"*** and ***"The First Lady of the United States"***, will have vector expressions that are far away. However, these two words are highly related and are actually similar in semantic level. The machine cannot judge the difference between similarity and relevance. Finally, when calculating the semantic similarity of the text, a low similarity score will be given.

The pre-trained word vector embedding method using a large corpus can have some improvements, such as Word2Vec (Rada R, 1989) and GLOVE (Fellbaum C, 1998). In addition, (Guo, J., 2014) uses bilingual dictionaries to determine word meanings and learn word vectors specific to word meanings, (Yu, 2014) uses relational knowledge as constraints to extend neural language models. These studies are based on a single word vector, which is still limited to the word itself, without comprehensive consideration of semantic information and relational representation.

In order to solve these problems, we propose a multi-granular semantic embedding method based on knowledge enhancement to calculate the semantic similarity of text based on the way that humans judge semantic similarity. In short, our model enhances the existing statistical similarity calculation method based on statistics, and carries out semantic understanding of text pairs from different levels such as words, words, sentences, and knowledge. The main contributions of this article are as follows:

A. Instead of semantically expressing text at the level of characters, words and sentences independently, we use character embeddings, word embeddings, and Bi-LSTM representations for semantic information of different granularities, combining the three methods to avoid segmented semantic understanding and improve the expression of semantic information.

B. We use the text vector expression of knowledge enhancement. Firstly, by extracting the key entities in the text pair, and then introducing the entity relationship in the knowledge base, we consider the semantic attribute and relationship in the vector representation of the key entities at the same time. Therefore, we solve the problem of different similarity and relevance.

C. Based on Siamese network to calculate similarity, the sub-network learns multi-granularity semantic vector of knowledge enhancement to improve the accuracy and completeness of semantic expression. Due to the consistency (same structure, shared parameters) and independence (two completely independent inputs) of the subnetwork, it can better adapt to solve the semantic similarity problem of long text pairs of different lengths.

2 Related Work

Research on text similarity has become one of the hot spots in recent years, and it has been widely used in tasks such as information retrieval, text classification, document clustering, and topic detection. In this field, many technologies have been proposed, we can divide them into two main categories, namely content-based methods and knowledge-rich methods, the main difference is that the former only uses the text information contained in the document, while the latter This information is enriched and the text is refined by extracting information from other sources (usually a knowledge base).

The most representative of the first method is the bag-of-words model, which represents the text as a weighted high-dimensional vector, each dimension corresponds to a feature, and the similarity is calculated by this vector space expression. There are limitations, that is, it cannot solve the problems of similarity and relevance.

Recently, many calculation methods of text similarity based on knowledge bases have been proposed. Explicit Semantic Analysis (ESA) (E. Gabrilovich, 2007) proposes to map documents to Wikipedia articles and represent each document as a vector of features extracted from the document and related text to capture the semantic information which contained in the document and then pass any Vector space comparison algorithm to calculate the similarity of two documents. For example, the method proposed by (N.P. Alexander, 2010) which is based on Wikipedia's contextual ad matching in order to embed candidate ads into relevant pages. (Fabio Benedetti, 2019) Use a general knowledge base to extract semantic context vectors to calculate the similarity between documents. (Wei Lu, 2016) Using corpus and ontology, a CBOW distributed word vector combined with knowledge base is proposed to enrich word semantics. (Xinhua Zhu, 2019) Proposed a bidirectional link vector based on the concept of Wikipedia, in which outlining and inlining are combined into a concept semantic interpreter, and then using TF-IDF-based two-way weighting method to calculate the similarity between concepts. As far as we know, most of these knowledge-based calculation methods of text similarity are at the word level, but the context information of long text still contains many logical semantics. In our method, we combine multi-granular semantic information and knowledge information, and use the Siamese network to estimate the similarity more accurately.

3 Multi-granularity Semantic Embedding Model Based On Knowledge Enhancement

In this section, we propose a multi-granular semantic embedding model based on knowledge enhancement for semantic similarity calculation. As we will introduce in Section 4 later, the most advanced performance has been achieved. Figure 1 shows the main framework of the proposed model, which includes local word vectors, word vector representation, embedding of external knowledge, summarizing and combining local and external information to form a sentence-level global expression, and finally computing semantic similarity.

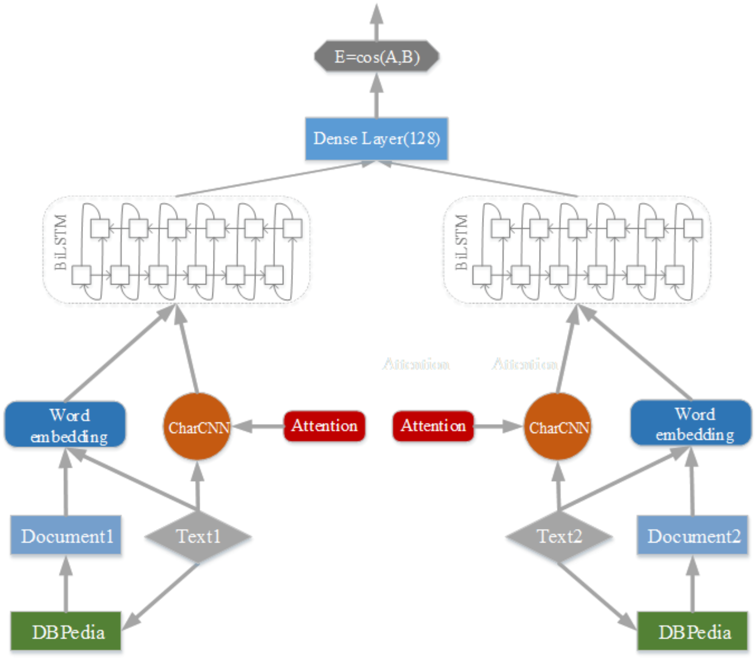


Figure 1: Multi-granularity Semantic Embedding Model Based On Knowledge Enhancement

3.1 Word Vector Embedding

The text character can be regarded as a kind of original signal, corresponding to the use of one-dimensional convolution. We adopt the method of Char-CNN proposed in (Xiang Zhang, 2015): first build a character table, replace all vectors that are not in the character table with all 0s, and do not distinguish between uppercase and lowercase letters.

After that, we convert the characters in the character table into one-hot vectors, then the input text X can be represented as a character vector matrix composed of characters one-hot $C \in \mathbb{R}^{t \times c}$, Where t is the size of the character table and c is the number of data characters. We send character vectors $C \in \mathbb{R}^{t \times c}$ to the temporal convolutional module, then we use a series of discrete convolution kernel functions $f(x) \in \mathbb{R}^{1 \times m}$, finally we get the convolution as follows.

$$h(y) = \sum_{x=1}^{c} f(x) \cdot C_i [y \cdot d - x + q] \tag{1}$$

Where d is stride, q = m − d + 1 is an offset constant. In addition, in order to learn feature weights and extract key information better, we add an attention mechanism, as follows.

$$s_{ti} = f_{att}(\alpha_i, \mathrm{h}_{t-1}) \tag{2}$$

$$a_{ti} = \frac{\exp(\mathrm{s}_{ti})}{\sum_{k=1}^{m} \exp(\mathrm{s}_{tk})} \tag{3}$$

Through the feature extraction of text characters by one-dimensional convolution and the weighting of attention mechanism, the model can learn the superficial semantic information of text.

3.2 Knowledge Enhancement

As discussed above, the semantic information of the text is not independently limited to the inside of the sentence. Therefore, in addition to the distributed expression of text, we also introduced knowledge to enhance the part of word vector embedding.

a. Entity recognition

We give the text X and the knowledge base KB. The first step is to identify the KB entity explicitly mentioned in the document d to form the entity set $SE_d = \{e_1, e_2, \cdots, \mathrm{e}_k\}$ of d. It is well known that finding the collection SE_d is an instance of solving the entity recognition (D. Nadeau, 2007), but the solution is beyond the scope of this article. Thus, we conducted an empirical assessment of some of the techniques already proposed, and finally selected DBpedia Spotlight (P.N. Mendes, 2011) to identify the entities based on the results obtained.

DBpedia Spotlight is one of the most widely used open source named entity recognition systems, which can label named references in natural language text as entities in DBpedia (S. Auer, 2007) knowledge base automatically. In order to improve the recognition accuracy, we will discard some unnecessary entities. DBpedia Spotlight calculates the labeling probability $P(annotation|s)$ of each naming reference s in Wikipedia, and characterizes the importance of a naming reference, so as to discard naming references below a certain threshold, and finally get the entity set SE_d. The formula is as follows.

$$P(annotation|s) = \frac{\sum_{e} count(e, s)}{count(s)} \tag{4}$$

Where e is the set of entities named by singularity reference s in Wikipedia, count (e, s) is the number of times the named s is marked as entity e, and count (s) is the appearance of the named s in Wikipedia The total number of times.

b. Knowledge Base

The Resource Description Framework (RDF) knowledge base can be viewed as a set of statements, each statement is in the form of a triple $\langle (h, r, t) \rangle$, where h, r, and t represent the head entity, relationship, and tail entity respectively, as shown in Figure 2, through relationships Describe the association between two entities. For our work, we chose the general domain knowledge base- “DBpedia”. Dbpedia can be seen as a structured version of Wikipedia, which uses a fixed pattern to extract entity information in Wikipedia, including abstract, category, page link, and info box.

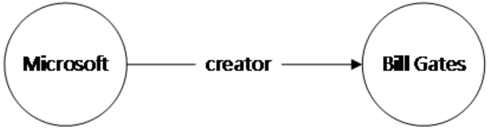


Figure 2: An RDF triple

c. Knowledge Representation

The main goal of knowledge representation is to learn the representations of entities, relationships, and related entities based on the structural information of triples in the knowledge base, so as to enrich the semantic expression (Alexander Miller, 2016) . Due to the logic of the language expression, there is a great possibility that the two entities in the document $e_1, e_2 \in SE_d$ have a certain correlation. Therefore, we define $r_{1,2}$ as the relationship between entities e_1, e_2 in the knowledge base KB. A direct relationship in the knowledge base is marked as 1, p is the relationship threshold, and relationships that exceed the threshold are discarded. Besides, for local entities e_i that are not related, we select the fact $\tilde{e}_i$ that has the highest probability of similar meaning in the knowledge base according to Wikipedia statistical data. We collectively refer to relations $r_{i,j}$ and facts $\tilde{e}_k$ as knowledge-enhanced expressions of text vectors, denoted as RE. Then, we combine the knowledge representation with the original text and make full use of the composition information and context information of the expressions to obtain a knowledge-enhanced text representation $\tilde{X}$.

3.3 Word Vector Distributed Representation

The input text is a sequence of words. In (T. Mikolov, 2013), the author proposed two word representation models: (1) continuous BOW (CBOW), and (2) continuous skip-gram. The input and output of these two models are different. For CBOW, the model predicts output expressions based on a given context, while skip-gram can predict context based on a given expression. Given an expression, we use word2vec to find its pre-training vectors to get the word vector matrix $\tilde{X} \in \mathbb{R}^{d \times l}$ of the input text, where d is the word vector dimension and l is the text length.

3.4 Text Representation

In order to get the representation of the sentence vector, we input the word vector matrix $\tilde{X}$ to the Bi-LSTM network. LSTM is a special RNN, which changes simple hidden layer nodes to storage units to effectively solve the problem of gradient disappearance. The unit of LSTM consists of three gates and memory cells. These three gates are input gate, output gate and forget gate (Fei Liu, 2017) . The input gate determines the number of input is currently stored in the storage unit. The output gate controls the least information the memory unit output to the current value of the LSTM. The forget gate determines the quantity of the last moment's memory cell state can be retained at the current moment. The entire process can be expressed by the following formulas.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{5}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{6}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \tag{7}$$

$$c_t = f_t * c_{t-1} + i_t * tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{8}$$

$$h_t = o_t * tanh(c_t) \tag{9}$$

Among them, i, f, o represent the above unit gates, which are: input gate, forget gate and output gate respectively. c is the state of the storage unit, which is continuously updated with the time series. h represents the output of the hidden layer. c must be equal to the hidden vector h. W and b respectively represent the corresponding weight coefficient matrix and deviation. σ and tanh represent Sigmoid and Hyperbolic tangent activation function, respectively. Sigmoid activation function, the output value is limited to $[0, 1]$, 0 means completely abandoned, 1 means completely passed. The hyperbolic tangent activation function limits the output value to $[1, -1]$ (Jason D Williams, 2017). The formula is as follows.

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{10}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{11}$$

The context feature sequence obtained through Bi-LSTM is $H=\{h_1, h_2, \cdots h_l\}$, Each part of the feature vector is generated by the connections of forward and backward LTM, which is $h_t = \vec{h}_t \| \overleftarrow{h}_t$, where ‖ is the connection operation.

3.5 Similarity Calculation

The Siamese network (Sumit Chopra, 2005) is a network structure with two or more identical branches, each of which has the same parameters and weights as well as performs parameter updates simultaneously. Inspired by (Neculoiu, P., 2016), we calculate the similarity between text representation vectors through the Siamese network structure. The input of the network is text pairs X_1 and X_2, the purpose of training is to minimize the distance in the embedded space between similar pairs and maximize the distance between dissimilar pairs. Then, get the vector expression of the text pair H_1 and H_2 , and the energy E of the model to be the cosine similarity between H_1 and H_2.

$$E\left(X_1, X_2\right) = \frac{\left\langle H_1, H_2 \right\rangle}{\left\| H_1 \right\| \left\| H_2 \right\|} \tag{12}$$

The contrastive loss can express the degree of matching of sample pairs properly. The formula is as follows.

$$L = \frac{1}{2N} \sum_{n=1}^{N} yd^2 + \left(1-y\right) max\left(margin - d, 0\right)^2 \tag{13}$$

$d = \left\| H_{1n} - H_{2n} \right\|_2$, is the Euclidean distance of text pair features, y = 1 \ 0 indicates that the text pair is similar or dissimilar, margin is a set threshold, and N is the number of samples.

4 Experiment

4.1 Dataset

We perform experiments on the following two long document datasets : LP50 and Microsoft Research Paraphrase Corpus (MSRP). The datasets are described as Table 1.

A. LP50 (Lee, Pincombe, and Welsh 2005) include 50 documents from Australian Broadcasting Corporation News mail, which was evaluated by 83 students from Adelaide University, and each pair of document pairs (total of 1225 pairs) had 8-12 manual judgments. These manual judgments have averaged each document pair, that is, only 67 different values are obtained for 1225 similarity scores. Each pair of sentences has a score [1, 5], which represents the degree of relevance between the two sentences. The higher the score of the sentence pair, the stronger the relevance of the two sentences.

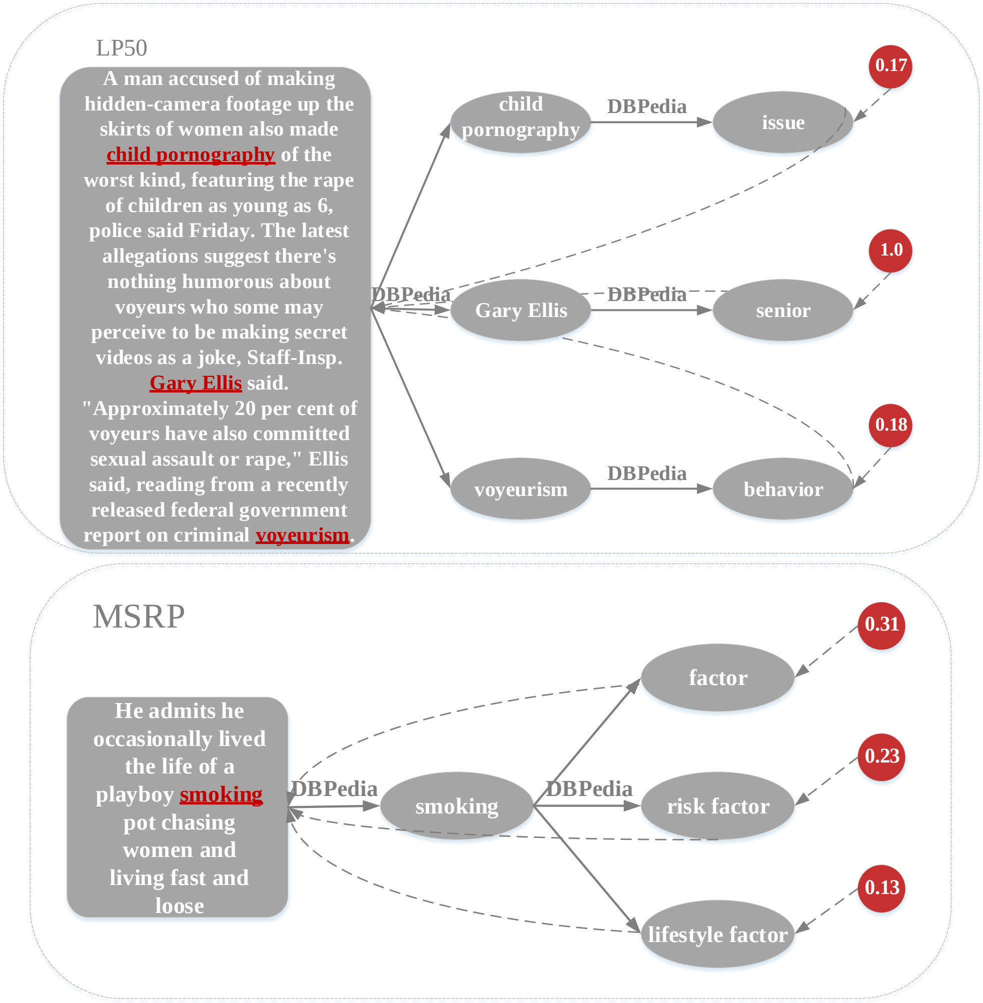
B. MSRP. It contains 5801 sentence pairs extracted from news articles on the Internet and then labeled by some experts using specific techniques. We randomly segment the data set and divide it into a training set and a test set. The task requirement of MSRP is to give two sentences, and then use an algorithm to judge whether the two sentences are semantically similar. If each pair of sentences in the dataset is semantically similar, the corresponding label is 1, otherwise, the corresponding label is 0. Among them, semantically similar sentence pairs account for approximately 66% of the total.

dataset	context	Quantity of text pairs	Picked maximum length
LP50	Train	11003	100
	Test	1222	
MSRP	Train	4076	30
	Test	1725	

Table 1: The quantity of text in dataset

4.2 Results and Discussion

The MSE based knowledge method proposed in this paper: first input the sentence pairs of the original data set into the Siamese frame at the same time, and train the sentence pairs separately with the same weight. In the input part, first use the DBpedia knowledge base to extract an entity set containing n entities. Obtain the subject relationship of the entity set in DBpedia, that is, the knowledge representation of the original data set. In this way, the data after knowledge representation not only gets more detailed knowledge expression, but also realizes further data expansion. Taking an input text as an example, the specific operation process of the knowledge part on the two datasets is shown in the Figure 3.


A man accused of making hidden-camera footage up the skirts of women also made child pornography of the worst kind, featuring the rape of children as young as 6, police said Friday. The latest allegations suggest there's nothing humorous about voyeurs who some may perceive to be making secret videos as a joke, Staff-Insp. Gary Ellis said. "Approximately 20 per cent of voyeurs have also committed sexual assault or rape," Ellis said, reading from a recently released federal government report on criminal voyeurism.

He admits he occasionally lived the life of a playboy smoking pot chasing women and living fast and loose

Figure 3: Knowledge Representation Process

In order to compare the improvement of long text similarity calculation combined with the subject relationship in the knowledge base, Table 2 below lists the similarity scores obtained by a pair of sentence pairs on the two datasets before and after the knowledge representation. Sentences_pair represents the original text pair, labels represents the similarity of the sentence pair to the original, B represents the similarity obtained before the knowledge representation, and A represents the similarity obtained after the knowledge representation.

Sentences_pair	labels	B	A
1. Wells Fargo and Quicken Loans couldn’t be reached for comment Wednesday afternoon 2. Wells Fargo was not available for comment and a Quicken Loans spokeswoman declined immediate comment	0	0.48	0.44

Table 2: The similarity of the sentence after/before knowledge representation

It can be seen from Table 2 that the text pair comes from the LP50 dataset. The similarity of the sentence pair before knowledge representation is 0.48, and after knowledge representation is 0.44, 0.44 is closer to the original text label 0. This phenomenon can explain that this article uses the text knowledge representation based on the DBpedia knowledge base to improve the accuracy of similarity calculation.

In Table 3, the MSE based knowledge method in this paper is compared with other literature techniques, such as a simple document representation based on a bag-of-words model that combines word frequency weights and cosine similarity, and uses Okapi BM25 combined with dot product for weight representation. No background LSA (only considering LP50 data set) and background LSA (using other documents for dimensionality reduction).

Model	Pearson coefficient (*r*)
Bag-of-Words (M.D. Lee, 2005)	0.41
BM25	0.50
Un-Backgrounded LSA (M.D. Lee, 2005)	0.52
Backgrounded LSA (M.D. Lee, 2005)	0.59
ESA reimplemented (D. Bär, 2011)	0.59
GED-based (Dbpedia) (M. Schuhmacher, 2014)	0.63
CSA	0.62
Knowledge-based MSE	0.67

Table 3: Performance comparison on dataset LP50

It can be seen from Table 3: By comparing the Pearson coefficient, the model in this paper is improved by 26% compared with the method of modeling the document corpus in the standard bag-of-words vector space model, and has surpassed the current latest technology LSA.

In order to explore the impact of dropout on the performance of the model in this paper on the MSRP data set, Figure 4 is the model accuracy results corresponding to different dropouts. A dropout ratio of 0 means that no hidden nodes are discarded, that is, overfitting is not performed using dropout. As can be seen from Figure 4, the dropout is set to 0 in this experiment, that is, when no hidden nodes are dropped, the highest accuracy can be achieved.

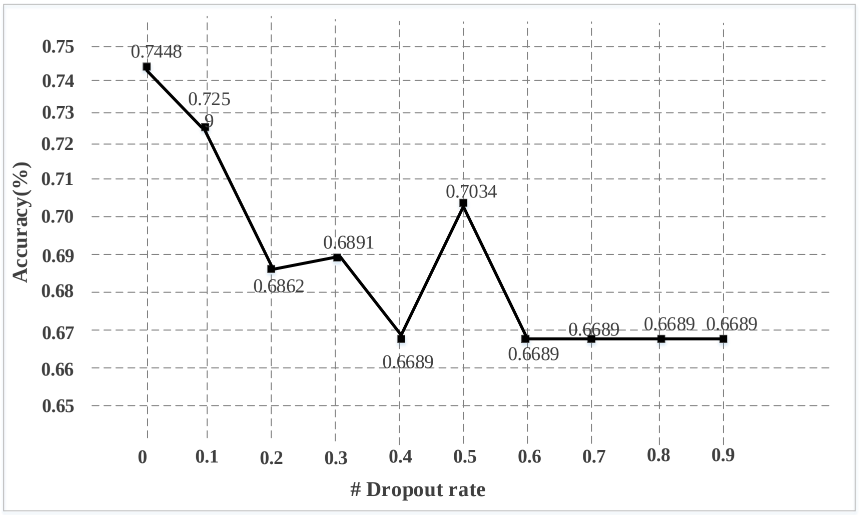


Figure 4: Impact of Dropout on Model Performance on MSRP

Table 4 is the performance comparison of each model on the MSRP dataset. In order to compare with the previous research, we used the accuracy and F1 score as the evaluation criteria for this experiment, in which the algorithm filled with gray is based on Neural Network.

model	accuracy	F1
BASELINE	65.4%	75.3%
Hu et al. (2014) ARC-I	69.6%	80.3%
Hu et al. (2014) ARC-II	69.9%	80.9%
Blacoe and Lapata (2012)	73.0%	82.3%
Fern and Stevenson (2009)	74.1%	82.4%
Yin et al. (2015)(without pretraining)	72.5%	81.4%
Knowledge-based MSE	74.5%	82.7%

Table 4: Performance comparison on dataset MSRP

It can be seen from Table 4 that the performance of the model in this paper is significantly higher than that of BASELINE (Fernando S, 2008). The model of Hu et al. (Hu B, 2014) is based on Cross-Convolution, which simultaneously uses sentence pair information during the convolution process. Although Hu et al. uses cross-convolution, it does not deal with the deep information mining of sentences. The model in this paper extracts features at three levels: word, word and sentence, and fully mines the information of the sentence itself, so it can achieve better results. Yin et al. (Yin W, 2015) used a pre-training technique to enhance the input data information. After using this technique, the experimental results can be greatly improved. The data set is not necessarily applicable. The other two experimental results based on the neural network model are currently the best. It can be seen from this that the method in this paper can better calculate the similarity between documents based on the deep model learning of long sentence level features, adding knowledge part representation and combining information between long texts.

5 Conclusion

In this paper, we propose a deep neural network model (MSE based knowledge) based on the Siamese framework and combined with the knowledge base to express long text similarity. In this model, we choose the Siamese network for training. Firstly, external knowledge sources are used to express knowledge of subject relations, that is, characters, words, sentences and other granularities are used to express documents. Then add an attention mechanism after the text representation to extract key information, and finally input into the two-way LSTM for similarity calculation. We have proved the effectiveness of the method in this experiment. This method achieves an accuracy rate far exceeding that of BASELINE with a shorter time cost, and surpasses a better model implemented in the field of text similarity calculation in recent years.

6 Acknowledgements

This work is supported by the National Nature Science Foundation of China under Project 61673079, 61703068 and the Natural Science Foundation of Chongqing under Project cstc2018jcyjAX0160.

References

QING Chunxiu, ZHU Ting, ZHAO Pengwei, Zhang Yi. Research on Semantic Analysis of Natural Language[J]. Library and Information Service. 2014, 58(22): 130-137.

Liu Q, Liu B, Zhang Y, Kim DS, Gao Z (2016) Improving opinion aspect extraction using semantic similarity and aspect associations. In: Proceedings of AAAI 2016, pp 2986–2992.

Zhu X, Yang X, Chen H (2018). A biomedical question answering system based on SNOMED-CT. In: Proceedings of the 11th international conference on knowledge science. Engineering and management, pp 16–28.

Huang P.-S., He X., Gao J., Deng L., Acero A., and Heck L. 2013. Learning deep structured semantic models for web search using click through data. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM), 2333–2338.

SHEN Y, HE X, GAO J, et al. 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. New York, USA, 2014: 101-110.

PALANGI H, DENG L, SHEN Y, et al. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2016, 24(4): 694-707.

YIN W, SCHÜTZE T, HINRICH. MultiGranCNN: An Architecture for General Matching of Text Chunks on Multiple Levels of Granularity[C]. Proceedings of the 53rd Annual meeting of the association for computational linguistics, Beijing, China, 2015: 63-73.

SOCHER R, HUANG E H, PENNIN J, et al. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection[C]. Proceedings of the Advances in Neural Information Processing Systems, Granada, Spain, 2011: 801-809.

WAN S, LAN Y, GUO J, et al. A deep architecture for semantic matching with multiple positional sentence representations[C]. Proceedings of the 30th AAAI Conference on Artificial Intelligence. Phoenix, USA, 2016: 2835-2841.

HadjTaieb MA, Aouicha MB, Hamadou AB (2014) A new semantic relatedness measurement using WordNet features. KnowlInfSyst 41:467–497

Miller GA, Charles WG (1991) Contextual correlates of semantic similarity. Lang Cogn Process 6:1–28

Resnik P. Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of 14th international joint conference on artificial intelligence, IJCAI, vol 1995. MorganKaufmann Publishers Inc., Montreal, Quebec, pp 448–453

Rada R, Mili H, Bicknell E, et al. Development and application of a metric on semantic nets[J]. IEEE Transaction on System Man & Cybernetics, 1989, 19(1):17-30.

Fellbaum C, Miller G. Combining Local Context and Wordnet Similarity for Word Sense Identification[M]// Word Net: An Electronic Lexical Database. 1998:265--283.

Guo, J., Che, W., Wang, H., & Liu, T. (2014). Learning sense-specific word embeddings by exploiting bilingual resources. Proceedings of COLING (pp. 497–507).

Yu, M., &Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics(Vol. 2, pp. 545–550).

E. Gabrilovich, S. Markovitch, Computing semantic relatedness using wikipedia-based explicit semantic analysis, in: Proceedings of the 20th International Joint Conference on Artifical Intelligence, IJCAI'07, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007, pp. 1606–1611.

N.P. Alexander, C. Chin-Wan, A wikipedia matching approach to contextual advertising, World Wide Web 13 (3) (2010) 251–274.

Fabio Benedetti, Domenico Beneventano, Sonia Bergamaschi, Giovanni Simonini, Computing inter-document similarity with Context Semantic Analysis, Information Systems 80 (2019) 136–147.

Wei Lu, Kailun Shi, Yuanyuan Cai, Xiaoping Che, Semantic Similarity Measurement Using Knowledge-Augmented Multiple-prototype Distributed Word Vector, International Journal of Interdisciplinary Telecommunications and Networking, volume 8 • Issue 2 • April-June 2016.

Xinhua Zhu, Qingsong Guo. Bo Zhang, Fei Li, An efficient approach for measuring semantic relatedness using Wikipedia bidirectional links, Applied Intelligence (2019) 49:3708–3730.

Xiang Zhang, Junbo Zhao, Yann LeCun, Character-level convolutional networks for text classification. Advances in Neural Information Processing Systems, p 649-657, 2015.

D. Nadeau, S. Sekine, A survey of named entity recognition and classification, Linguist. Investig. 30 (1) (2007) 3–26.

P.N. Mendes, M. Jakob, A. García-Silva, C. Bizer, Dbpedia spotlight: Shedding light on the web of documents, in: Proceedings of the 7th International Conference on Semantic Systems, I-Semantics ’11, ACM, New York, NY, USA, 2011, pp. 1–8.

S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. Ives, DBpedia: A Nucleus for a Web of Open Data, Springer, 2007.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. In Proceedings of the 2016Conference on Empirical Methods in Natural Language Processing, pages 1400–1409, Austin, Texas. Association for Computational Linguistics.

T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, Clin. Orthop. Relat. Res. (2013) abs/1301.3781.

Fei Liu, Julien Perez. 2017. Gated end-to-end memory networks. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 1–10, Valencia, Spain. Association for Computational Linguistics.

Jason D Williams, Kavosh Asadi, and Geoffrey Zweig.2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics.

Sumit Chopra, RaiaHadsell, and YannLeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pages 539–546. IEEE.

Neculoiu, P., Versteegh, M. and Rotaru, M. (2016) Learning Text Similarity with Siamese Recurrent Networks. Proceedings of the 1st Workshop on Representation Learning for NLP, Berlin, 11 August 2016, 148-157.

M.D. Lee, M. Welsh, An empirical evaluation of models of text document similarity, in: Proceedings of the XXVII Annual Conference of the Cognitive Science Society, CogSci, Erlbaum, 2005, pp. 1254–1259.

D. Bär, T. Zesch, I. Gurevych, A reflective view on text similarity, in: Recent Advances in Natural Language Processing, RANLP 2011, 12–14 September, 2011,Hissar, Bulgaria, 2011, pp. 515–520.

M. Schuhmacher, S.P. Ponzetto, Knowledge-based graph document modeling, in: Proceedings of the 7th ACM International Conference on Web Search and Data Mining, WSDM ’14, ACM, New York, NY, USA, 2014, pp. 543–552.

Fernando S, Stevenson M. A semantic similarity approach to paraphrase detection[C]// Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics. 2008: 45-52.

Hu B, Lu Z, Li H, et al. Convolutional neural network architectures for matching natural language sentences[C]// Advances in neural information processing systems. 2014: 2042-2050.

Yin W, Schütze H. Convolutional Neural Network for Paraphrase Identification[C]// Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015:901-911.