International Journal of Information Technologies and Systems Approach
Volume 16 • Issue 2

SRU-based Multi-angle Enhanced Network for Semantic Text Similarity Calculation of Big Data Language Model

Jing Huang, School of Information Science and Engineering, Zhejiang Sci-Tech University, China*
Keyu Ma, School of Computer Science and Technology, Zhejiang Sci-Tech University, China

ABSTRACT

As a fundamental problem of natural language processing (NLP), the calculation of semantic text similarity plays a crucial role in a variety of big data application situations. In the process of text similarity modeling, however, owing to the complexity and ambiguity of Chinese semantics, effectively capturing the semantic interaction characteristics of Chinese text only from a single angle is impossible. This study proposes a deep learning-based computational model for semantic text similarity called SRU-based multi-angle enhanced network (SMAEN). Specifically, the authors firstly combine character-grained embeddings and word-granularity embeddings obtained from the pre-trained model to represent text. The text is encoded using a bidirectional simple recurrent unit (Bi-SRU) network, and the local text similarity is represented using a soft-aligned attention technique. In addition, the authors integrate Bi-SRU with an improved convolutional neural network (CNN) for global similarity modeling to capture semantic, time, and spatial characteristics of short text interaction. Finally, they employ a pooling layer to aggregate the calculation results into a fixed-length vector and a multi-layer perceptual (MLP) classifier to make a determination. Experimental results on Chinese public datasets LCQMC and PAWS-X show that the proposed method fully captures semantic interaction features from multiple angles and achieves advanced performance. This method can produce better matching results and enhance the accuracy of large data analysis. It is applicable to numerous scenarios involving large data, such as information retrieval and recommendation systems.

KEYWORDS

Big Data, Chinese Semantic Text Similarity, Deep Learning, Natural Language Processing, Semantic Interaction, Simple Recurrent Unit

INTRODUCTION

In the big data era, how to accurately find the required data from massive texts is an important issue (Mohamed et al., 2020). The advancement of deep learning and big data (Wu et al., 2021; Liu, 2022) provides excellent support. Big data application research relies on the calculation of semantic text similarity (STS). Because of the drastic advancement of deep learning, the effect of calculating STS has been significantly enhanced.

STS calculation (Chen et al., 2021) is the essential topic in natural language processing (NLP), which is used to determine the similarities of two text pieces in a variety of big data applications,

DOI: 10.4018/IJITSA.319039 *Corresponding Author

such as information retrieval (Song et al., 2018; Chen et al., 2021), automatic question answering (Chen, & Xu., 2021; Hu et al., 2021; Zheng et al., 2021), machine translation (Mistree et al., 2022; Niu et al., 2021; Cheng et al., 2021), and recommendation system (Gong et al., 2021; Ghasemi, & Momtazi, 2021). These tasks can be abstracted predominately as text semantic matching problems. The information retrieval task finds matching documents through user queries. The automatic question answering task finds the most appropriate candidate answer based on the question's relevance. The task of machine translation is to match two languages based on their relevance. The recommendation system matches the relevant metrics that the user may be interested in with the user's behavior. In light of the fact that the rich semantic information provided within the text cannot be fully used, the similarity calculation of Chinese semantic text still faces great challenges.

Chinese text is abstract and complicated, and the standards for representing Chinese text are stricter. A CNN or recurrent neural network (RNN) is typically employed to encode text. Long short-term memory (LSTM) and gate recurrent unit (GRU) can effectively mitigate the gradient disappearance issue. However, the scalability of the cyclic neural network is extremely poor, and the CNN requires an enormous amount of calculations that cannot be balanced within the model's capacity.

Recently, numerous scholars have made major contributions to STS activities, which can be grouped into three categories. The first category includes conventional methods that focus solely on the literal resemblance of elements, such as words, string sequences, and phrases between texts, and have significant limits, mainly including Jaccard distance and SimHash.

The second category relies on machine learning techniques that represent text as vectors and analyze semantic similarity using statistical methods, primarily including vector space model (VSM), latent semantic analysis (LSA), and others. However, the consideration of the position of words is ignored and performance in complex tasks is not so good.

The third group depends on deep learning approaches that use deep learning models to capture semantic information and interaction features from text. There are primarily three frameworks involved. One framework is a representational framework. The main idea is the "Siamese structure" (Bromley et al., 1993); it uses two symmetric networks to represent the text and calculate the similarity. It also has shared parameters and low complexity. The typical examples include deep structured semantic mode (DSSM) (Huang et al., 2013; Chen et al., 2020), and ARC-I (Hu et al., 2014). This framework lacks semantic interaction information during the encoding process and cannot measure the contextual importance of words. As a result, an interactive framework is proposed with "matching aggregation" as the central concept (Wang, & Jiang, 2016) and the attention mechanism used to boost textual interaction by collecting both interactive and semantic information. As a third framework, the pretraining model (Devlin et al., 2019; Liu et al., 2019; Zhang et al., 2021) is used to complete specific matching tasks by fine-tuning the model. Although its accuracy has improved, its order of magnitude, parameter size, and time cost are higher than the previous two frameworks. This model also has a big problem in balancing model capacity and accuracy.

Although significant advancements in existing techniques have been made, the current research still faces two difficulties. First, Chinese is more complex than English in terms of grammatical structure and context expression; the standards for Chinese text representation are stricter, and Chinese characters have richer semantic information. Embeddings at a single granularity do not represent Chinese text well, requiring a mixed embedding of multiple granularities and a lightweight network to encode the text. Second, during the semantic interaction process, capturing semantic interaction features only from a single perspective cannot facilitate more comprehensive decision-making in global analysis.

To solve the aforementioned difficulties, this research provides an innovative method for calculating semantic text similarity that optimizes interaction features and text representation, outperforming existing methods.

This paper has three main contributions.

- We established a special model, SMAEN, that uses semantics and interaction information to the fullest extent and provides an effective method for solving semantic text similarity calculation;
- We combined character embedding and word embedding to convey the entire meaning of Chinese text, and use lightweight Bi-SRU instead of bidirectional LSTM (Bi-LSTM) for text encoding;
- By combining simple recurrent networks and CNNs, we captured finer-grained interaction features between texts from three perspectives: semantics, time, and space.

In the remaining sections of this paper, we provide a summary of relevant work, describe our proposed approach, cover the relevant experimental data and analyses, and summarize our study.

RELATED WORK

Because of the limitations of traditional text similarity calculation methods, deep learning methods extract more grammatical, semantic, and structural information of short texts. In recent years, the traditional method for estimating text similarity based on statistics has been largely replaced.

Based on the Siamese network architecture (Huang et al., 2013; Thyagarajan, 2016), two identical encoders map text pairs into the same space and perform similarity calculations on two text vectors simply using the distance between vectors. However, the two texts are independent of each other in the encoding process; they lack clear interactive information, leading to the loss of vital information. Some researchers have applied the attention machine to the encoding layer to strengthen the interaction between texts (Lin et al., 2017; Yuan, & Jun, 2020). However, the effect has not been much enhanced.

On the basis of the match-aggregation architecture, an encoder of RNN or CNN is typically employed to encode two texts into vectors of equal length and to record the matching signal of two smaller text units (such as characters, words, or contextual information). The matching findings are then aggregated, and a global analysis of the similarity is carried out. This framework essentially interacts through a certain technology or method in the two twin networks, significantly improving the interaction ability, and the semantic focus and interaction information are better grasped. For instance, the enhanced sequential inference model (ESIM) (Chen et al., 2017) is used to construct a network using Bi-LSTM that reaches cutting-edge performance by improving the interaction between two phrases using an attention mechanism.

However, single-angle matching cannot capture the interaction information between sentence pairs. Inspired by the concept of ESIM, Wang et al. (2017) describe a bilateral multi-perspective matching model (BIMPM) used to build a network with Bi-LSTM that achieves four different matching algorithms and improves the ability to extract sentence interaction features. However, the captured features are confined to word-level characteristics. Enhanced recurrent convolutional neural networks (ERCNN) (Peng et al., 2020) combine CNN and RNN to capture the similarity and difference between sentence pairs on the basis of ESIM. This approach is more effective than ESIM at capturing the interaction information of sentences' key points.

Kim et al. (2019) describe a deeply recursive convolutional network (DRCN) that executes numerous rounds of the encoder and connects interaction modules via residual connections, preventing information loss by preserving the original information and common attention feature information from the bottom word embedding layer to the top recurrent layer. Deep interaction text matching (DITM) (Yu et al., 2021) uses multiple loop interaction modules to gather deep interaction information of text and extracts information using multi-angle pooling to forecast the relationship between text pairings. In many NLP text matching tasks, it has obtained the top results. Multiway semantic interaction based on multi-granularity semantic embedding (MSIM) (Tang et al., 2022) uses multiple semantic interactions to estimate text similarity, enabling the model to capture as much interaction information between phrases as possible with superior performance. Enhanced attentive convolutional neural networks (EACNN) (Xu et al., 2020) adopt three attention mechanisms and full use of CNN features to obtain sentence interaction features. These features are highly competitive in training efficiency

compared with LSTM. Although the aforementioned methods explore the capture of the semantic interaction features between texts, the semantic alignment and interaction information between text pairs cannot be fully extracted from only a single perspective. It is necessary to capture the semantic interaction features between texts from multiple perspectives.

In the process of similarity calculation, the semantic features of sentences and words are taken into account, and good results are obtained. Architecture for general matching of text chunks on multiple levels of granularity (MultiGranCNN) (Yin, & Schütze, 2015) employs CNN to extract text information of various granularities, including words, phrases, and sentences, and then measures the similarity between two sentences. This MultGranCNN architecture improves the preservation of the detailed information of sentences and the accuracy of text matching. Multi-channel information crossing (MIX) (Chen et al., 2018) augments the multi-channel convolutional neural network with an attention mechanism. Moreover, it extracts multi-granularity text features by parsing sentences into text segments of various granularities, including unary, binary, and ternary. Although the approaches all employ granular semantic elements, such as words, phrases, and sentences, for text representation, they continue to disregard the influence of character granularity on Chinese text. The semantic information of character granularity and word granularity should therefore be used to the fullest extent.

This research offers a multi-angle enhanced network based on simple recurrent units by using the concept of interactive framework matching aggregation in an effort to address the deficiencies of the existing Chinese semantic text similarity calculation. The focus is on capturing the interaction information and semantic information of sentences in depth from the three different angles of semantics, time, and space, and the global analysis of sentence similarity is improved. It combines Bi-SRU and CNN to calculate text similarity from many angles using multi-granularity mixed embedding and an encoder based on a soft attention approach.

MODEL INTRODUCTION

We present SMAEN for Chinese semantics text similarity calculation to completely express the semantic features of Chinese texts and account for interaction information in multi-angle enhancement global similarity modeling. Figure 1 depicts the primary structure of SMAEN, which comprises three components: an input encoding layer, local similarity modeling, and global similarity modeling.

$A = \left\{a_1, a_2, ..., a_m\right\}$ and $B = \left\{b_1, b_2, ..., b_n\right\}$ denote two input sentences, where a_i or $b_j \in \mathbb{R}^k$, m and n are the lengths of the two sentences, respectively, and n is the length of the two sentences, respectively. Predicting the similarity label $y \in \left\{0,1\right\}$ is the objective between A and B, where 0 means that sentences A and B are semantically dissimilar, and 1 means that sentences A and B are semantically similar.

Input Encoding

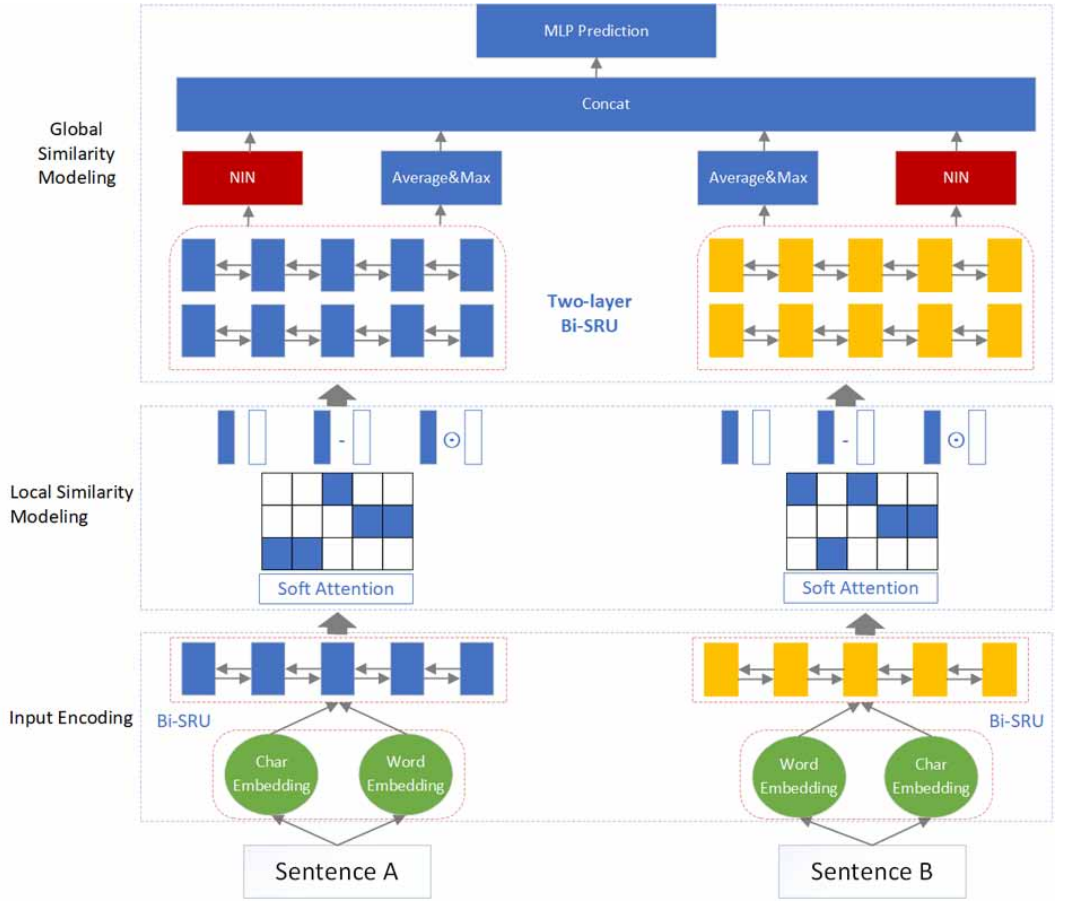
In this layer, we perform multi-granularity embedding operations and encoding operations on the input sentences to extract the contextual features of the sentences.

Embedding

In Chinese semantic text similarity calculation, sentences can be represented as words or characters. Chinese characters are the most basic expression of Chinese sentences, while words have richer semantic information than Chinese characters. Character embedding-based models outperform word embedding-based models in the majority of Chinese NLP tasks (Cheng et al., 2021). Therefore, we can add character embedding on the basis of word embedding to increase fine-grained semantic information.

The two input Chinese sentences are segmented using the Jieba word segmentation tool, and the embedding of the word granularity and the embedding of the character granularity are integrated to generate the sentence representation. Word embedding and character embedding are pretrained by

Figure 1. Structure of SMAEN model



word2vec (Mikolov et al., 2013) on Wikipedia Chinese corpus and Baidu Encyclopedia Chinese corpus. Because trainable word embeddings are prone to overfitting, we fix the word embeddings. We set the word vector and character vector dimensions to 300. For each sentence, the word vector matrices $A \in \mathbb{R}^{m \times d}$ and $B \in \mathbb{R}^{n \times d}$ are determined, and d is the word vector's dimension.

Encoding

After the word and character embeddings are fused, they are fed into the Bi-SRU (Lei et al., 2018). Unidirectional simple recurrent unit (SRU) cannot effectively capture bidirectional semantic information, so the forward and reverse SRUs are integrated to form a bidirectional SRU structure. At the same time, a skip connection is used to optimize the gradient propagation and exclude the gradient disappearance owing to the long propagation distance when increasing the network depth. As shown in Figure 2, in the SRU, the forget gate regulates the retention of essential information from the previous instant to the present. The reset gate is used to determine the state of the output. The formulas for SRU are shown in equations (1) through (5).

$$\tilde{x}_t = W x_t \tag{1}$$

$$f_t = \sigma(W_f x_t + b_f) \tag{2}$$

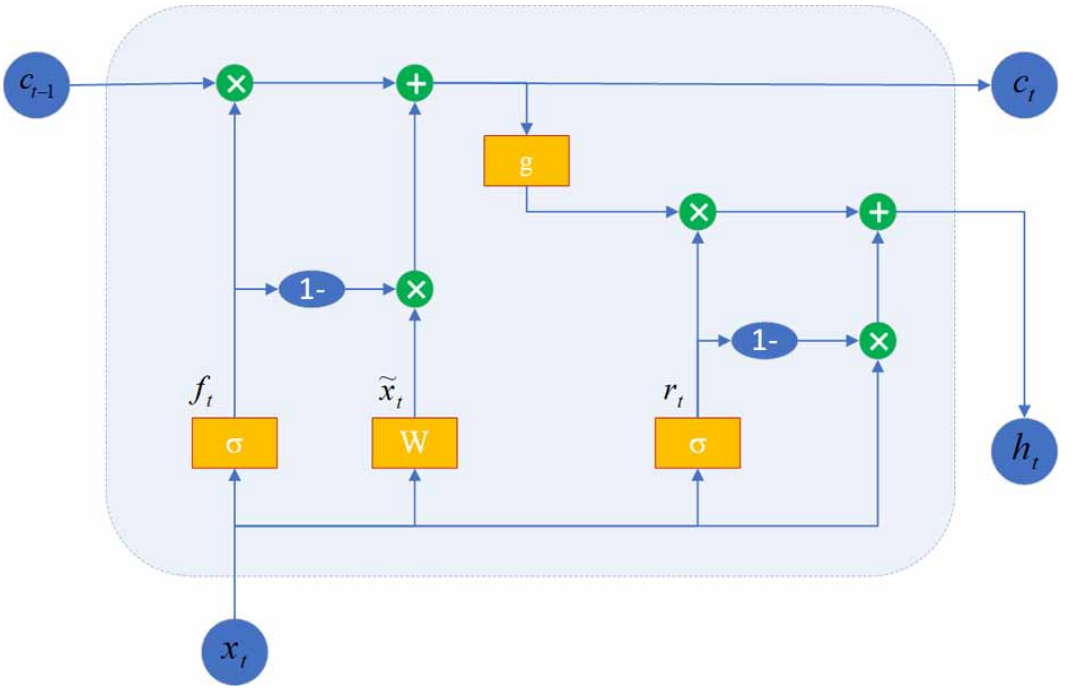
$$r_t = \sigma(W_r x_t + b_r) \tag{3}$$

$$c_t = f_t \odot c_{t-1} + (1 - f_t) \odot \tilde{x}_t \tag{4}$$

$$h_t = r_t \odot g(c_t) + (1 - r_t) \odot x_t \tag{5}$$

where x_t represents the input and t the time step, W and W_f are the parameter weights, and σ and g are the activation functions. f_t and b_f are the output and bias term of the forget gate in equation (2). r_t and b_r are the output and bias terms of the reset gate in equation (3). c_t is the state vector in equation (4), and the value is determined by f_t adaptively averaging the previous state c_{t-1}, and the present moment $\tilde{x}_t$. h_t in equation (5) is the hidden state vector at time t, computed through skip connections. An activation function g is used to activate hidden states.

Figure 2. Structure of SRU cell



The feature sequences obtained after passing through Bi-SRU are $\vec{H} = (h_1, h_2, ...h_t)$ and $\overleftarrow{H} = (h_1, h_2, ...h_t)$, and the feature vectors are their connected vectors. The resulting sentence representations A and B are input into Bi-SRU, and the formulas are shown in equations (6) and (7),

$$h_i^A = BiSRU(h_{i-1}^A, A_i) \qquad i = 1, ..., m \tag{6}$$

$$h_j^B = Bi\text{SRU}(h_{j-1}^B, B_j) \qquad j = 1, ..., n \tag{7}$$

where h_i^A is the *i*th encoding vector of A after encoding, and h_j^B is the *j*th encoding vector of B after encoding. A_i represents the ith word vector of A. B_j represents the jth word vector of B. The word vector A_i represents the ith word in A, and B_j is also applicable.

Local Similarity Modeling

Soft-Alignment Attention

We use soft-alignment attention (Chen et al. 2017; Bahdanau et al., 2016) to compute the similarity of hidden state groups $< h_i^A, h_j^B >$ between sentence pairs to associate related parts between two sentences. The formula is shown in equation (8),

$$s_{ij} = (h_i^A)^T h_j^B \tag{8}$$

where s_{ij} is the attention weight, which determines the local similarity of sentence pairs.

For an already encoded element in one of the sentences; i.e., h_i^A, s_{ij} is used to determine its semantic similarity information in another sentence in equation (9).

$$\tilde{h}_i^A = \sum_{j=1}^{n} \frac{\exp\left(s_{ij}\right)}{\sum_{k=1}^{n} \exp\left(s_{ik}\right)} h_j^B, \forall i \in \left[1, \ldots, m\right] \tag{9}$$

$$\tilde{h}_j^B = \sum_{i=1}^{m} \frac{\exp\left(s_{ij}\right)}{\sum_{k=1}^{m} \exp\left(s_{kj}\right)} h_i^A, \forall i \in \left[1, \ldots, n\right] \tag{10}$$

In these equations, $\tilde{h}_i^A$ is the weighted sum of $\{h_j^B\}_{j=1}^m$. Intuitively, the information in $\{h_j^B\}_{j=1}^m$ related to h_i^A is selected and denoted as $\tilde{h}_i^A$. The same calculation operation is used for $\tilde{h}_j^B$ in equation (10).

Sentence Interaction Modeling

By computing the difference and element-wise product of tuples $< h_i^A, \tilde{h}_i^A >$ and $< h_j^B, \tilde{h}_j^B >$, we augment the information collected further. This can help capture semantic interaction information between elements in tuples. The original vectors h_i^A and $\tilde{h}_i^A$, h_j^B and $\tilde{h}_j^B$ are then connected to the difference and the product of the elements (Mou et al., 2016), respectively. The calculation process is shown in equations (11) and (12).

$$I^A = [h_i^A; \tilde{h}_i^A; h_i^A - \tilde{h}_i^A; h_i^A \odot \tilde{h}_i^A] \tag{11}$$

$$I^B = [h_j^B; \tilde{h}_j^B; h_j^B - \tilde{h}_j^B; h_j^B \odot \tilde{h}_j^B] \tag{12}$$

In these equations, $\odot$ stands for element-wise multiplication.

Global Similarity Modeling

The global similarity modeling consists of an enhanced composition layer and a pooling layer.

Enhanced Composition Layer

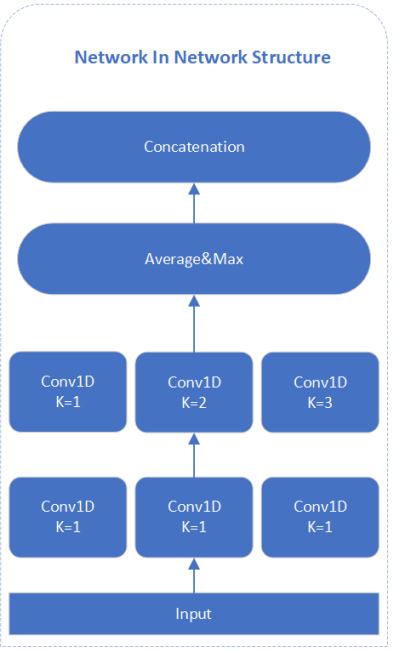
This part is used to composite local similarity information. Here, it is different from using two Bi-LSTMs in ESIM. We use a two-layer Bi-SRU to combine local similarity information and capture more fine-grained global similarity information from a temporal perspective and a semantic one, respectively. Simultaneously, we employ CNN to obtain global similarity information from a spatial perspective. SRU has better scalability and parallelism and has an excellent effect on balancing model capacity and performance, whereas CNN performs better in capturing key similarity information. By combining the two, more fine-grained global similarity information from multiple perspectives can be captured in the combined process. We use the mapping G in equations (13) and (14) to prevent overfitting owing to the increase in the global parameters caused by the combination of local information. G is a feed-forward neural network with activation function rectified linear unit (ReLu), and l_t represents the output of Bi-SRU at time t.

$$l_t^A = BiSRU(G(I_t^A), t) \tag{13}$$

$$l_t^B = BiSRU(G(I_t^B), t) \tag{14}$$

After fully capturing the global similarity information in both the temporal and semantic perspectives, we use CNN layers to further capture the global similarity information in the spatial perspective. At the same time, we use the "NIN" idea (Lin et al., 2014; He et al., 2015; Zhang et al., 2021); namely, "network in network," to improve CNN's performance. Its internal structure is shown in Figure 3. The first layer of CNN contains three convolution kernels with the same size of 1×1, which are used to extract the overall spatial information. It can increase the degree of nonlinearity through the subsequent nonlinear activation function while reducing the number of output feature maps. The second layer of CNN contains three different convolution kernels; namely, 1×1, 2×2 and 3×3, to further capture the key abstract features of spatial information (Peng et al., 2020).

Figure 3. Structure of the NIN layer



We express the result A output in the previous step as shown in the formula in equation (15).

$$l_{1:m} = l_1 \oplus l_2 \oplus l_3 \oplus \ldots \oplus l_m \tag{15}$$

In this equation k_0 is the hidden state size and $\oplus$ is the connection operation. $l_{i:i+j}$ is the connection of $(l_i, l_{i+1}, \ldots l_{i+j})$, which is input into the improved CNN, i.e., NIN. For each convolution operation, steps are taken to create a new feature $\overline{l}_i$ through a window of word $l_{i:i+w-1}$. These steps are shown in equations (16) and (17).

$$\overline{l}_i = \sigma(W \cdot l_{i:i+w-1} + b) \tag{16}$$

In equation (16) $W \in R^{w \cdot k_0}$, and W is the filter. The convolution acts on units of w words to capture features, and σ is the activation function. ReLu (Nair, & Hinton, 2010) is used here, and b is the bias factor. This filter acts on each possible window in $\{l_{1:w}, l_{2:w+1}, \ldots, l_{m-w+1:m}\}$ to generate feature maps as shown in equation (17).

$$\overline{l} = [l_1, l_2, \ldots, l_{m-w+1}] \tag{17}$$

In this equation, $\overline{l} \in R^{k_1}$ and $k_1 = m - w + 1$.

On top of the output of the convolution, max pooling and column-wise average pooling are performed to capture the most valuable features for each feature map, and these vectors are then combined. The whole formula is shown in equation 18.

$$\tilde{l} = NIN(l) \tag{18}$$

Pooling Layer

We process the result after passing through Bi-SRU as a fixed length. Because the summation operation is greatly affected by the length of the sequence (Parikh et al., 2016), we combine the two pooling operations and concatenate the obtained results with the results of the NIN layer to create the final vector of fixed length o . The operation is shown in equations (19), (20), and (21).

$$l_{ave}^{A} = \sum_{i=1}^{m} \frac{l_i^A}{m}, \qquad l_{\max}^{A} = \max_{i=1}^{m} l_i^A \tag{19}$$

$$l_{ave}^{B} = \sum_{j=1}^{n} \frac{l_j^B}{n}, \qquad l_{\max}^{B} = \max_{j=1}^{n} l_j^B \tag{20}$$

$$o = [l_{ave}^{A}; l_{\max}^{A}; \tilde{l}^{A}; l_{ave}^{B}; l_{\max}^{B}; \tilde{l}^{B}] \tag{21}$$

Then we feed o into the MLP classifier, as shown in equation (22).

$$y = MLP(o) \tag{22}$$

In this equation, MLP is a hidden layer with tanh activation and an output layer with softmax.

Loss Function and Metrics

Loss Function

We adopted the cross-entropy loss function in our experiments. The network parameters for an N-capacity data set are calculated where there is a cross-entropy minimum between the predicted label and the actual label, as shown in equation (23).

$$L(\hat{y}, y) = -\frac{1}{N} \sum_{x} [y \log \hat{y} + (1 - y) \log(1 - \hat{y})] \tag{23}$$

In this equation, $\hat{y}$ represents the predicted value, y represents the real label, and backpropagation is used for network training.

METRICS

Model performance is evaluated using accuracy and F1-score in this paper. In our network, accuracy refers to the capacity of the model to accurately distinguish similar and different texts. The precision rate indicates how accurately the model identifies the correct sentence among comparable sentences. Recall rate is the performance of the model in all similar sentences. F1-score reflects the average performance of the model on precision rate and recall rate.

Experiments

On two public Chinese semantic similarity datasets, we conducted numerous tests to assess the efficacy of SMAEN.

Dataset

LCQMC Dataset

LCQMC (Liu et al., 2018) is a large-scale Chinese question semantic matching dataset constructed by Harbin Institute of Technology in 2018. It comes from real questions from users in different fields of Baidu QA. A focus is placed on the intent of the question when matching. The dataset contains 260,086 instances of labeled questions, including similar questions and dissimilar questions. There are 238,766 pairs of questions in the training set, 8,802 pairs in the validation set, and 12,500 pairs in the test set. The examples are presented in Table 1. In the case of semantically similar questions, it is marked as 1, and for the semantically dissimilar ones it is marked as 0.

Table 1. Examples from LCQMC dataset

Sentence A	Sentence B	Label
如何清洗手表的皮表带?	手表表带脏了怎么清洗?	1
(How to clean the leather strap of a watch?)	(How to clean the dirty watch strap?)	
哪种修脚刀锋利?	扬州三把刀哪家好?	0
(Which pedicure knife is sharp?)	(Which of the three knives in Yangzhou is better?)	

PAWS-X (Chinese) Dataset

The PAWS-X (Yang et al., 2019) (Chinese) dataset is a synonym judgment dataset released by Google. The Chinese part includes paraphrased pairs and non-paraphrased pairs. The main feature is that it has highly overlapping vocabulary and pays more attention to the judgment ability of the model for syntax. The dataset contains 53,401 pairs of label sentences. There are 49,401 pairs of instances in the training set, and 2,000 pairs in each verification set and test set. Table 2 shows examples of the PAWS-X (Chinese) dataset. In the case of semantically similar questions, it is marked as 1, and when the semantics are not similar it is marked as 0.

Implementation

Both character embedding and word embedding had dimensions of 300.Word embedding used word2vec vectors trained on the Wikipedia_zh corpus, whereas character embedding used word vectors trained on Baidu Baike. For uniformity, we padded sentences with fewer than 50 words to ensure that the input sequence had a maximum length of 50 words. The Bi-SRU layer's hidden state was set to 300. We adopted two methods to avoid overfitting: the dropout value was to 0.1, and early stopping

Table 2. Examples from PAWS-X (Chinese) dataset

ID	Sentence A	Sentence B	Label
1	安装有一个三脚架, 但前腿是一个脚轮。	安装是三脚架, 但前腿有一个方向盘。	0
	(There is a tripod mounted, but the front legs are a caster.)	(The mount is a tripod, but the front legs have a steering wheel.)	
2	CUTA有五个国家和地区委员会。	CUTA有五个国家委员会和五个地区委员会。	1
	(CUTA has five national and regional committees.)	(CUTA has five national committees and five regional committees.)	

was used. We used the dev set for evaluating the training loss in this experiment, and the tolerance set was set to five. There would be an early termination of the training process if the accuracy rate on the five dev sets did not improve. Using the Adam optimizer, we set the learning rate as 0.0002. Training for 50 epochs with batch size was set to 256.

Baseline Methods

To demonstrate that SMAEN is effective in the Chinese STS task, we compared and evaluated the model using these baseline approaches:

- Decomposable attention (Parikh et al., 2016): This approach uses soft alignment to decompose the whole task into independent subtasks, instead of the work of LSTM encoding two sentences; this approach integrates and classifies the results of the subproblems.
- Attention-based convolutional neural network (ABCNN): This approach uses three attention-based CNNs to encode two sentences, capturing finer-grained interaction features; it also uses logistic regression for similarity measurement.
- BiMPM: This approach encodes sentence pairs by using Bi-LSTM; it performs multiview matching in two different directions using four methods and uses fully connected layers for classification.
- ESIM: This approach uses attentional soft alignment to collect deep interaction information between sentence pairs to measure similarity.
- RE2 (Yang et al., 2019): This approach uses the residual network for information enhancement; it captures the information using a variety of alignment methods and predicts the target.
- ERCNN: This approach uses CNN and RNN to capture finer-grained interactions between sentence pairs and a special fusion layer to model overall similarity.

Results and Discussion

The results of SMAEN were compared with those of other baseline models that achieved good performance on both Chinese and English datasets. After we reproduced these models, we applied them to the dataset selected in this experiment. We optimized models on the dev set before evaluating them on the test set in all experiments.

LCQMC Dataset Result

Table 3 displays the experimental findings about LCQMC. To evaluate various models, we concentrated on the model's accuracy and F1 score. The decomposable attention model relies solely on soft alignment attention and cannot account for the temporal and spatial characteristics of the sentence context, so the accuracy and F1 score are inferior to those of competing models. The ABCNN model uses CNN to capture sentence spatial features and lacks the capture of sentence temporal features.

Table 3. Experimental results on LCQMC dataset

Model	Accuracy (%)	F1 (%)
Decomposable Attention	80.50	82.97
ABCNN	81.03	82.90
BiMPM	82.08	84.61
ESIM	84.32	85.66
RE2	84.14	85.49
ERCNN	77.64	80.66
SMAEN	85.88	86.62

The BIMPM model uses different perspectives and multiple angles for matching; it also uses Bi-LSTM for sentence encoding. The accuracy and F1 score are 1.05% and 1.71% higher than ABCNN, respectively. ESIM uses Bi-LSTM to build the network and then uses the attention mechanism for information interaction and overall similarity modeling. Although it also obtains good results, it takes only word vectors as input, which is somewhat thin in the semantic richness of sentences. The overall similarity modeling uses Bi-LSTM, which has great limitations in model scalability and feature richness. RE2 uses a residual network and uses CNN to encode and interact with the attention mechanism in encoding, but the effect on this dataset is not so good as ESIM. The performance of ERCNN in this dataset is poor, which may be due to the low fitting degree of ERCNN on this dataset. The captured interaction information also does not play a significant role in text matching. Our model integrates embeddings at two levels of granularity and captures and fuses global similarity information at multiple angles to calculate sentence pairs' similarity. Compared with ESIM, the accuracy rate has an increase of 1.74%, and the F1 score has an increase of 0.96%.

According to Table 3, the proposed SMAEN model has superior accuracy and F1 score compared with all baseline models. Figure 4 depicts the training process for the LCQMC model.

The setting of the dropout parameter has a great influence on the model, so we explored it on the dataset LCQMC. In the process of training each batch, dropout will discard some neurons with a certain probability; this action can significantly reduce overfitting. As shown in Figure 5, when the

Figure 4. Specific training process on LCQMC dataset. (a) Training set loss value (b) Training set accuracy

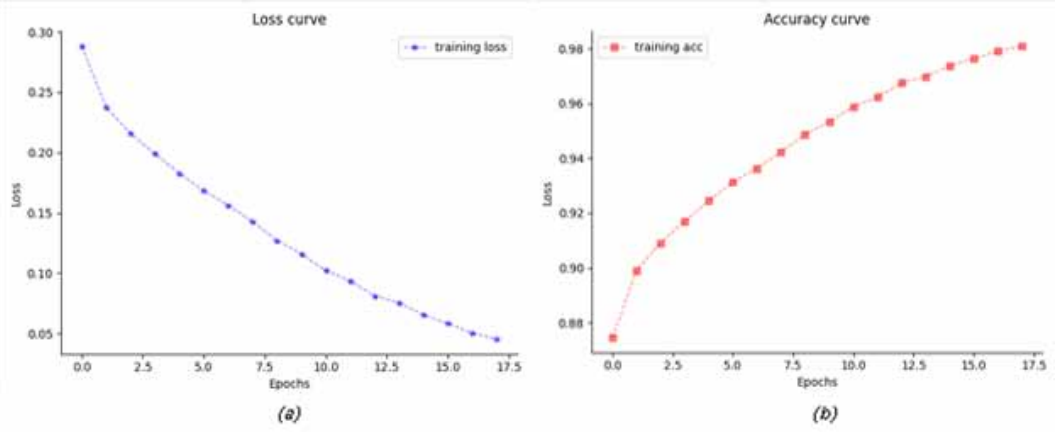
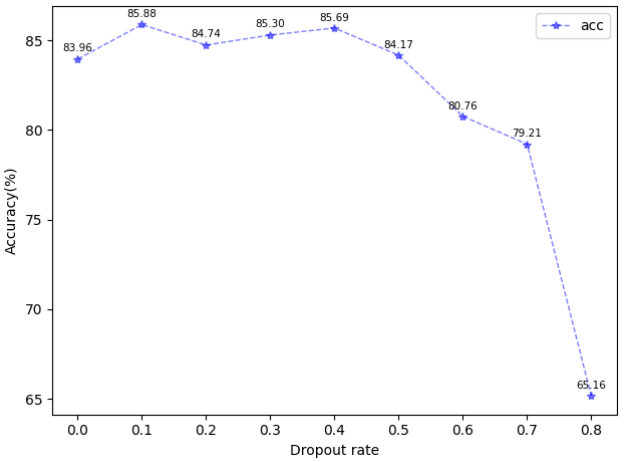


Figure 5. Dropout value and accuracy rate changes on LCQMC



dropout is 0, the model accuracy is the lowest, and when it is greater than 0.1, the model accuracy gradually decreases, so we set the dropout to 0.1 to achieve the best performance.

PAWS-X (Chinese) Dataset Result

Table 4 shows the results of the experiment on the PAWS-X (Chinese) dataset. Compared with other models, the accuracy and F1 score of SMAEN are superior than those of other baseline models. Compared with the latest technology, our model has an increase of 3.11% and 4.8% in accuracy and F1 score, respectively.

Figure 6 show the results of the model stopping training at the 18th epoch.

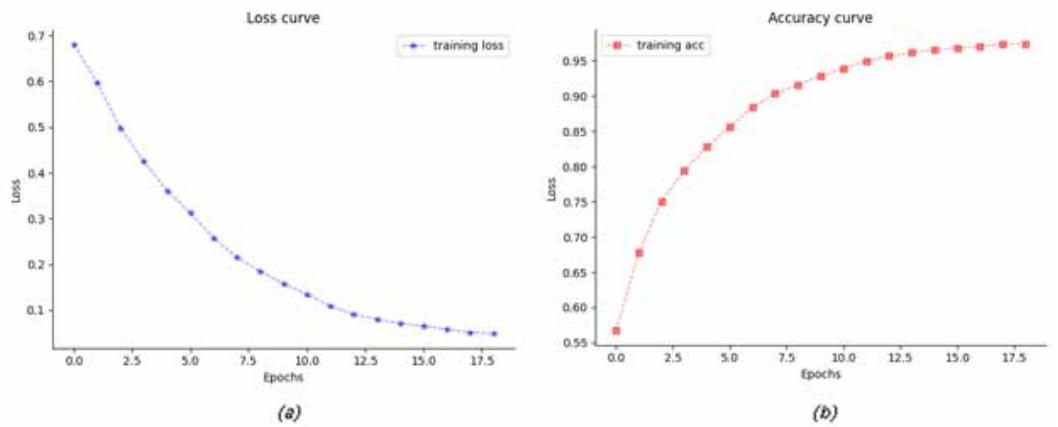
Classic Case Analysis

Based on the analysis of the experimental results of SMAEN and other models on the LCQMC and PAWS-X (Chinese) datasets, we compared the best-performing ESIM with the prediction results of SMAEN in specific cases. Table 5 shows that SMAEN is more accurate than ESIM in calculating text similarity.

Table 4. Experimental results on PAWS-X (Chinese) dataset

Model	Accuracy (%)	F1 (%)
Decomposable Attention	55.72	10.51
ABCNN	55.27	17.81
BiMPM	57.17	60.53
ESIM	67.95	62.62
RE2	56.35	20.85
ERCNN	63.82	48.36
SMAEN	71.06	67.42

Figure 6. Specific training process on PAWS-X (Chinese) dataset. (a) Training set loss value (b) Training Set Accuracy



In the first set of cases, both SMAEN and ESIM made correct predictions for short and semantically similar sentences. It is proved that both of them are more accurate in calculating the similarity of two sentences with similar semantics.

In the two sentences in the second set of cases, although the characters and words are highly similar, they are semantically quite different. ESIM did not capture the key difference features, but only captured sentence features from a single perspective, making a wrong judgment. SMAEN fully captures the key differences and similar features of two sentences from three perspectives, improving its judgment performance.

The two sentences in the third group of cases still have a high degree of similarity in terms of words, but because of the ambiguity and abstraction of Chinese itself, this group of sentences is more difficult to judge than the first two groups. In the process of global similarity modeling, ESIM used

Table 5. Classic case forecast comparison

No	Sentence A	Sentence B	Label	SMAEN	ESIM
(1)	刻舟求剑这则寓言告诉我们什么?	刻舟求剑告诉我们什么道理?	1	1	1
	(What does the fable of carving a boat and seeking a sword tell us?)	(What does Carving a Boat and Seeking a Sword tell us?)			
(2)	它于 2011 年 12 月 22 日出版并于 2012 年 2 月公布。	它在 2011 年 12 月 22 日公布并在 2012 年 2 月发行。	0	0	1
	(It was published on December 22, 2011, and announced in February 2012.)	(It was announced on December 22, 2011, and released in February 2012.)			
(3)	他受邀于 1924 年在多伦多担任 ICM 发言人,1932 年在苏黎世和 1936 年在奥斯陆也相继担任发言人。	1924 年、1932 年和 1936 年,他分别在多伦多、奥斯陆和苏黎世担任 ICM 的特邀发言人。	0	0	1
	(He was invited to be ICM speaker in Toronto in 1924. He was also a speaker in Zurich in 1932 and in Oslo in 1936.)	(In 1924, 1932 and 1936 he was a guest speaker at the ICM in Toronto, Oslo, and Zurich, respectively.)			

Table 6. Ablation study on LCQMC dataset

Ablation Model	Accuracy (%)	F1 (%)
Base model	85.88	86.62
-NIN	84.76	86.02
-modeling Bi-SRU	84.58	85.76
-encoding Bi-SRU	80.43	83.09
-All Bi-SRU	81.27	83.27
-avg pool	83.54	85.20
-max pool	83.65	85.28
-diff/prod	81.31	83.59
-char embedding	85.33	86.26

Bi-LSTM to match and aggregate only local information, and did not make a better global analysis on abstract text, which leads to the judgment of similar sentences. However, SMAEN used not only multi-granularity embedding when extracting the features of the sentence itself, but also CNN combined with two-layer Bi-SRU to conduct a more comprehensive analysis in global similarity modeling, which improved the model's understanding of abstract features and ambiguous features.

Ablation Study

To determine the effectiveness of each component of SMAEN, we performed ablation experiments on the LCQMC dataset. Table 6 presents the results.

On the LCQMC dataset, we first removed the NIN module, and the accuracy dropped to 84.76% and the F1 score to 86.02%. The performance also dropped significantly, which shows that the NIN module plays a good role in global similarity modeling. We then removed the two-layer Bi-SRU in the global similarity modeling, and the accuracy dropped to 84.58%, and the F1-score dropped to 85.76%. This indicates that the two-layer Bi-SRU can better capture the feature information of inter-sentence interactions in global similarity modeling. The Bi-SRU accuracy and F1 score after removing the coding layer dropped to 80.43% and 83.09%, respectively, which indicates that Bi-SRU has a strong coding function.

When we removed all Bi-SRU modules in the model, the accuracy rate dropped to 81.27%, and the F1 score to 83.27%, indicating the scalability of Bi-SRU. When the average pooling was removed, the accuracy rate dropped to 83.54% and the F1-score dropped to 85.20%. When the maximum pooling was removed, the accuracy rate dropped to 83.65% and the F1 score dropped to 85.28%. This indicates that the average pooling has a greater impact on the overall model than the maximum pooling. If the difference and element-wise product were removed, the accuracy fell to 81.31% and the F1 score dropped to 83.59%. In addition, we explored the contribution of character embedding to the model. If the character embedding was removed, the accuracy would decrease to 85.33%, while the F1 score would fall to 86.26%. This shows that character embedding can effectively alleviate the out-of-vocabulary (OOV) problem, and more fine-grained features at the character granularity can be extracted with effectiveness.

Experimental results indicate that the SMAEN model outperforms all baseline models on both datasets, clearly indicating that the model is accurate and generalizable.

CONCLUSION

In this paper, we propose a new model named SMAEN to solve the problem of insufficient sentence interaction information and the complexity of Chinese semantic features in semantic text similarity calculation. SMAEN shows excellent performance on the Chinese public datasets LCQMC and PAWS-X using the baseline method, with an accuracy increase by 1.56% and 3.11%, respectively. In our approach, character embedding and word embedding are fused, and the use of two granular embeddings helps to capture finer-grained semantic feature information in Chinese text.

In addition, an easily extensible Bi-SRU is used instead of Bi-LSTM to extract sentence syntax and semantic information to reduce the computational load of the model. To reduce the loss of partial global feature information caused by global similarity modeling of sentence interactions from a single perspective, we model the global similarity of sentence interactions from three perspectives of semantics, time, and space to capture richer global similarity information. It provides a solid foundation for big data application research. The proposed method can produce more precise matching results and support the extraction of crucial data from massive data.

Future research will investigate the robustness and generalizability of the proposed language model on various tasks involving NLP. Simultaneously, we intend to integrate graph convolutional networks and knowledge graphs into the language model to enrich semantic interaction data.

FUNDING STATEMENT

This work was supported by Zhejiang Province Key Research and Development plan program of China (No. 2022C01207).

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Bahdanau, D., Cho, K., & Bengio, Y. (2016). *Neural machine translation by jointly learning to align and translate.* arXiv.org. 10.48550/arXiv.1409.0473

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1993). Signature verification using a "Siamese" time delay neural network. In J. Cowan, G. Tesauro, & J. Alspector (Eds.), Advances in neural information processing systems: Vol. 6. Morgan-Kaufmann. doi:10.1142/S0218001493000339

Chen, G., Shi, X., Chen, M., & Zhou, L. (2020). Text similarity semantic calculation based on deep reinforcement learning. *International Journal of Security and Networks*, *15*(1), 59–66. doi:10.1504/IJSN.2020.106526

Chen, H., Han, F. X., Niu, D., Liu, D., Lai, K., Wu, C., & Xu, Y. (2018). MIX: Multi-Channel information crossing for text matching. In *KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery. doi:10.1145/3219819.3219928

Chen, Q., Sun, J., & Zhao, Y. (2021). Novel architecture with separate comparison and interaction modules for Chinese semantic sentence matching. *Neural Processing Letters*, *53*(5), 3677–3692. doi:10.1007/s11063-021-10561-3

Chen, Q., Zhu, X., Ling, Z., Wei, S., Jiang, H., & Inkpen, D. (2017). Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (*Volume 1*: Long Papers)*. Association for Computational Linguistics. doi:10.18653/v1/P17-1152

Chen, S., & Xu, T. (2021). Long text QA matching model based on BiGRU-DAttention-DSSM. *Mathematics*, *9*(10), 1129. doi:10.3390/math9101129

Chen, Z., Cheng, X., Dong, S., Dou, Z., Guo, J., Huang, X., Lan, Y., Li, C., Li, R., Liu, T.-Y., Liu, Y., Ma, J., Qin, B., Wang, M., Wen, J., Xu, J., Zhang, M., Zhang, P., & Zhang, Q. (2021). Information retrieval: A View from the Chinese IR community. *Frontiers of Computer Science*, *15*(1), 151601. doi:10.1007/s11704-020-9159-0

Cheng, H., Shen, Y., Liu, X., He, P., Chen, W., & Gao, J. (2021). *UnitedQA: A hybrid approach for open domain question answering*. arXiv.org. 10.48550/arXiv.2101.00178

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. arXiv.org. 10.48550/arXiv.1810.04805

Ghasemi, N., & Momtazi, S. (2021). Neural text similarity of user reviews for improving collaborative filtering recommender systems. *Electronic Commerce Research and Applications*, *45*, 101019. doi:10.1016/j.elerap.2020.101019

Gong, J., Hu, X., Song, W., Fu, R., Sheng, Z., Zhu, B., Wang, S., & Liu, T. (2021). IFlyEA: A Chinese essay assessment system with automated rating, review generation, and recommendation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*. Association for Computational Linguistics. doi:10.18653/v1/2021.acl-demo.29

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition*. arXiv.org. 10.48550/arXiv.1512.03385

Hu, B., Lu, Z., Li, H., & Chen, Q. (2014). Convolutional neural network architectures for matching natural language sentences. In Advances in Neural Information Processing Systems, 27 (NIPS 2014). Curran Associates, Inc.

Hu, J., Hayashi, H., Cho, K., & Neubig, G. (2021). *DEEP: DEnoising entity pre-training for neural machine translation*. arXiv.org. 10.48550/arXiv.2111.07393

Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., & Heck, L. (2013). Learning deep structured semantic models for web search using clickthrough data. In *CIKM '13: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*. Association for Computing Machinery. doi:10.1145/2505515.2505665

Kim, S., Kang, I., & Kwak, N. (2019). Semantic sentence matching with densely-connected recurrent and co-attentive information. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(1), 6586-6593. doi:10.1609/aaai.v33i01.33016586

Lei, T., Zhang, Y., Wang, S. I., Dai, H., & Artzi, Y. (2018). *Simple recurrent units for highly parallelizable recurrence*. arXiv.org. 10.48550/arXiv.1709.02755

Lin, M., Chen, Q., & Yan, S. (2014). *Network in network*. arXiv.org. 10.48550/arXiv.1312.4400

Lin, Z., Feng, M., Santos, C. N. dos, Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). *A structured self-attentive sentence embedding*. arXiv.org. 10.48550/arXiv.1703.03130

Liu, H. (2022). Financial risk intelligent early warning system of a municipal company based on genetic tabu algorithm and big data analysis. *International Journal of Information Technologies and Systems Approach*, *15*(3), 1–14. doi:10.4018/IJITSA.307027

Liu, X., Chen, Q., Deng, C., Zeng, H., Chen, J., Li, D., & Tang, B. (2018). LCQMC: A large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A robustly optimized BERT pretraining approach*. arXiv.org. 10.48550/arXiv.1907.11692

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. arXiv.org. 10.48550/arXiv.1301.3781

Mistree, K., Thakor, D., & Bhatt, B. (2022). A machine translation system from Indian sign language to English text. *International Journal of Information Technologies and Systems Approach*, *15*(1), 1–23. doi:10.4018/IJITSA.313419

Mohamed, A., Najafabadi, M. K., Wah, Y. B., Zaman, E. A. K., & Maskat, R. (2020). The state of the art and taxonomy of big data analytics: View from new big data framework. *Artificial Intelligence Review*, *53*(2), 989–1037. doi:10.1007/s10462-019-09685-9

Mou, L., Men, R., Li, G., Xu, Y., Zhang, L., Yan, R., & Jin, Z. (2016). *Natural language inference by tree-based convolution and heuristic matching*. arXiv.org. 10.48550/arXiv.1512.08422

Mueller, J., & Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. *Proceedings of the AAAI Conference on Artificial Intelligence*, *30*(1). doi:10.1609/aaai.v30i1.10350

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *ICML'10: Proceedings of the 27th International Conference on Machine Learning*. Association for Computing Machinery.

Niu, Y., Huang, F., Liang, J., Chen, W., Zhu, X., & Huang, M. (2021). *Semantic-based method for unsupervised commonsense question answering*. arXiv.org. 10.48550/arXiv.2105.14781

Parikh, A. P., Täckström, O., Das, D., & Uszkoreit, J. (2016). *A decomposable attention model for natural language inference*. arXiv.org. 10.48550/arXiv.1606.01933

Peng, S., Cui, H., Xie, N., Li, S., Zhang, J., & Li, X. (2020). Enhanced-RCNN: An efficient method for learning sentence similarity. In *WWW'20: Proceedings of the Web Conference*. Association for Computing Machinery. doi:10.1145/3366423.3379998

Song, M., Liu, Q., & Haihong, E. (2018). Deep hierarchical attention networks for text matching in information retrieval. *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*, 476-481. doi:10.1109/ICISCAE.2018.8666926

Tang, X., Luo, Y., Xiong, D., Yang, J., Li, R., & Peng, D. (2022). Short text matching model with multiway semantic interaction based on multi-granularity semantic embedding. *Applied Intelligence*, *52*(13), 15632–15642. doi:10.1007/s10489-022-03410-w

Wang, S., & Jiang, J. (2016). *A compare-aggregate model for matching text sequences*. arXiv.org. 10.48550/arXiv.1611.01747

Wang, Z., Hamza, W., & Florian, R. (2017). *Bilateral multi-perspective matching for natural language sentences*. arXiv.org. 10.48550/arXiv.1702.03814

Wu, C., Yan, B., Yu, R., Huang, Z., Yu, B., Yu, Y., Chen, N., & Zhou, X. (2021). Improvement of K-means algorithm for accelerated big data clustering. *International Journal of Information Technologies and Systems Approach*, *14*(2), 99–119. doi:10.4018/IJITSA.2021070107

Xu, S. E. S., & Xiang, Y. (2020). Enhanced attentive convolutional neural networks for sentence pair modeling. *Expert Systems with Applications*, *151*, 113384. doi:10.1016/j.eswa.2020.113384

Yang, R., Zhang, J., Gao, X., Ji, F., & Chen, H. (2019). *Simple and effective text matching with richer alignment features.* arXiv.org. 10.48550/arXiv.1908.00300

Yang, Y., Zhang, Y., Tar, C., & Baldridge, J. (2019). *PAWS-X: A cross-lingual adversarial dataset for paraphrase identification*. arXiv.org. 10.48550/arXiv.1908.11828

Yin, W., & Schütze, H. (2015). MultiGranCNN: An architecture for general matching of text chunks on multiple levels of granularity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (*Volume 1*: Long Papers)* (pp. 63-73). Association for Computational Linguistics. doi:10.3115/v1/P15-1007

Yin, W., Schütze, H., Xiang, B., & Zhou, B. (2016). ABCNN: Attention-based convolutional neural network for modeling sentence pairs. *Transactions of the Association for Computational Linguistics*, *4*, 259–272. doi:10.1162/tacl_a_00097

Yu, C., Xue, H., Jiang, Y., An, L., & Li, G. (2021). A simple and efficient text matching model based on deep interaction. *Information Processing & Management*, *58*(6), 102738. doi:10.1016/j.ipm.2021.102738

Yuan, Z., & Jun, S. (2020). Siamese network cooperating with multi-head attention for semantic sentence matching. In *2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)* (pp. 215-218). IEEE. doi:10.1109/DCABES50732.2020.00068

Zhang, H., Zhang, H., Lu, X., & Gao, Q. (2021). Attention-based overall enhance network for Chinese semantic textual similarity measure. *Journal of Applied Science and Engineering*, *25*(2), 287–295. doi:10.6180/jase.202204_25(2).0005

Zhang, X., Lei, F., & Yu, S. (2021). Self-attention based text matching model with generative pre-training. In *2021 IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC), International Conference on Pervasive Intelligence and Computing (PiCom), International Conference on Cloud and Big Data Computing (CBDCom), International Conference on Cyber Science and Technology Congress (CyberSciTech)* (pp 84-91). IEEE. doi:10.1109/DASC-PICom-CBDCom-CyberSciTech52372.2021.00027

Zheng, L., Wu, Z., & Kong, L. (2021). *Cascaded head-colliding attention*. arXiv.org. 10.48550/arXiv.2105.14850

Jing Huang, professor, doctor, graduated from Huazhong University of Science and Technology in 2004. Worked in Zhejiang Sci-Tech University. Her research interests include new generation information technology (artificial intelligence, cloud computing, big data).

Keyu Ma, Master of Zhejiang Sci-Tech University. Her interests include natural language processing and big data.