IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 4, 2018, accepted November 18, 2018, date of publication November 27, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2883637

Multi-Scale Attentive Interaction Networks for Chinese Medical Question Answer Selection

SHENG ZHANG, XIN ZHANG, HUI WANG, LIXIANG GUO, AND SHANSHAN LIU
Science and Technology on Information Systems Engineering Laboratory, College of Systems Engineering, National University of Defense Technology, Changsha 410073, China

Corresponding author: Hui Wang (huiwang@nudt.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 71331008 and Grant 61105124.

ABSTRACT The past few years have witnessed a trend that the deep learning techniques have been increasingly applied in healthcare due to the explosive growth of big data. The online medical community, where users can ask qualified doctors about medical questions with just a few keystrokes and mouse clicks anytime and anywhere, has become quite popular recently. In this paper, we investigate the problem of Chinese medical question answer selection, which is a crucial subtask of automatic question answering and fairly challenging because of its language and domain characteristics. We introduce an end-to-end multi-scale interactive networks framework to address the issue. The framework consists of several multi-scale deep neural layers which extract the deep semantic information of medical text from different levels of granularity, shortcut connections which prevent network degradation problem, and attentive interaction which mines the correlation between questions and answers. To evaluate our framework, we update and expand a dataset called cMedQA v2.0. Experimental results demonstrate that our model outperforms the existing state-of-the-art models with noticeable margins.

INDEX TERMS Medical question answering, interactive attention, deep learning, deep neural networks.

I. INTRODUCTION

Deep learning methods are increasingly used in quantities of tasks in healthcare, such as medical image analysis [1], electronic health recording analysis [2], providing a new theoretical foundation for big data application. The online healthcare community can provide users with distant medical support, which on the one hand makes it convenient for the user, on the other hand contributes to accumulate large amounts of data. However, compared to the explosive growth of the number of questions, the number of doctors is rather limited. It is badly in need of the ability to automatically answer the question proposed by patients, based on the data collected before, which, as a consequence, greatly reduces the workload of doctors and increases the user experience of the online medical community.

This paper aims to study the topic of *Chinese medical question answer selection*, which is a key problem of automatic medical question answering and text understanding. Compared to question answer selection problems in English, problems in Chinese are far more challenging. Due to the lack of separator in Chinese text, word segmentation is a necessary preprocessing in many Chinese natural language processing tasks. Nevertheless, existing Chinese word segmentation toolkits generate errors when dealing with text of professional domain, resulting in cascade error and performance degradation in the subsequent models of the whole pipeline [3]. An effective way to mitigate the problem of word segmentation on medical text, according to the study conducted by Zhang *et al.* [3], is to leverage character-level embeddings rather than word-level embeddings.

Although Chinese characters have more semantic information than English letters, they still have less semantic information than Chinese words. It is very critical to mine the semantic information and correlation from questions and answers. Fig. 1 illustrates an example of Chinese question answer selection. For the question proposed by a user from one forum, the good answer is supplied by a qualified doctor, while the irrelevant answer is mistakenly chosen by a state-of-the-art model [4] from the answer candidate pool. It is obvious that expressions of questions and answers are different in a way: questions in the online forum are usually versatile and the expression is quite colloquial, in contrast the answers are much more professional and expressed in a standardized way. Therefore, the two major challenges lie in the ability

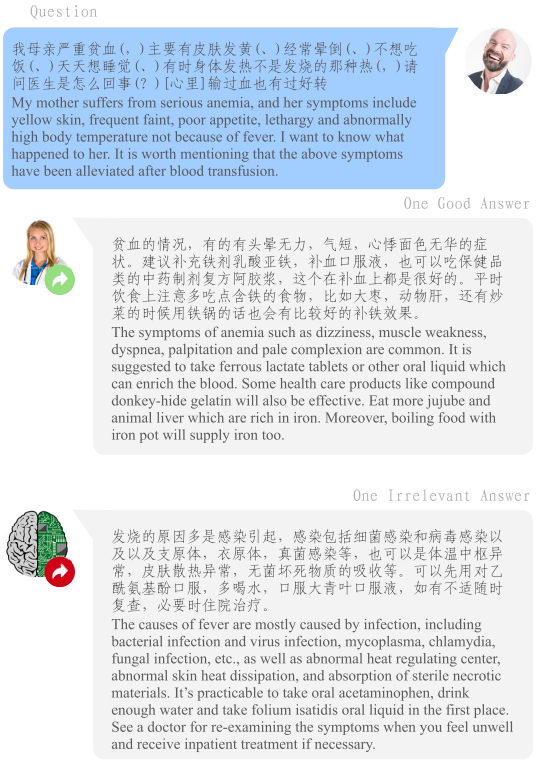


FIGURE 1. An example of Chinese question answer selection. The user wants to ask the question about his mother's anemia rather than fever. The irrelevant answer describes the cause of fever and countermeasures, which is apparently unsuitable for this question, while the good answer illustrates symptoms of anemia and provides useful advice for users.

to select the right answer from answer candidates: 1) How to sufficiently extract semantic information from Chinese text, to be specific, with character-level embeddings in this task. 2) How to effectively capture the correlative information between questions and answers.

To tackle the challenges mentioned above, a multi-scale interactive networks framework is introduced in this paper to capture the interactive information between questions and answers from different levels of granularity. The framework consists of many components including bidirectional gated recurrent units, multi-scale convolutional neural networks, shortcut connections and attentive interaction. Bidirectional gated recurrent units networks are able to extract global sequential information, while multi-scale convolutional neural networks have the capability to capture local information from characters, words and phrases. In addition, we introduce shortcut connections to the networks to enrich the information and avoid performance degradation. Moreover, the attentive interaction layer mines the correlation between questions and answers. The model addresses the challenges of the Chinese medical question answer selection from two main aspects: 1) The multi-layer neural networks and multi-scale convolutional neural networks sufficiently extract the information of character-level embeddings. 2) Attentive interaction networks effectively capture the correlative information between questions and answers.

The main contributions of this paper can be summarized as follows:

- We expand and update the cMedQA dataset, and release cMedQA v2.0, which has more samples and fine-tuned preprocessing.
- We propose a multi-scale attentive interaction networks framework, which is able to extract semantic information of different levels of granularity, and also interactive information between the question and answer.
- Our proposed model outperforms former state-of-the-art models, and reaches a higher top-1 accuracy score.
- Our proposed model is capable of giving visual interaction information, showing the connections between the question and answer.

The remainder of this paper is organized as follows: Section II briefly summarizes related work; Section III introduces our proposed framework in detail; Section IV presents the results of experiments on our proposed framework and other competitive models; Section V gives a discussion about the experimental results. Finally, we draw a conclusion and evaluate the future work in Section VI.

II. RELATED WORK

In the following we briefly describe two aspects of related work. The first one is recent studies on applying deep learning techniques to question answering of not only general domains but also the specific ones like the medical domain. The second one is the research on attention mechanism which helps deep learning model to extract attentive information.

A. DEEP LEARNING FOR QUESTION ANSWERING

Traditional methods such as matching-based, rule-based or statistics-based algorithms pay more attention to literal information, failing to extract the semantic information from the text [3]. Due to the powerful semantic feature extraction capabilities of neural networks, researchers have begun to study methods based on deep neural networks gradually. Kalchbrenner *et al.* [5] propose a method using convolutional neural networks (CNNs) to represent sentences in fixed-length one-dimensional vectors. Their method is wildly used in modeling sentences afterwards. Feng *et al.* [6] apply CNNs into feature extraction to generate vectors for questions and answers, and use cosine to measure the similarities. Qiu and Huang [7] also use multi-layer CNNs to generate vector representation for questions and answers, and the model in their paper uses k-max pooling to select features from CNNs. Tan *et al.* [8] apply long short-term memory networks (LSTMs) [9] instead of CNNs to obtain sequential information of questions and answers.

The work introduced above makes full use of the advantages of deep neural networks in extracting semantic information. For example, the convolutional neural network is capable of extracting local contextual information, while the recurrent neural network (RNN) and its variants are adequate to capture global sequential information form the text.

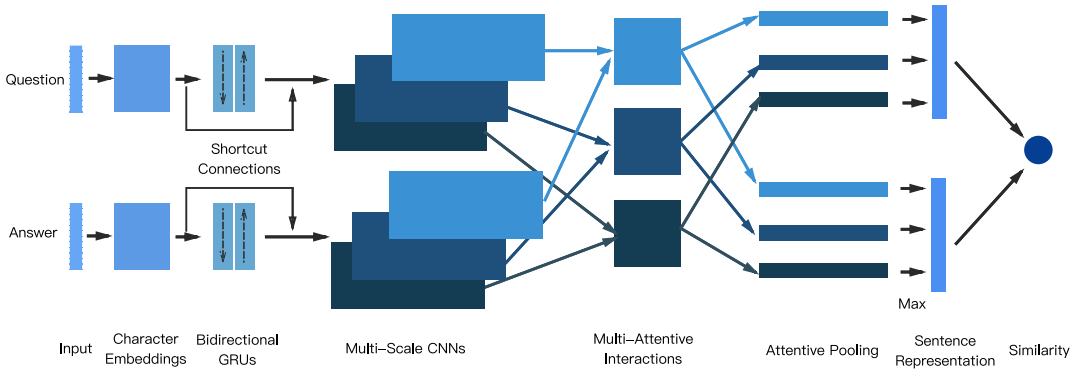


FIGURE 2. The architecture of end-to-end multi-scale interactive networks framework. The question and answer are cut into characters and encoded as character embeddings. Then these embeddings are fed into bidirectional gated recurrent units networks (GRUs) and multi-scale convolutional neural networks (CNNs). Shortcut connections are added between embedding layers and bidirectional GRUs. After that, the question and answer perform multi-attentive interaction with each other, and the results are used for attentive pooling to generate vectors. The framework uses max pooling to produce the final representation of the question and answer. Finally, the similarity is calculated by cosine. At each stage, components which have the same color share the identical parameters.

The methods using deep learning techniques have been demonstrated their superiority over some traditional methods such as rule-based models [10] and statistics-based models [11]. However, there is only a small amount of work which applies these techniques to question answer selection in some specific domains, especially in the Chinese medical domain.

Zhang *et al.* [3] adopt character-level embeddings to encode medical texts, and apply multi-scale CNNs to extract semantic information of different levels of granularity. They construct a Chinese community medical question answering dataset (cMedQA) and the framework proposed in their work defeats traditional statistical methods and single layer of CNN or RNN.

Ye *et al.* [4] propose a multi-level composite deep neural networks, which are not just stacking different types of neural networks. Instead, they combine features extracted from each layer of the framework to enrich the representation of final vectors. They reach the state-of-the-art performance on cMedQA dataset.

However, when extracting features from the text, the work mentioned above has not taken the interaction and connection between questions and answers into consideration until measuring their similarities at the last stage.

B. ATTENTION MODEL

As is known that questions usually have a lot of common information with answers, which is vital to capture the interactive information between them. Some parts of the sentence contain interactive information between the question and answer, which should be given more attention. In order to mine the interactive information, attention mechanism is generally used and attention-based approaches have been proven to have very promising performance on a large number of tasks, such as image caption generation [12], reading comprehension [13], [14], machine translation [15]–[17], sentence summarization [18].

Tan *et al.* [19] propose an attentive model which uses the representation vector of the question as memory vector to guide the pooling of the answer. Song *et al.* [20] and Wang *et al.* [21] propose four different matching strategies to compare each time step of the question against other time steps of the answer.

These are general one-way attention methods which use questions to guide attentive pooling of answers, or leverage answers to guide questions, or combine them together. However, the one-way attention mechanism ignores the interaction between questions and answers, which will influence the information extraction from both questions and answers.

Seo *et al.* [22] illustrate a bidirectional attention flow model which calculates similarity matrix between two sentences and obtains a query-aware representation without early max pooling. Santos *et al.* [23] present attentive pooling networks, which use interactive attention to generate attentive vectors and guide the attentive pooling next. Zhang *et al.* [24] improve the interaction matrix by concatenating each time step of the question and answer.

However, approaches described above only use single scale interaction between two sentences, which performs well on English dataset or Chinese dataset of general domain. When they are directly transferred to medical domain, they are likely to suffer significant performance degradation [3].

III. MODEL

In this section, we first offer a general overview of our proposed multi-scale interactive networks, and then decompose and introduce each component in detail.

Fig. 2 illustrates our end-to-end multi-scale interactive networks framework. The question and answer are first cut into characters and encoded as character-level embeddings, which are then fed into bidirectional gated recurrent units networks (biGRUs). The outputs of biGRUs along with the embeddings delivered by shortcut connections are then conveyed

to several multi-scale convolutional neural networks. After that, the CNNs' outputs of the question and answer interact with each other to generate attentive interactions, which are the weights for attentive pooling to generate vectors. Fig. 4 illustrates the detailed structure of our attentive interaction. The representation of the question and the answer is the max pooling of different scales vectors. Finally, we use cosine metrics to measure the similarity between the question and answer. It is worth mentioning that layers of the question and answer with the identical color in the figure share the same parameters of neural networks.

A. EMBEDDINGS

Embedding Layer is usually used at the beginning of the networks to encode words into fixed-length vectors. Word2Vec [25] and Glove [26] are the most prevalent tools for training word vectors recently. Commonly, word vectors are pre-trained with corpus, and stored in embedding tables. Therefore, the embedding layer can be regarded as table lookup: look up each word's embeddings in pre-trained embedding tables and then concatenate them into embedding matrix for the subsequent usage.

Owing to the lack of separators in Chinese text, the word segmentation, which cuts sentence into words, is a crucial preprocessing step in Chinese natural language processing. According to the work of Zhang *et al.* [3], who demonstrate that the medical text contains a large variety of professional terms, making it more difficult for existing tools to do word segmentation tasks. As a result, we adopt character-level embeddings rather than word-level embeddings in this paper to avoid the problems brought by word segmentation in medical text.

Given a question sentence $S_q = [s_1, s_2, \cdots, s_{l_q}]$ and a candidate answer $S_a = [s_1, s_2, \cdots, s_{l_a}]$, where l_q and l_a are lengths of the question and answer respectively, and s_i is the index value of each character in the vocabulary. After the embedding layer, they are encoded into $E_q = [c_1, c_2, \cdots, c_{l_q}]$ and $E_a = [c_1, c_2, \cdots, c_{l_a}]$, where $E_q \in \mathbb{R}^{l_q \times d_c}$ and $E_a \in \mathbb{R}^{l_a \times d_c}$ are character-level embeddings for the question and answer, and the length of each character-level embedding is d_c.

B. GATED RECURRENT UNITS

The Recurrent Neural Network (RNN) is a neural network that repeatedly transmits information from current node to the next node along the time step. Therefore, recurrent neural networks are naturally suitable for temporal and sequential data, and wildly used in time series analysis and natural language processing. The RNN is capable of extracting sequential and global information of the text. However, the standard recurrent neural networks may suffer gradient vanishing or gradient exploding problems [27], which reduce the accuracy and make the training harder.

Gated Recurrent Units networks (GRUs) [15], which introduce gated mechanism in hidden state, are the variant of

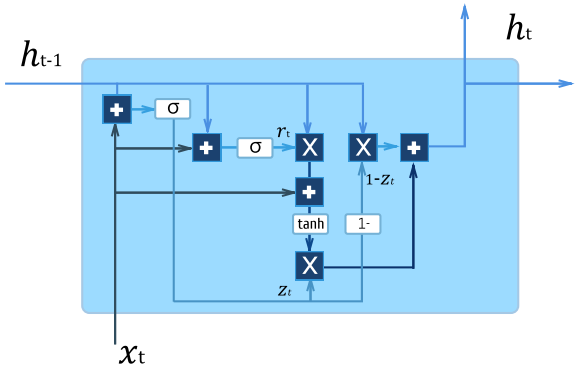


FIGURE 3. The architecture of the gated recurrent units network. Squares represent matrix operation, and the rounded rectangles represent function operation. h_t is the hidden state and x_t is the input at current time step t. The reset gate r_t and update gate z_t control the information flow.

Recurrent Neural Networks (RNNs). The GRUs simplify the long short-term memory (LSTM) networks and consume less computer memory. Instead of using memory cells in hidden state, the GRU network controls information from previous state to current state by adjusting the value of gates.

The architecture of GRU is shown in Fig. 3. There are two gates control the information flow: the reset gate r_t and the update gate z_t. The formula of the GRU is given as follows:

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]), \tag{1}$$
$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]), \tag{2}$$
$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]), \tag{3}$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t, \tag{4}$$

where the reset gate r_t determines the amount of the past information to be used to combine input information for computing the candidate hidden node $\tilde{h}_t$, and the update gate z_t controls how much of the past information h_{t-1} to be used for updating current hidden node h_t.

The word in the text may refer to the information forward or backward. The unidirectional recurrent neural network can not capture the contextual information of the latter part of the sequence very well. Thus, a bidirectional recurrent neural network is introduced, which processes the contextual information from two directions and generates two different output vectors at each time step. These two directions do not affect each other, and two vectors from two directions are concatenated into one vector as the final output at each time step.

The medical text in community forum is quite long, especially when we adopt character-level embeddings. Using standard recurrent neural networks may suffer gradient vanishing problems when training the network. Also, long short-term memory networks consume too much computer memory when dealing with long text sequence. Therefore, in this paper, bidirectional gated recurrent units networks are adopted.

Let the output of the forward GRU network at time step t be $\overrightarrow{h_t}$ and the output of the backward be $\overleftarrow{h_t}$, then the output of the bidirectional GRU at each time step is $h_t = \overrightarrow{h_t} || \overleftarrow{h_t}$. The bidirectional GRU layer can be denoted as:

$$H = \text{biGRU}(E), \tag{5}$$

where $E \in \mathbb{R}^{l \times d_c}$ is the embedding input of the neural network, $H \in \mathbb{R}^{l \times 2h_r}$ is the output of the bidirectional GRU layer.

C. SHORTCUT CONNECTIONS

Many complicated neural network frameworks leverage multiple stacked neural network layers to extracting deep features, which bring the drawback that deep neural networks will suffer degradation problem: with the layers of the network increasing, the accuracy of the network degrades rapidly.

In order to mitigate the problem, He *et al.* [28] propose the Deep Residual Networks (ResNet), which add shortcut connections among the neural network layers. These shortcut connections accelerate the training process of the network and effectively restrain the accumulation of errors in deeper layers. The shortcuts have been widely used in a wide diversity of research, such as image recognition [28], natural language inference [29], speech recognition [30], many-task modeling [31], reinforcement learning in mastering the game of Go [32].

In this paper, we add short connections between the embedding layer and the bidirectional GRU layer, which can be denoted as:

$$R = [\text{biGRU}(E), E] = [H, E], \tag{6}$$

namely, the output of the shortcut $R \in \mathbb{R}^{l \times (d_c + 2h_r)}$ is the concatenation of bidirectional GRU and character-level embeddings.

D. MULTI-SCALE CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNNs) extract local features of the input with a fixed length sliding window, which is called the feature map. In natural language processing, one dimensional convolutional operation is commonly adopted to extract features along the text sequence.

As we adopt character-level embeddings, using the convolutional neural network layer with one scale of feature map is not sufficient. Hence, the multi-scale convolutional neural networks [3] are introduced to capture different levels of granularity from the character and word level to the phrase or even the sentence level, since Chinese words consist of different numbers of characters.

Suppose the multi-scale convolutional neural networks have different types of feature maps $S = \{s_1, s_2, \cdots, s_t\}$, where the width of the i^{th} feature map $W_j^{s_i} \in \mathbb{R}^{s_i \times (d_c + 2h_r)}$ is s_i and the length is $(d_c + 2h_r)$. The convolution layer operates on continuous s_i vectors within its sliding windows. Suppose $Z_k = [r_k, r_{k+1}, \cdots, r_{k+s_i-1}]$ is the concatenation of continuous vectors from r_k to r_{k+s_i-1}, where $r_k \in \mathbb{R}^{d_c + 2h_r}$, the convolutional operation can be defined as follows:

$$O_j^{s_i} = f(W_j^{s_i} \circ [Z_1, Z_2, \cdots, Z_l] + b), \tag{7}$$

where $W \circ Z = \sum_{p,q} W_{pq} Z_{pq}$ means the summation of element-wise multiplication, $f(\cdot)$ is the activation function and b is the bias.

Given h_c feature maps, the output of convolutional neural network with filter size s_i is the concatenation of $O_j^{s_i}$, which is:

$$O^{s_i} = [O_1^{s_i}, O_2^{s_i}, \cdots, O_{h_c}^{s_i}], \tag{8}$$

where $O^{s_i} \in \mathbb{R}^{l \times h_c}$.

In order to simplify the representation, the convolutional neural network layer can be noted as:

$$O^{s_i} = \text{CNN}^{s_i}(R). \tag{9}$$

E. MULTI-SCALE ATTENTIVE INTERACTION LAYERS

A question and its corresponding answer usually have similar semantic expression, context or structure. By mining their correlation, the right answer can be distinguished from inappropriate answers.

The neural network models described above are useful when extracting deep textual features. However, they just extract the question and answer separately. Santos *et al.* [23] introduce attentive pooling networks which enlighten the thought of our multi-scale attentive interaction layers.

Fig. 4 shows the structure of our one-scale attentive interaction layers. The multi-scale attentive interaction is the correlation between every two CNNs' outputs of the question and answer.

Let $O_q^{s_i} \in \mathbb{R}^{l_q \times h_c}$ be the CNN output for the question, and $O_a^{s_j} \in \mathbb{R}^{l_a \times h_c}$ be the CNN output for the answer. The interactive attention can be calculated as:

$$I^{ij} = \sigma(O_q^{s_i} \cdot U \cdot {O_a^{s_j}}^T), \tag{10}$$

where $I^{ij} \in \mathbb{R}^{l_q \times l_a}$ is the attentive interaction matrix that indicates the relationship between the i^{th} CNN output of the question and the j^{th} CNN output of the answer, and $U \in \mathbb{R}^{h_c \times h_c}$ is the parameter which is learned through training the neural network. The element $I_{(m,n)}^{ij}$ indicates the similarity between the m^{th} position of the question and the n^{th} position of the answer. $\sigma(\cdot)$ is the sigmoid function, which can also be other activate functions, such as $\tanh(\cdot)$.

After that, we perform row-wise and column-wise max pooling to acquire attentive vectors:

$$\text{att}_q^{ij} = \max_n \frac{e^{I_{(m,n)}^{ij}}}{\sum_n e^{I_{(m,n)}^{ij}}}, \tag{11}$$

and

$$\text{att}_a^{ij} = \max_m \frac{e^{I_{(m,n)}^{ij}}}{\sum_m e^{I_{(m,n)}^{ij}}}, \tag{12}$$

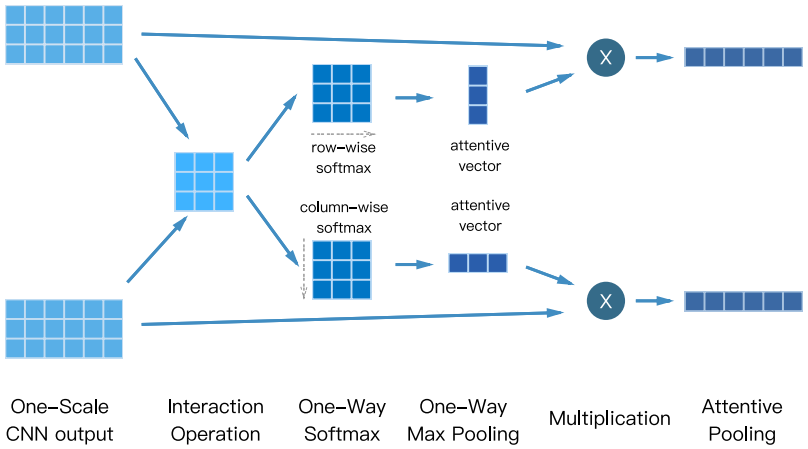


FIGURE 4. The architecture of the one-scale attentive interaction layer. The outputs of one-scale CNN interact with each other to generate the interaction matrix. One-way softmax and max pooling are applied to generate attentive vectors of the question and answer, which are then applied to perform attentive pooling.

where $\text{att}_q^{ij} \in \mathbb{R}^{l_q}$ and $\text{att}_a^{ij} \in \mathbb{R}^{l_a}$ are two interactive attention vectors which represent the importance of the information at each time step, and max is the function to get the element-wise maximum.

F. REPRESENTATION AND SIMILARITY

In order to generate the finial vector representation of the question and answer, some research just uses max pooling over all time steps of the former layer [6]. However, it has a drawback that the max pooling treats all time steps equally, which is inconsistent with the thought that some parts of the sentence contribute more to the task.

As for question answer matching task, attention mechanism is used to concentrate on question-related parts in the answer. For example, the representation of the question is used to guide the pooling process of the answer [19].

In this paper, we leverage attentive pooling to simplify the representation of the question and answer. The attentive pooling uses attentive interaction vector acquired before as weight to sum the output of CNN, which can be noted as:

$$q^{ij} = O_q^{s_i} \cdot \text{att}_q^{ij}, \tag{13}$$

and

$$a^{ij} = O_a^{s_j} \cdot \text{att}_a^{ij}, \tag{14}$$

After that, we can acquire the final vector representation of the question and answer by maximizing the element of each vector:

$$q = \max\{q^{11}, q^{12}, \cdots, q^{21}, q^{22}, \cdots, q^{s_t s_t}\}, \tag{15}$$

and

$$a = \max\{a^{11}, a^{12}, \cdots, a^{21}, a^{22}, \cdots, a^{s_t s_t}\}. \tag{16}$$

In order to accelerate the calculation, we simplify the formula by just calculating the attentive interaction between questions and answers with the same granularity. Thus we can get:

$$q = \max\{q^{11}, q^{22}, \cdots, q^{s_t s_t}\}, \tag{17}$$

and

$$a = \max\{a^{11}, a^{22}, \cdots, a^{s_t s_t}\}. \tag{18}$$

Then, the similarity between the question and answer can be calculated by cosine distance, of which the definition is:

$$\text{Sim}(q, a) = \text{Cosine}(q, a) = \frac{||q \cdot a||}{||q|| \cdot ||a||}. \tag{19}$$

where $||\cdot||$ is the vector length.

For each question q_i in the training set, its corresponding answer is denoted as a_i^+, and a randomly selected answer from the whole answer pool is noted as a_i^-. The aim of our models is to maximize $\text{Sim}(q_i, a_i^+)$, and minimize $\text{Sim}(q_i, a_i^-)$. We use the margin loss to optimize the parameters of neural networks. The margin loss function used in training networks is described as follows:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \max\{0, M - \text{Sim}(q_i, a_i^+) + \text{Sim}(q_i, a_i^-)\}, \tag{20}$$

where margin value M is a constant, which controls the margin between good answer and bad answer, and N is the total number of tuple (q_i, a_i^+, a_i^-).

After that, we use Adaptive Gradient (Adagrad) algorithm [33] to train and update the parameters of the framework. The algorithm uses a relatively large learning rate at the beginning, and decreases gradually as the model continues training.

IV. EXPERIMENTS

In this section, we give a detailed description about our experiments. First the dataset and metrics used in this paper are introduced, followed by experimental settings. At last, the results and analysis are shown at the end of the section.

TABLE 1. The statistics of cMedQA v2.0 dataset. Ave.: abbreviation for Average. "#" denotes "the number of".

	#Question	#Answer	Ave. #Character Per Question	Ave. #Character Per Answer
Training	100,000	188,490	48	101
Development	4,000	7,527	49	101
Test	4,000	7,552	49	100
Total	108,000	203,569	49	101

A. DATASET

The dataset used in this paper is called cMedQA v2.0, which is the extension and amendment of version 1.0. We collect the data from an online Chinese medical question answering forum (http://www.xywy.com/). Qualified doctors will give answers to questions asked by Internet users. In the forum, doctors give the diagnosis and suggestions according to the symptoms that users describe. We suppose that the answer responded by a qualified doctor is the ground truth answer to the original question.

To protect the privacy of users, we anonymize the data and remove all possible personal information. The dataset is available at website (https://github.com/zhangsheng93/cMedQA2) for non-commercial research.

The statistics of cMedQA v2.0 are summarized in Table 1. Compared to version v1.0, the cMedQA v2.0 dataset doubles the number of questions and answers, and performs meticulous data cleaning preprocessing steps, such as eliminating greeting words, replacing English punctuation with Chinese punctuation. As is shown in Table 1, the dataset consists of 3 parts: training set, development set and test set. The training set is used to train the model, the development set is used to tune the hyper-parameters in models, and the test set is used to evaluate different models. The average character number of questions is about 49, and that of answers is about 101.

During training the framework, for each question in the training set, we repeatedly generate 50 tuples of (q, a^+, a^-), where a^+ is the ground truth answer of question q, and a^- is randomly selected from the answer pool. Hence, during each epoch of the training process, $5,000,000$ tuples are fed into the model, which are more than three times as many tuples as those in dataset version v1.0.

During development and test process, for each question, it has 100 candidate answers with the ground truth answer included. The target is to choose the most suitable answer which is exactly the ground truth answer from candidate answers.

B. METRICS

We use top-1 accuracy, denoted as ACC@1, as the metrics in our paper. The top-k accuracy (ACC@k) is commonly used in various information retrieval tasks. The formula of ACC@k is:

$$\text{ACC@}k = \frac{1}{N}\sum_{i=1}^{N} 1[a_i^+ \in C_i^k] \tag{21}$$

where N is the number of samples in development or test set, $a_i^+ \in \alpha_i^+$ is one ground truth answer of question q_i, C_i^k is the set of candidate answers with the top-k highest similarities, and $1[\cdot] \rightarrow \{0, 1\}$ is the indicator function, which equals 1 if the logical expression in the bracket is true or equals 0 otherwise.

C. BASELINES

In order to prove the superiority of our model, we compare our model with some baselines, which are listed as follows:

- **Multi-Scale CNNs**: The multi-scale convolutional neural networks model [3] uses different sizes of feature maps to extract the semantic information from the text. The model shows a significant improvement when compared to statistical models.
- **Multi-Level Composite CNNs**: This model proposed by Ye *et al.* [4] reaches the state-of-the-art performance on cMedQA v1.0 dataset. The model combines the intermediate features extracted from each layer of the multi-level CNNs instead of just stacking different networks.

D. EXPERIMENTAL SETTINGS

For the preprocessing process, we adopt the corpus of training data to pre-train character-level vectors, whose length is $l = 300$. The character list contains $4,067$ characters. Out-of-vocabulary characters are initialized as zero vectors. All character vectors are fed into the neural network and not updated during the training process. We set the max length of the sentence to $l = l_q = l_a = 200$, and apply zero padding if the sentence is short or truncation if the sentence is long to all questions and answers. In order to compare word-level embeddings with character-level emebddings, we also perform experiments on cMedQA v2.0 dataset using word-level embeddings. The word segmentation tool used in our experiment is ICTCLAS (pynlpir).

As for our framework, the output shape of bidirectional GRU is 300 ($h_r = 150$ for each direction). The output shape of one scale CNN is $h_c = 500$. The margin value M of the loss function is set to 0.1.

For training the neural networks, we employ Adagrad Optimizer with 256 batch size and the initial learning rate is 0.01.

E. RESULTS

Table 2 illustrates the evaluation of several models on cMedQA v2.0 dataset. The first column is the index and the second is the model used in the experiment. The third and forth columns are the top-1 accuracy on development and test set, respectively. The multi-scale CNNs (Row 3) and "2CNNs" use the feature maps with size 2 and 3, while "3CNNs" means the convolutional neural networks which adopt the feature maps with size 1, 2 and 3. Multi-Level Composite CNNs (Row 7) use the feature maps with size 3 and have 3 CNN layers.

Rows 1 to 3 show results of the models which have one layer in common. Bidirectional GRU performs better than

TABLE 2. The top-1 accuracy (ACC@1) of neural network models. biGRU: bidirectional gated recurrent unit; CNN: convolutional neural network.

id	Model	Dev(%)	Test(%)
1	biGRU	68.9	68.7
2	CNN	67.6	67.8
3	Multi-Scale CNNs [3]	70.0	70.9
4	biGRU+CNN	69.5	70.0
5	CNN+biGRU	67.9	67.7
6	biGRU+biGRU	68.4	68.7
7	Multi-Level Composite CNNs [4]	70.4	70.1
8	biGRU+2CNNs	70.2	70.3
9	biGRU+shortcuts+CNN	70.8	71.4
10	biGRU+shortcuts+biGRU	70.7	71.4
11	biGRU+CNN+interact	70.1	71.1
12	biGRU+2CNNs+interact	71.7	70.9
13	biGRU+shortcuts+CNN+interact	70.9	71.2
14	biGRU+shortcuts+2CNNs+interact	72.1	**72.1**
15	biGRU+shortcuts+3CNNs+interact	**72.3**	71.6

CNN, while multi-scale CNNs reach the highest score among three single layer networks, as the framework extracts different levels of granularity from character-level embeddings.

Rows 4 to 8 present the performance of two-layer models. Instead of just stacking different neural networks (Rows 4-6), multi-level composite CNNs have a better accuracy score by extracting semantic features from each layer and combining them to enrich the final feature representation. In addition, bidirectional GRU combined with multi-scale CNNs (Row 8) also reaches a similar performance, showing good adaptability of multi-scale CNNs in handling character-level embeddings.

Rows 9 to 10 describe the results of two-layer models with shortcut connections. It is shown that models with shortcut connections surpass models without shortcuts (Rows 4 and 6).

Rows 11 to 12 illustrate the performance of two-layer models with attentive interaction. The test performance increases as the question and answer interact with each other, demonstrating that attentive interaction contributes to understand character-level information.

Rows 13 to 15 summarize our multi-scale interactive model with shortcut connections. Compared to the single scale model (Row 13), our multi-scale model reaches the highest top-1 accuracy. The model with two-scale (2,3) filter sizes achieves a new state-of-the-art performance on test dataset.

Table 3 reveals the results of our model compared to previous state-of-the-art models on both cMedQA v1.0 dataset and v2.0 dataset. First three rows illustrate the performance of models with word-level embeddings while the following three rows present the results of models with character-level embeddings. As can be seen from the table, models using character-level embeddings surpass those using word-level embeddings over about 9% and 3% on two datasets respectively. When it comes to different models, our multi-scale attentive interaction networks model surpasses two former state-of-the-art models on both cMedQA v1.0 version and v2.0 version. Due to more training samples and fine-tuned preprocessing, the top-1 scores of three models on cMedQA v2.0 show a noticeable increase. More data means complicated models can be well trained with more training samples.

TABLE 3. Comparison on two datasets. CNN: convolutional neural network.

Embeddings	Model	cMedQA v1.0		cMedQA v2.0	
		Dev(%)	Test(%)	Dev(%)	Test(%)
Word	Multi-Scale CNNs [3]	54.1	55.4	67.1	68.6
	Multi-Level Composite CNNs [4]	56.3	58.3	67.4	67.4
	Multi-Scale Attentive Interaction Networks	-	-	**68.9**	**69.2**
Character	Multi-Scale CNNs [3]	65.4	64.8	70.0	70.9
	Multi-Level Composite CNNs [4]	65.6	66.2	70.4	70.1
	Multi-Scale Attentive Interaction Networks	**66.1**	**67.1**	**72.1**	**72.1**

TABLE 4. Ablation analysis of our model. "+" denotes adopting the component, while "-" denotes removing the component from model in Row 1. CNN: convolutional neural network.

	Components	Test(%)
1	biGRU +shortcuts +2CNNs +interact	72.1
2	-shortcuts	-0.7
3	-CNN	-0.9
4	-interact	-1.2
5	-shortcuts -CNN	-1.0
6	-shortcuts -interact	-1.8
7	-shortcuts -CNN -interact	-2.1
8	-shortcuts -2CNNs -interact	-3.4

To summarize, our models are superior to single layer or two-layers models. This indicates that the new components of our models play an important role in extracting medical text, which is thoroughly discussed in Subsection V-A. Also, our models outperform the former state-of-the-art models, not only multi-scale CNNs but also multi-level composite CNNs, with noticeable margins.

V. DISCUSSION

A. ABLATION ANALYSIS

In order to compare the importance and contribution of components in our model, ablation analysis is applied in this subsection. Each time we remove one or more components from the original model and evaluate the rest model on test set.

Table 4 shows the performance of our model and its ablations. The first row is the result of our model, the second column lists all key components of the model. From Rows 2 to 8, each ablation removes specific components from the original model. Rows 2 to 4 remove single component, while Rows 5 to 8 remove multiple components of our model.

We can see that all components positively contribute to the performance of our model. Components sorted by their contribution in descending order are attentive interaction, shortcut connections and multi-scale CNNs respectively. It is noted that attentive interaction plays a more significant role in our model than shortcut connections and multi-scale CNNs. As we continue to remove different components, the top-1

accuracy of our model on test set decreases gradually, demonstrating that each component in our model contributes to our framework. We can also see that removing multiple components hurts performance dramatically, which indicates that multi-layer neural networks are crucial to extract character-level embeddings.

To summarize, the ablation analysis has demonstrated that multi-scale convolutional neural networks and attentive interaction are two key components in capturing character-level semantic information in Chinese medical question answer selection tasks.

B. INTERACTION ANALYSIS

The interaction matrix I^{ij}, shown in Equation (10), is a signal of the correlation between questions and answers. Fig. 5 presents the heat map of the interaction between the question and its corresponding answer in Fig. 1 in terms of different-scale CNNs' outputs. To be specific, the filter width of convolutional layers in Fig. 5(a) is two, while that in Fig. 5(b) is three. The rows represent CNNs' output of the question, and the columns represent that of the answer. As the filter width goes larger, the colors of blocks become darker.

In terms of the problem, there are some shallow blocks, which seem less significant in finding good answer. On the contrary, there are some dark blocks, which may be the clue for good answer and should be given more weight when performing attentive pooling.

Fig. 1 illustrates an example that models without attentive interaction components may incorrectly choose an inappropriate answer. The user wants to ask the question about his mother's anemia rather than fever. Symptoms described in the question such as ''yellow skin'', ''frequent faint'' and ''poor appetite'', are usually caused by anemia. Despite ''high body temperature'' mentioned in the problem, the user points out that this is not caused by fever. The irrelevant answer describes the cause of fever and countermeasures, which is apparently unsuitable for this question, but the good answer illustrates symptoms of anemia and provides useful advice for users.

By comparison analysis we find that the model proposed in this paper can effectively extract the correlation of the question and answer. We can see from the figures that n-gram words which contain similar semantic meaning between the question and answer have the darker blocks. Also, 3-gram interaction matrix has more darker blocks which mainly appear on the right side of the matrix.

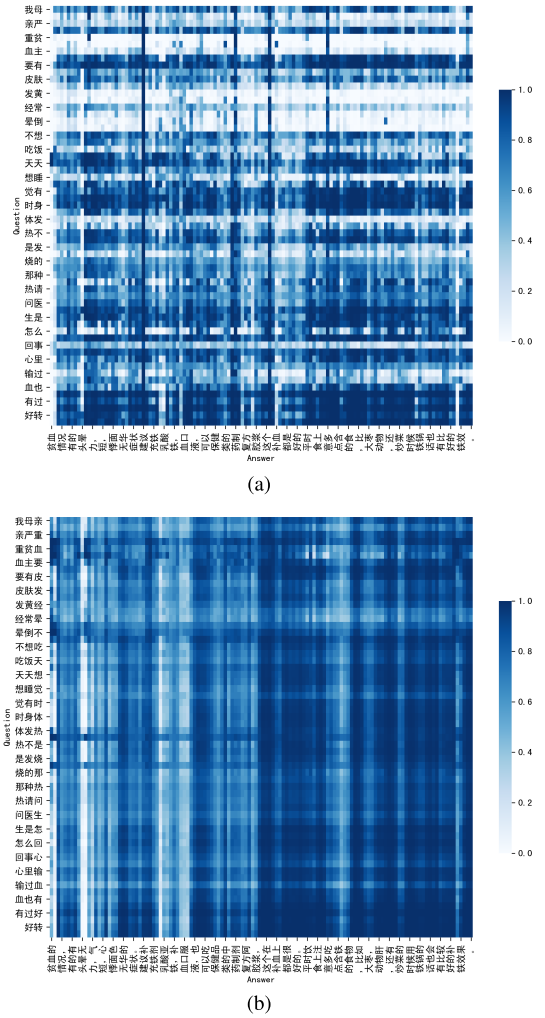


FIGURE 5. The visualization of attentive interaction matrices. The heat map represents the soft alignment between the question and the answer. The rows represent CNNs' output of the question, and the columns represent that of the answer. The depth of the color represents the magnitude of the similarity. The deeper the color, the higher the similarity. (a) The heat map of 2-gram outputs of CNNs. The filter width of convolutional layers is two. (b) The heat map of 3-gram outputs of CNNs. The filter width of convolutional layers is three.

VI. CONCLUSIONS

In this paper, we introduce an end-to-end multi-scale interactive networks framework for Chinese medical question answer selection. The framework not only sufficiently captures the semantic information from character-level embeddings, but also extracts the correlative information between questions and answers. The experimental results conducted in this work demonstrate that our model achieves a competitive performance and outperforms former state-of-the-arts models on the Chinese medical question answer selection dataset.

Our future work would like to integrate the medical knowledge base into our model to better support medical question answering. The knowledge base contains structured knowledge and well processed data. Knowledge map usually contains processed data, which is very convenient for machine to understand and utilize. Some research [34]–[36] has investigated hybrid methods which apply structured knowledge data (such as DBpedia, Freebase) and unstructured text to general question answering. However, knowledge based question answering in medical field falls short of relevant research due to lack of medical field knowledge bases. In the future, constructing such a medical knowledge database or using

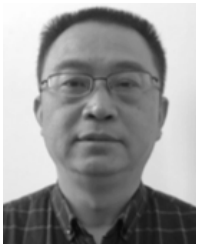
related technologies in transfer learning is considered to try to solve this problem.

REFERENCES

[1] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.
[2] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep patient: An unsupervised representation to predict the future of patients from the electronic health records," *Sci. Rep.*, vol. 6, May 2016, Art. no. 26094.
[3] S. Zhang, X. Zhang, H. Wang, J. Cheng, P. Li, and Z. Ding, "Chinese medical question answer matching using end-to-end character-level multi-scale CNNs," *Appl. Sci.*, vol. 7, no. 8, p. 767, 2017.
[4] D. Ye *et al.*, "Multi-level composite neural networks for medical question answer matching," in *Proc. IEEE 3rd Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2018, pp. 139–145.
[5] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. (2014). "A convolutional neural network for modelling sentences." [Online]. Available: https://arxiv.org/abs/1404.2188
[6] M. Feng, B. Xiang, M. R. Glass, L. Wang, and B. Zhou, "Applying deep learning to answer selection: A study and an open task," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand. (ASRU)*, Dec. 2015, pp. 813–820.
[7] X. Qiu and X. Huang, "Convolutional neural tensor network architecture for community-based question answering," in *Proc. IJCAI*, 2015, pp. 1305–1311.
[8] M. Tan, C. D. Santos, B. Xiang, and B. Zhou. (2015). "LSTM-based deep learning models for non-factoid answer selection." [Online]. Available: https://arxiv.org/abs/1511.04108
[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[10] S. Jain and T. Dodiya, "Rule based architecture for medical question answering system," in *Proc. 2nd Int. Conf. Soft Comput. Problem Solving (SocProS)*. New Delhi, India: Springer, 2012, pp. 1225–1233.
[11] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *Found. Trends Inf. Retr.*, vol. 3, no. 4, pp. 333–389, 2009.
[12] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.
[13] Y. Cui, Z. Chen, S. Wei, S. Wang, T. Liu, and G. Hu. (2016). "Attention-over-attention neural networks for reading comprehension." [Online]. Available: https://arxiv.org/abs/1607.04423
[14] K. M. Hermann *et al.*, "Teaching machines to read and comprehend," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1693–1701.
[15] K. Cho *et al.* (2014). "Learning phrase representations using RNN encoder-decoder for statistical machine translation." [Online]. Available: https://arxiv.org/abs/1406.1078
[16] D. Bahdanau, K. Cho, and Y. Bengio. (2014). "Neural machine translation by jointly learning to align and translate." [Online]. Available: https://arxiv.org/abs/1409.0473
[17] M.-T. Luong, H. Pham, and C. D. Manning. (2015). "Effective approaches to attention-based neural machine translation." [Online]. Available: https://arxiv.org/abs/1508.04025
[18] A. M. Rush, S. Chopra, and J. Weston. (2015). "A neural attention model for abstractive sentence summarization." [Online]. Available: https://arxiv.org/abs/1509.00685
[19] M. Tan, C. D. Santos, B. Xiang, and B. Zhou, "Improved representation learning for question answer matching," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 464–473.
[20] L. Song, Z. Wang, and W. Hamza. (2017). "A unified query-based generative model for question generation and question answering." [Online]. Available: https://arxiv.org/abs/1709.01058
[21] Z. Wang, W. Hamza, and R. Florian. (2017). "Bilateral multi-perspective matching for natural language sentences." [Online]. Available: https://arxiv.org/abs/1702.03814
[22] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. (2016). "Bidirectional attention flow for machine comprehension." [Online]. Available: https://arxiv.org/abs/1611.01603
[23] C. D. Santos, M. Tan, B. Xiang, and B. Zhou. (2016). "Attentive pooling networks." [Online]. Available: https://arxiv.org/abs/1602.03609
[24] X. Zhang, S. Li, L. Sha, and H. Wang, "Attentive interactive neural networks for answer selection in community question answering," in *Proc. AAAI*, 2017, pp. 3525–3531.
[25] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[26] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[27] S. C. Kremer and J. F. Kolen, Eds., *A Field Guide to Dynamical Recurrent Neural Networks*, 1st ed. Hoboken, NJ, USA: Wiley, 2001.
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[29] Y. Nie and M. Bansal. (2017). "Shortcut-stacked sentence encoders for multi-domain inference." [Online]. Available: https://arxiv.org/abs/1708.02312
[30] Y. Zhang, G. Chen, D. Yu, K. Yaco, S. Khudanpur, and J. Glass, "Highway long short-term memory RNNS for distant speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 5755–5759.
[31] K. Hashimoto, C. Xiong, Y. Tsuruoka, and R. Socher. (2016). "A joint many-task model: Growing a neural network for multiple NLP tasks." [Online]. Available: https://arxiv.org/abs/1611.01587
[32] D. Silver *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, p. 354, 2017.
[33] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.
[34] Y. Feng *et al.*, "Hybrid question answering over knowledge base and free text," in *Proc. COLING 26th Int. Conf. Comput. Linguistics, Tech. Papers*, 2016, pp. 2397–2407.
[35] R. Das, M. Zaheer, S. Reddy, and A. McCallum. (2017). "Question answering on knowledge bases and text using universal schema and memory networks." [Online]. Available: https://arxiv.org/abs/1704.08384
[36] U. Sawant, S. Garg, S. Chakrabarti, and G. Ramakrishnan. (2017). "Neural architecture for question answering using a knowledge graph and Web corpus." [Online]. Available: https://arxiv.org/abs/1706.00973

SHENG ZHANG received the B.S. degree in systems engineering and the M.S. degree in management science and engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the College of Systems Engineering. His research interests include natural language processing, deep learning, and data mining.

XIN ZHANG received the B.S. and Ph.D. degrees in system engineering from the National University of Defense Technology (NUDT), China, in 2000 and 2006, respectively. He is currently a Professor with the State Key Lab of Information System Engineering, College of Systems Engineering, NUDT. His research interests include cross-modal data mining, information extraction, and event analysis.



HUI WANG received the B.S., M.S., and Ph.D. degrees in system engineering from the National University of Defense Technology (NUDT), China, in 1990, 1998, and 2005, respectively. He is currently a Professor with the State Key Lab of Information System Engineering, College of Systems Engineering, NUDT. His research interests include natural language processing, deep learning, data mining, and social analysis.

LIXIANG GUO was born in Wuhan, China, in 1991. He received the B.Eng. degree in systems engineering and the M.Eng. degree in management science and engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2014 and 2016, respectively, where he is currently pursuing the Ph.D. degree with the College of Systems Engineering. His research interests include information extraction, data mining, and text analytics.

SHANSHAN LIU was born in Gansu, China, in 1994. She received the B.A. degree from the School of Information Management, Nanjing University, in 2017. She is currently pursuing the master's degree in management science and engineering with the College of System Engineering, NUDT, Changsha. Her research interests include natural language processing, data mining, and social computing.