IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 30, 2019, accepted October 16, 2019, date of publication October 30, 2019,
date of current version November 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2949993

Answer Acquisition for Knowledge Base Question Answering Systems Based on Dynamic Memory Network

LEI SU[1], TING HE[1], ZHENGYU FAN[1], YIN ZHANG[2], (Senior Member, IEEE), AND MOHSEN GUIZANI[3], (Fellow, IEEE)
[1]School of Information and Automation, Kunming University of Science and Technology, Kunming 650500, China
[2]School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China
[3]Department of Computer Science and Engineering, Qatar University, Doha, Qatar

Corresponding author: Lei Su (s28341@hotmail.com)

This work was supported by the National Science Foundation of China under Grant 61365010.

ABSTRACT In recent years, with the rapid growth of Artificial Intelligence (AI) and the Internet of Things (IoT), the question answering systems for human-machine interaction based on deep learning have become a research hotspot of the IoT. Different from the structured query method in traditional Knowledge Base Question Answering (KBQA) systems based on templates or rules, representation learning is one of the most promising approaches to solving the problems of data sparsity and semantic gaps. In this paper, an answer acquisition method for KBQA systems based on a dynamic memory network is proposed, in which representation learning is employed to represent the natural language questions that are raised by users and the knowledge base subgraphs of the related entities. These representations are taken as inputs of the dynamic memory network. The correct answers are obtained by utilizing the memory and inferential capabilities. The experimental results demonstrate the effectiveness of the proposed approach.

INDEX TERMS Internet of things, human-machine interaction, knowledge base question answering systems, dynamic memory network.

I. INTRODUCTION

With the integration and development of artificial intelligence (AI) technology and Internet of Things (IoT) technology, human-machine interaction question answering (QA) systems are considered to be an important research direction of the IoT. Traditional search engines method require users to input one or more keywords to return massive web links lists, and the system cannot directly give the answer that users want. In the IoT, human-machine interaction QA systems provide people with a good human-computer interaction interface. It is based on natural language processing (NLP), which can provide users with personalized services by using knowledge base (KB) retrieval to return accurate answers in real time for the natural language questions that are raised by users in real environments. At present, intelligent Chinese QA systems have been widely applied in the IoT. People want to obtain answers directly via using an intelligent Chinese QA system to related questions in the IoT environment. For example, users can answer questions and automatically control a smart home environment with home devices. In distance education, students can consult with the devices to answer questions. With respect to the online customer service of a financial institution, a QA system can automatically give answers to the questions that are raised by users. Finally, a human-machine interactive QA system can be also applied to answer the colloquial questions that are asked by players on a game platform. For the human-machine interaction QA system of the IoT, our work based on deep learning combines the dynamic memory network (DMN) [1] with presentation learning and uses simple and precise answers to automatically respond to the questions that are asked by users in natural language. This improves the query efficiency and provides users with a more natural way of human-machine interaction.

Knowledge Base Question Answering (KBQA) automatically returns answers from a large-scale structured KB given natural language questions, which is an important direction of current QA. In recent years, there are several large-scale knowledge bases that have emerged, such as YAGO [2], Freebase [3], DBpedia [4], and Chinese knowledge bases

The associate editor coordinating the review of this manuscript and approving it for publication was Hong-Ning Dai.

such as CN-Probase [5] and CN-DBpedia [6]. The entities and properties in the KB are kept in a discrete graph structure, and there is a great amount of low-frequency knowledge in a KB. Therefore, a high-dimensional representation that is generated by using one-hot encoding cannot address the data sparsity problem [7]. Existing KBQA is regarded as a process that calculates similarities between questions and entities or edges in KB. However, it cannot take full advantage of KB resources and bridge the semantic gap. With the progress of deep learning, great breakthroughs have been made in the fields of image, video, voice and NLP [8]. Yih *et al.* [9] use a convolutional neural network (CNN) to compute the similarity of attributes between the questions and the entities in the KB, and then it selects the triples with the highest similarity as final answers. Dong *et al.* [10] propose a multicolumn CNN model to represent questions and candidate answers in a KB with respect to different answer aspects including answer paths, answer contexts, and answer types. Then, it grades the questions and candidate answers to determine the best answer. Bordes *et al.* [11] build a KBQA system with a memory network (MN) model. First, they leverage the input module to parse the KB and save the triples to the memory. Next, they take all question-answer pairs as inputs to find the candidate facts using the entity links (ELs). Finally, it outputs the most relevant facts via an embedding model.

Traditional KBQA methods (such as, Semantic Parsing [12], rule-based and template-based methods [13].) cannot be applied to the large-scale KBQA. Moreover these methods cannot tackle the large-scale linked data, serious ambiguity, etc. In view of the data sparsity and semantic gap of traditional methods, we utilize representation learning to learn the questions, the related KB entities and the properties [14], and deduce the correct answer via a DMN [1]. For each question *q*, we use the named entity recognition (NER) to identify a topic entity. After getting the topic entity, we collect the corresponding KB subgraph from the KB. Then, we take the related entities and properties of the KB subgraph as inputs to the DMN. Finally, we iteratively determine the answer using the memory module and answer module. In a real environment, the KB can be updated and improved in various ways. For example, we can use human-machine interaction to obtain new knowledge from the outside world, which can update and improve the KB to meet various information needs of users.

The structure of the remainder of this paper is organized as follows. Section II presents related work of KBQA and Chinese NER. In Section III, we introduce the answer acquisition method of the KBQA system based on a DMN. Section IV describes the experimental results and comparative analysis. Finally, we conclude this paper in Section V.

II. RELATED WORKS

A. KNOWLEDGE BASE QUESTION ANSWERING

In traditional KBQA when users ask a question, it first parses the question into a semantic representation via semantic analysis technology. Then, the answers in the KB are acquired via semantic matching, query and reasoning techniques. The architecture of traditional KBQA is shown in Figure 1. In recent years, deep neural networks (DNNs) have been widely applied to many NLP and KBQA tasks. Bordes *et al.* [15] introduce a neural network-based approach to tackle KBQA problems by transforming questions and triples into vector representations in a low-dimensional space via representation learning. Specifically, they convert the KBQA process into the process of matching the similarities between questions and candidate answers in the same semantic space and take the highest similarity score as the best answer. Subsequently, Bordes *et al.* [16] propose the concept of subgraph embedding to improve the above work.

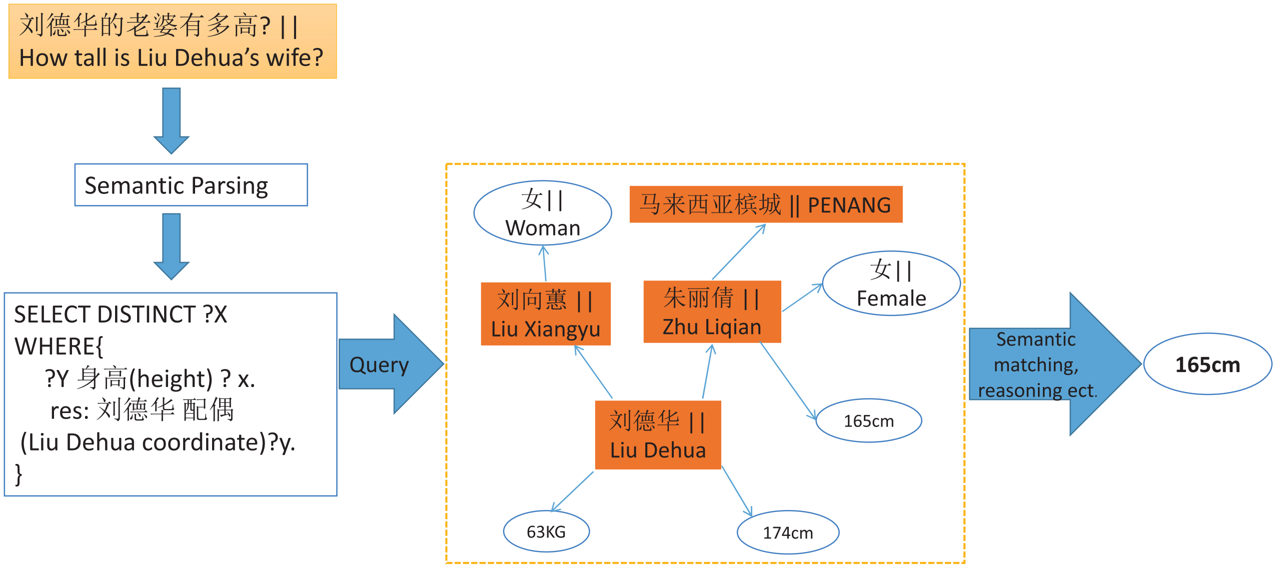


FIGURE 1. The architecture of traditional KBQA.

To address multi-hop link prediction, Zhang et al. propose a variational reasoning framework for KG (knowledge graph) reasoning, which combines path-finding and path- reasoning closely for joint reasoning. They utilize negative samples into training and improve the robustness of the existing KG reasoning mode [17]. To represent questions more clearly, Wang *et al.* [18] use recurrent neural networks (RNN) with an attention mechanism to express the questions and present three new RNN models with attention mechanisms, which achieve good results on answer selection tasks. Zhang *et al.* [19] present a neural attention-based model to express questions according to different aspects of the candidate answers with a global KB. Tan *et al.* [20] utilize bidirectional long short-term memory (Bi-LSTM) [21] to embed questions and answers in answer selection and compute their similarities using cosine similarity. Zhou *et al.* [22] propose a large-scale KBQA with long short-term memory (LSTM) [23], which divided KBQA into NER and property mapping and all entities are returned by constructing an alias dictionary. They combine an attention mechanism with Bi-LSTM to predict the attributes and obtain the final answer. Lai *et al.* [24] adopt word vector similarity and fine-grained word segmentation to map properties. Meanwhile, they use many artificially constructed rules and features to select the correct answers.

In Figure 1, we take the question ''How tall is Liu Dehua's wife?'' First, traditional KBQA uses semantic analysis technology to parse the question and obtains all entities and properties in n-hops from the corresponding entity ''Liu Dehua'' in the KB. Next, it compares the question to the KB subgraph via techniques such as semantic matching, querying, and semantic reasoning to gain the correct answer. For instance, the system first finds the entity ''Liu Dehua'' corresponding to ''Liu Dehua'' in KB and gets that the wife of ''Liu Dehua'' is ''Zhu Liqian'' according to the reasoning technique. Finally, it utilizes the semantic matching method to obtain the highest similarity as the final answer ''165 cm''.

B. CHINESE NAMED ENTITY RECOGNITION

The primary goal of NER is to identify the topic entity of a question and estimate the result. The KB subgraph that is centered on the topic entity is queried in the KB, and these entities and properties of the subgraph constitute candidate set answers. Lample *et al.* [25] adopt LSTM in the English NER task to construct words using letters, which were spliced into word vectors before being input to the LSTM to capture the morphological features. Dong *et al.* [26] extend this method to the Chinese NER task and construct Chinese characters with partial radicals. To accomplish the NER task and improve the coverage accuracy of the candidate answers, we utilize the Chinese NER method-based Bi-LSTM-CRF [26], [27] and use Bakeoff-3 [28] to evaluate the label sets. Concretely, B-PER and I-PER represent the first word of a person's name and noninitials of a person's name, respectively. B-LOC and I-LOC represent the first word of a place's name and the noninitials of a place's name, respectively. B-ORG represents the initial word of an organization's name. Conversely, I-ORG represents the words of an organization's name that are not the first, and O represents the part that is not a named entity [26], [27]. Bi-LSTM consists of both forward and backward networks. The forward LSTM processes the sentence from left to right, and the backward LSTM handles it in reverse order (one in sentence order and the other in reverse sentence order). Bi-LSTM concatenates the hidden unit of the forward and backward LSTM to represent each word, which means that each word contains both the information of the former word and latter word. It is more conducive to labeling each word. Figure 2 is the architecture of the Chinese NER model.

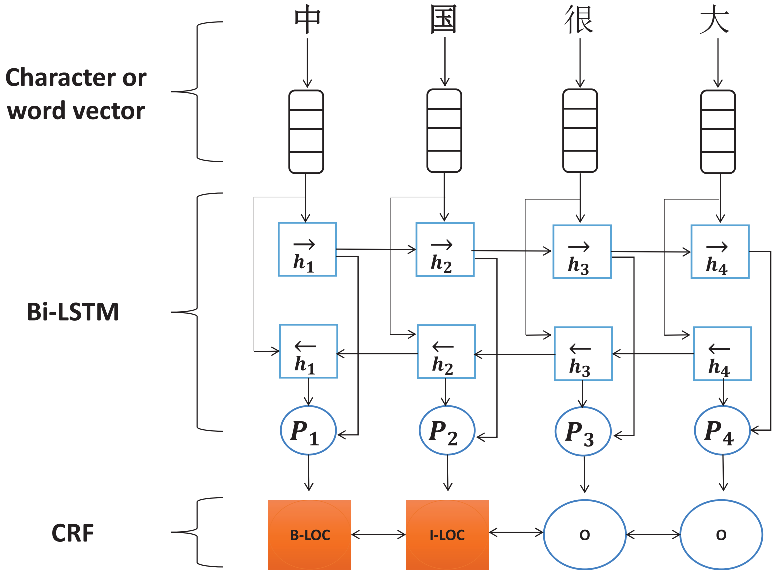


FIGURE 2. The chinese NER model of Bi-LSTM-CRF.

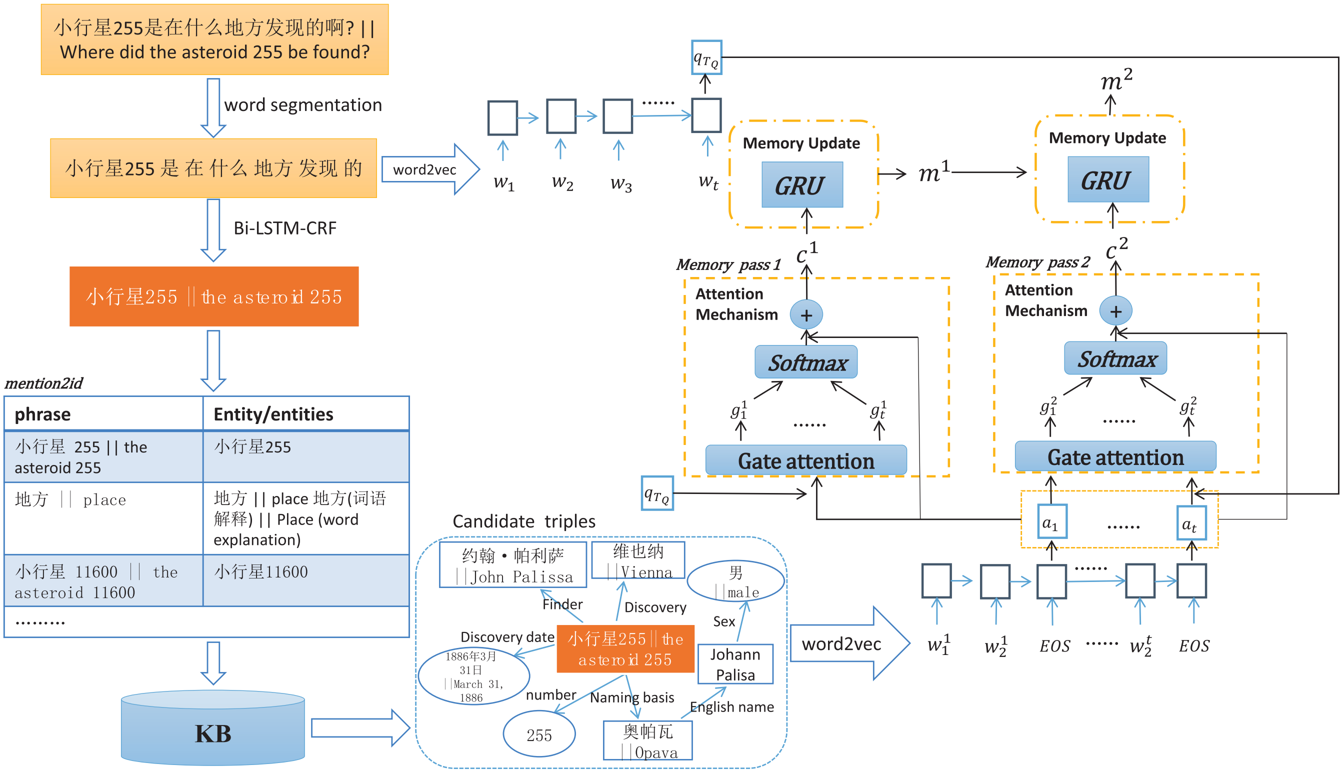
III. ANSWER ACQUISITION BASED ON DMN

As far as the automatic QA system is concerned, researchers hope that the system could have a memory mechanism similar to a human being, which might memorize the context or information of the KB. Therefore, researchers propose a number of memory models, such as the traditional deep learning RNN [23], LSTM, and the gated recurrent unit (GRU) [29]. They take the hidden states or attention mechanism as the memory function. However, a deficiency of these methods is that the generated memory is too small to meet the needs of KBQA. There are some questions in this experimental datasets that require certain reasoning to find answer. In addition, the relationship between '' 创作 || create'' and '' 写 || write'', '' 年代 || age'' and '' 时候 || time'' can be better discovered through attention mechanism. Therefore, to get more correct answers, this paper adopts the dynamic memory network (DMN) [1] model with certain reasoning ability to conduct the answer to the Chinese KBQA system. For example, when we would like to get the answer of the question ''Which dynasty is the author of the Dream of Red Chamber in?'' Firstly, DMN performs an iteration based on the question to get the relevant information: The author of ''Dream of Red Chamber'' is ''Cao Xueqin''. Updating the memory segment of model, then combing the updated memory and question to extract the supporting fact C from KB to get into the next round of iteration. Finally the answer ''Qing Dynasty'' is obtained. Hence, we apply the DMN to KBQA to get correct answers. The architecture of this model is shown in Figure 3.

In Figure 3, KBQA based on a dynamic memory network is as follows. First, given a natural question q, NER is performed on q after the word segmentation. Next, we utilize entity linking (EL) to map the entities to the KB (this paper uses mention2id to map entities) to obtain the corresponding entities in the KB and the 2-hops KB subgraph with the topic entity is extracted. Then, we use the triples that are extracted from the subgraph as inputs to the DMN. Finally, we input QA pairs into the DMN for training to get the final answer.

A. KNOWLEDGE BASE

This work uses the KB that is provided by the KBQA evaluation task in NLPCC-ICCPOL 2016 [30], which is a large-scale Chinese general KB. This KB now has more than 40 million facts, approximately 6,502,738 entities and 587,875 properties. In this KB, facts are represented by subject-property-object triples (s, p, o) and there is one triple in each line. Essentially, NLPCC-ICCPOL 2016 is a collection of multiple triples, which are collected from Baidu Encyclopedia and automatically extracted from the item infobox [31]. Therefore, there is much noise, such as irregular or useless characters and wrong attribute values. It is necessary to preprocess the KB before the experiment. The sample content of the KB is shown in Table 1.



phrase	Entity/entities
小行星 255 || the asteroid 255	小行星255
地方 || place	地方 || place 地方(词语解释) || Place (word explanation)
小行星 11600 || the asteroid 11600	小行星11600
………	

FIGURE 3. Overview of the proposed KBQA system.

TABLE 1. Parts of NLPCC-ICCPOL knowledge base.

Entities	Properties	Entities
Dream of the Red Chamber (one of the four masterpieces)	The title of book	Dream of the Red Chamber
Dream of the Red Chamber (one of the four masterpieces)	Creation time	Qing dynasty
Dream of the Red Chamber (one of the four masterpieces)	Author	Cao Xueqin

B. KNOWLEDGE BASE ENTITY MAPPING

The topic entities that are extracted from questions and linked to the KB are a significant part of the KBQA task. The first step of the system is to determine all entities that are mentioned with respect to question q, and identify which one is the topic of q. Most of the data that are used in this work are single relational facts, and each question contains only one entity. Therefore, this work uses the Bi-LSTM-CRF to determine whether the entity that is mentioned is the topic entity in q. After getting the topic entity, the next step is to map the topic entity to the corresponding entities in the KB. Concretely, we upload the topic entity into the mention2id file, and locate all possible IDs (entities) for the question. Specifically, we save all possible combinations of mention2id in a dictionary and calculate id and question similarity to find the most possible id. Then, we map them to the corresponding entities in the KB. The mention2id is a phrase-entity dictionary for EL. For instance, with the question ''Who is the author of the Stone Record? '', several possible IDs that are related to '' the Stone Record'' are obtained from the mention2id file, and then these IDs are mapped to the corresponding entities in the KB, such as, '' Dream of the Red Chamber '','' the Stone Record (jewelry brand name)'', etc. Finally, the triples within 2-hops are acquired as the input of the DMN via the knowledge extraction method. We take 4 sample triples in the KB as examples:

- Dream of the Red Chamber || author || Cao Xueqin;
- the Stone Record (jewelry brand name) || Industry || accessories;
- the Stone Record (UMG Reissue) || Number of Tracks || 10; and
- Miller Hill suspected 5 || Original Name || the Stone Record.

It can be observed that subject of the question '' the Stone Record '' is mapped to the corresponding entity of the KB using the mention2id file. Mention2id puts identical entities that distributes in different locations in KB into a dictionary, which saves time and improves efficiency in the knowledge base extraction. In addition, some cascade level errors are reduced by loading the mention2id file. Table 2 shows the parts of the mention2id about ''the Stone Record'' and its corresponding IDs .

TABLE 2. Parts of Mention2id.

the Stone Record ||| Dream of the Red Chamber , the Stone Record (jewelry brand name), the Stone Record (UMG Reissue), Miller Hill suspected 5, the Stone Record (British film, 1972), the Stone Record (Da Ming's song of the same name), the Stone Record (Da Ming Yipai album)

C. INPUT MODULE

To deal with data sparsity, word2vec [32], [33] is used to train the natural language triples a = (s, p, o) to get the vector representation $E[w_t^A]$, where E is the embedding matrix and $E[w_t^A]$ expresses the vector of the t-th word of the triple sequence. The input question vector $E\left[w_t^Q\right]$ is similar to the triple encoding step above, where $E[w_t^Q]$ represents the vector representation of the t-th word in the question. $E[w_t^A]$ and $E[w_t^Q]$ both are input sequences. The question and triples are encoded with the GRU that combines the word vectors. The GRU is used to encode the input sequences as distributed representations. At each time step t, the hidden states for the triples and question are as follows [1]:

$$a_t = GRU(E[w_t^A], a_{t-1}) \tag{1}$$

$$q_t = GRU(E[\mathrm{w}_t^Q], q_{t-1}) \tag{2}$$

a_{t-1} is the hidden state at time t for the triple representation, a_t indicates that the current hidden state is computed by the previously hidden state a_{t-1} and the current triple input $E[w_t^A]$. It is worth noting that if the input sequence is a list of triples, the triples are concatenated into a long list of word tokens, and token is inserted after each triple at the end-of-the-triple. The output of the question q is the final hidden state q_{T_Q} of the GRU encoder, where T_Q represents the given question that consists of sequences of T_Q words. Unlike the output of q, the hidden states at each of end-of- the-triple token are the final representation of the input triple sequences. Employing GRU encode to get hidden layer representations of questions and attributes.

D. MEMORY MODULE

The memory module iterates over the representations a_t and q_t is output by the input module. Meanwhile, its internal memory is updated via the output of the input module. The memory that is generated by the i-th iteration is represented as m^i. During each iteration, the attention mechanism computes a gating value g for each fact's triple representation a_t according to the question representation q_T, and g represents the degree of attention that is given to the input triples. Then, we consider g to produce an episode c for each input a_t. Finally, we take c into the GRU to generate a memory m^i. The initial state of the GRU is initialized as the question vector itself $m^0 = q$. The memory m^i is updated by the GRU, whose representation is as follows[1]:

$$m^i = GRU\left(c^i, m^{i-1}\right) \tag{3}$$

As described in the above formula, the GRU considers the episode c and previous memories m^{i-1} to update the memory m^i. This module uses a gating function as an attention mechanism. To be specific, for each iteration i, this attention mechanism takes the triple a, the previous memories m^{i-1}, and the question q_T as inputs to compute a gate value as follows[1]:

$$g_t^i = W^{(2)} \tanh\left(W^{(1)} z(a, m, q) + b^{(1)}\right) + b^{(2)} \tag{4}$$

$z(a, m, q)$ is the feature set, which is determined by calculating the similarities between the triples a, memories m, and question q. Once we get g_t^i, we will use an attention mechanism to extract a contextual vector c^i based upon the current focus. Unlike the original DMN [1] model, we use soft attention mechanism to extract a contextual vector c. After the i-th iteration, the final episode c^i was obtained, where T_C is the number of sequences of triples:

$$c^i = \sum\nolimits_{t=1}^{T_C} softmax(g_t^i) a_t \tag{5}$$

$$softmax\left(g_t^i\right) = \frac{\exp(g_t^i)}{\sum_{j=1}^{T_C} \exp(g_j^i)} \tag{6}$$

Memory m is updated by using the current episode c^i and the previous memory state, as shown in Equation 3. After the iteration is completed, the memory consists of multiple memory segments. The attention mechanism will focus on the crucial information of each memory to form recursive reasoning until the specified number of iterations is completed. Ultimately sent m to the answer module to generate answer.

E. ANSWER MODULE

The answer module is triggered once at the end of the memory. The answer module takes the question q_T, the last memory m^i as inputs to generate the model's predicted answer. DMN model is primarily for simple answers, such as a single word, and a linear layer with softmax activation can be used. But the dataset in our work contains the answer of one sentence or several phrases. Hence, RNN model can be used to decode a $= [q_T; m^i]$ into an ordered set of tokens. The cross-entropy error is used to train and propagate back through the entire network. The cross-entropy loss function J $= \alpha E_{CE}(\text{Gates}) + \beta E_{CE}(Answers)$ [1] is used as the loss for the backpropagation training through the entire network. E_{CE} is a standard cross-entropy loss function, and α and β are the hyper-parameters. Then, the correct sequence of words representing the answer is decoded from the memory to generate the final answer.

IV. EXPERIMENTS

A. DATASETS

This work uses a Chinese knowledge base, which was provided by a KBQA evaluation task from NLPCC-ICCPOL 2016[1]. The format of the triples in this KB is Subject ||| Predicate ||| Object. This KB includes 6,502,738 entities, 587,875 attributes, and 43,063,796 triples (s, p, o). After the KB is loaded via the mention2id, 19 triples are skipped, which results in 43,037,009 triples. In addition, this KB also provides a training set and a testing set. The training set contains 14,609 question answer pairs, and the testing set includes 9,870 question answer pairs. The KBQA evaluation task from NLPCC-ICCPOL 2016 also offers a file named "mention2id" that could map the entities that are mentioned in the question to the entity names in the KB. The corpus that is used in the experiment is a Chinese corpus that is crawled from Wikipedia, which contains much noise, such as article title bars, URLs, invalid characters, etc. There are some noises in answers on account of the answers of datasets are labeled manually. Before the word vector training, we first need to perform noise removal on the corpus. The details of KB cleaning are explained in Table 3. We use 'jieba' to segment the words and ensure that each line represents a document. Finally, the word vector model is trained via the word2vec tool.

TABLE 3. Knowledge base cleaning rules.

Rules	Before denoising	After denoising
Remove the appendix labels in properties	发现[1] / Discover[1]	发现 / Discover
Remove whitespace characters from properties	体 重 / Weight	体重 / Weight
Remove prefixes and suffixes between properties: '-', '•'	- 社区数 / - Number of communities	社区数/ Number of communities
Properties is the same as object	凯旋宫 ||| [1][2][3] ||| [1][2][3] / Triumph palace ||| [1][2][3] ||| [1][2][3]	Delete

B. EVALUATION METRICS

The quality of a KBQA system is generally evaluated by precision, averaged F1, MAP and accuracy@N. For entity recognition task, the accuracy, precision, recall and F1 are utilized to judge the performance of the model. Precision is defined as follows:

$$P = \left|\frac{1}{Q}\right| \sum_{i=1}^{|Q|} \left(\frac{\#(C_i A_i)}{|C_i|}\right) \tag{7}$$

where $\frac{\#(C_i A_i)}{|C_i|}$ denotes the precision for the question Q_i calculated based on the generated answer set and the correct answers A_i. $\#(C_i A_i)$ denotes the number of answers that both C_i and A_i contain, where $|C_i|$ and $|A_i|$ denote the answers number occur in C_i and A_i respectively. Similarly, the definition of recall is as follows:

$$R = \left|\frac{1}{Q}\right| \sum_{i=1}^{|Q|} \left(\frac{\#(C_i A_i)}{|A_i|}\right) \tag{8}$$

[1] http://tcci.ccf.org.cn/conference/2016/pages/page05_evadata.html

where $\frac{\#(C_iA_i)}{|A_i|}$ expresses the recall for question Q_i computed based on C_i and A_i. And the average F1 is defined as follows:

$$AverageF1 = \frac{2 \cdot P \cdot R}{P + R} \tag{9}$$

The result of answer selection is to select the candidate with the highest score, which is the top 1 answer of the model. Therefore, it is concluded that the number of correct answers to the original question is equal to the total number of answers given by the system model and the total number of questions. With the definition of precision, recall and F1 only under the top 1 answer, the value of accuracy can be obtained as follows:

$$ACC = P = R = F1 \tag{10}$$

C. EXPERIMENT SETTINGS

In the experiment, the dimension of word vectors is set to 300, and they are initialized by the pretrained word vectors that are provided by word2vec. When there is a word out of the vocabulary, we will use a randomized method to create a new 300-dimensional word vector for each unknown word. The embedding vectors are trained using the Gensim version of word2vec on the Baidu Encyclopedia corpus. We use CBOW model to train word vector, sliding window windows = 5, min_count = 5, filtering out words with fewer than 5 occurrences, setting multi-workers. When the input module performs vector coding of question and triple knowledge, the number of GRU hidden layer units is set to 100, batch_size = 100, and each batch performs 300 rounds of iteration. We use the adadelta [34] (learning rate = 0.001) rule to update the parameters to optimize the objective functions. This work uses the *L2* regularization and dropout to prevent the overfitting of the model during training to improve the performance of the model.

D. BASELINES

We compare our work to other work:

1) CRF & RULE MATCHING [40]

This work employs a combination of custom dictionary word segmentation and CRF model to identify the subject in question. Then they apply direct matching, combination predicate and word similarity to the open-domain knowledge base question and answer.

2) MULTI-GRANULARITY KBQA [41]

This model uses Bi-LSTM-CRF model to identify entities, and a multi- granularity feature representation model is proposed to perform property selection. It also utilizes character level and word level to represent questions and properties respectively. This work uses a ranking model, which leads the model to output high scores for question entities and question predicate pairs while generating lower scores for unreasonable pairings.

3) CGRU & PARAPHRASE & RANKING [42]

This model leverages the classifier to judge whether the properties in the triple are questionable and uses question-property pairs to train the classifier. They use the resource of lexical paraphrase to identify the right property. The dataset is initialized by the pre-trained word vector provided by word2vec, and the dimension of word vector is set to 100. The initial learning rate used in AdaGrad is set to 0.001. The small batch consists of 2000 question-property pairs.

E. RESULTS OF ENTITY RECOGNITION

The experiment uses Bi-LSTM-CRF to implement named entity recognition on the datasets. The bert [35] model is used to replace the part of the original word2vec training word vector. The bert utilizes google-trained Chinese bert pre-training model (chinese_L-12_H-768_A-12) for word vector training and the vector as the input for Bi-LSTM-CRF. The results are shown in Table 4.

TABLE 4. Results of named entity recognition (%).

Datasets	Acc.	Precision	Recall	F1
Training set	99.37	98.11	98.24	98.65
Testing set	99.14	94.14	95.19	94.66

During the NER model test, which found 9116 phrases, and correctly identified 8582. As can be seen from table 4, accuracy of the model can reach 99.14%, which proves that this model can identify the named entity in the question well, and lays a good foundation for the follow-up work.

F. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we discuss the experimental results of our experiments. During the experiment, our work experimented with the softmax instead of the GRU in the memory module. The original model and the modified model were tested separately. The results are presented in table 5. We can observe that this alternative setting achieved a better experimental result in table 5. The reason may be because our experiment is based on a large knowledge base background. On account of the candidate triples have some noise and similar triples. We use softmax, the candidate triplet vectors can be normalized to highlight the largest value and suppress other components well below the maximum, thereby reducing some erroneous inputs to subsequent modules. Another reason is because the softmax encourages sparsity, and so it is better suited to selecting one triple at each time. And if the softmax

TABLE 5. Results of different memory update components (%).

Memory(attention)	Acc.
GRU	76.26
Softmax	79.41

activation is spiky, it can still be differentiated by selecting a single fact for only the context vector [36]. For example, if it is assumed that at time t, $g_t^i \approx 0$, the previous state will be preserved and the triples at the current t time will be ignored. However, if $g_t^i \approx 1$, then the previous state will be forgotten, and more attention will be paid to the current input triples. It is used for subsequent input, so it is more appropriate to select one triple each time. Moreover, compared with GRU, the softmax from input to output is smooth, which is easy to compute. Therefore, among the plurality of the triples, the triples with high weights can be selected at each time, which can prevent information loss.

We also experimented with different iterations. The experimental results are listed in Table 6. The effect of the model after the second iteration is better than the number of other settings. It shows that the attention mechanism that uses two iterations is more focused than that with one iteration. We can observe that in one iteration, the attention mechanism will focus on more factual triples, which will bring much interference to the subsequent answer selection, thus affecting the accuracy of the answer selection. This is likely because with fewer iterations, the hidden state of the input module will capture more triples of adjacent time steps. Attention requires passing all captured content to the answer module at once. As a consequence, the effect of rereading is invalid, and some important information may be lost or ignored. From Table 6, we can also observe that after three iterations of the model, the experimental results have decreased compared to the second iteration. After two iterations, the attention will be more concentrated on several key pieces of information and the memory will be updated to generate new memories. Because the dataset that is provided by the NLPCC-ICCPOL 2016 KBQA evaluation task are mostly simple question answer pairs, the answer can be found in a triple relationship directly. Hence, with three and four iterations, attention has found the required answer information in the first two pieces. Therefore, the subsequent attention will become more dispersed and sometimes the attention will find the answer in the first iteration. During the experiment, when setting three and four iterations, the accuracy during the training reached 100%, but for the testset was decreased. And setting L2 regularization did not improve the results. And then after three or more iterations of the model, over-fitting occurs.

TABLE 6. Results of different iterations (%).

Iterations	Acc.
1	77.86
2	79.41
3	78.35
4	77.92

The experimental comparison results are shown in Table 7. In the context of the KB, this work has achieved good results via the dynamic memory network model, which demonstrates the effectiveness of the memory network model combined with KBQA. Compared with the results of the fourth to fifth place methods of the NLPCC-ICCPOL 2016 KBQA evaluation task, our method has achieved better results. The top two methods of the NLPCC-ICCPOL 2016 KBQA evaluation task achieved 82.47% (SPE & Pattern Rule [38]) and 81.59% (NBSVM & CNN [39]), respectively, but they both used more complex features and manual rules. On the premise of combining the dynamic memory network model and the KB, we have achieved 79.41%, and our method is more robust.

TABLE 7. Comparison of accuracy with other baselines (%).

System	Acc.
CRF & Rule Matching (2018) [41]	69.56
NEU(NLP Lab)(2016)	72.72
Multi-granularity KBQA(2016) [42]	73.96
CGRU & paraphrase & ranking(2016) [43]	79.14
Our method	79.41

This work modifies and adjusts the dynamic memory network model accordingly and applies it to obtain answers based on KBQA. However, the experimental effect of KBQA is not as good as that in reading comprehension. We analyze the reasons for the following. Since our experiment is performed on the current largest Chinese KBQA dataset, we also conduct error analysis on the dataset. We randomly extract 100 questions that our system did not generate the correct answer. The statistical results are shown in Table 8.

TABLE 8. Counts of errors on sampled data.

Cause	Counts.
Wrong entities	5
Wrong labeled answers	8
Ambiguity	21
Wrong predicts	24
Dataset caused errors	42

The KB that is provided by the NLPCC 2016 KBQA task evaluations does not fully cover the question-answer pairs in the dataset. Most importantly, there is only one standard answer for each question, but there are many entities with the same name in KB, and no ambiguity can be eliminated based on the context of the question. Similar to ''Where was Wang Jun born?'' ''What is the date of birth of Li Ming?'' there are many people who are named ''Wang Jun'' and ''Li Ming'' that are in the given KB and there is no other clue to identify to which one the question refers. As far as ''Li Ming'' is concerned, there are 108 entities named ''Li Ming'' in the KB. Therefore, it is impossible to determine which entity is correct. There are also some problems with aliases of entities. According to statistics, this situation has a high proportion in the question-answer dataset. There are 3189 training sets, accounting for 21.83% and 1584 test sets, accounting for 16.05%. This leads to a low accuracy in the evaluation results on this data set [39]. In addition, the annotations of some answers in the dataset are not consistent with the corresponding label of the given KB. For example, with ''What is the greening rate of ''Shuimu Tsinghua'', does anyone know?'' the labeled answer in the dataset is ''46.50%,'' but the answer in the knowledge base is ''46.5%''. Furthermore, there are another some problems with the dataset itself. For example, there is ambiguity between entities in the knowledge base. For ''Barack Obama'', the question-answer pairs about the entity are ''Who is Barack Obama's wife?'', and the partial triples of ''Barack Obama'' in the knowledge base are show in table 9.

TABLE 9. Triples of ''Barack Obama'' in the NLPCC-ICCPOL knowledge base.

Barack Obama(Current president of USA) ||| Alias ||| Barack Obama
Barack Obama(Current president of USA)||| Constellation ||| Leo
Barack Obama(Current president of USA) ||| Wife ||| Michelle Obama
Barack Obama(Current president of USA) ||| Father ||| Barack Hussein Obama I
Barack Obama ||| Chinese Name ||| Barack Hussein Obama
Barack Obama ||| College ||| Western College
Barack Obama ||| Wife ||| Michel Lavon Obama

From table 9, we can observe that there are multiple entities of ''Barack Obama'' in the knowledge base (some of the triples are presented in table 9) which may be because of the fusion of multiple data sources. Therefore, we cannot fully guarantee the alignment of the information. We find that Barack Obama's wife has ''Michelle Obama'' and ''Michelle Lavon Obama'', and the answer that is given in our question-answer pairs is ''Michel Obama''. Thus, when our model retrieves the correct triple, the final answer may still be judged as an error. In addition, for the question ''When was the Dr. to Worship written?'' the corresponding answer in dataset is ''1461''. Some of the triples that contain ''Dr. to Worship'' are shown in Table 10.

TABLE 10. Triples of ''Dr. to Worship'' in the NLPCC-ICCPOL knowledge base.

Dr. to Worship (Dieric Bouts the Elder Painting) ||| Age ||| 1445
Dr. to Worship (Domenico Ghirlandaio Painting)||| Creation Time ||| 1487
Dr. to Worship (Fabriano Painting) ||| Age ||| 1423
Dr. to Worship (Tiepolo Ciovanni Battista Painting) ||| Creation Age || 1753
Dr. to Worship (Mantegna Painting) ||| Age ||| 1461

The ''Dr. to Worship'' was created by many painters at different times. We cannot get the painter's creation date from the current question. There is ambiguity in the entity of the question. As a result, even if the correct answer is retrieved, the experimental model can still make a wrong judgment. Due to the large number of related entity triples in the knowledge base, it is also a challenge to the effectiveness and efficiency of the retrieval model. In addition, there are some similar questions in the dataset, which are prone to over-fitting and are not conducive to the training of the model. We leave all these issues to our future work.

V. CONCLUSION

With the development of AI and IoT technology, Chinese intelligent QA systems have been widely applied to all aspects of our daily life. The demand for information retrieval will continue to increase, and people's requirements for the accuracy of information retrieval are also rising. Therefore, research on intelligent Chinese QA systems has become one of the hotspots of current research. In our work, the dynamic memory network model is applied to the KBQA system, so that KBQA system has its own memory and reasoning judgment according to the given questions. Different from the traditional KBQA method (template or rule) for the structured query, we use representation learning to express the questions and related KB subgraph and take them as input to the dynamic memory network. Hence, the answer is obtained via its memory and reasoning capabilities. Finally, this method has achieved a good result on the KBQA task in NLPCC-ICCPOL 2016. Due to the lack of Chinese corpora and datasets, there is much noise and a large amount of nonlogical and erroneous data in the dataset. This will have a certain impact on the experimental results. In future work, expanding and perfecting the corpus and dataset approach is considered. Meanwhile, the entities will be combined in the next research, and whether the experimental results can be improved will be the focus of the next step.

REFERENCES

[1] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, ''Ask me anything: Dynamic memory networks for natural language processing,'' in *Proc. 33rd Int. Conf. Int. Conf. Mach. Learn. (ICML)*, New York, NY, USA, Jun. 2016, pp. 1378–1387.

[2] M. F. Suchanek, G. Kasneci, and G. Weikum, ''Yago: A core of semantic knowledge,'' in *Proc. 16th Int. Conf. World Wide Web*, Banff, AB, Canada, May 2007, pp. 697–706

[3] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, ''Freebase: A collaboratively created graph database for structuring human knowledge,'' in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2008, pp. 1247–1250.

[4] J. Lehmann, ''DBpedia—A large-scale, multilingual knowledge base extracted from Wikipedia,'' *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015.

[5] J. Chen *et al.*, ''CN-Probase: A data-driven approach for large-scale Chinese taxonomy construction,'' in *Proc. ICDE*, 2019, pp. 1706–1709. [Online]. Available: https://arxiv.org/pdf/1902.10326.pdf?

[6] B. Xu, Y. Xu, J. Liang, C. Xie, B. Liang, W. Cui, and Y. Xiao, ''CN-DBpedia: A never-ending chinese knowledge extraction system,'' in *Proc. Int. Conf. Ind., Eng. Other Appl. Appl. Intell. Syst.* Cham, Switzerland: Springer, Jun. 2017, pp. 428–438.

[7] Z. Y. Liu, ''Knowledge representation learning—A review,'' *J. Comput. Res. Develop.*, vol. 53, no. 2, pp. 247–261, 2016.

[8] K. Liu, Z. Yuan-Zhe, J. Guo-Liang, L. Si-Wei, and Z. Jun, ''Representation learning for question answering over knowledge base: An Overview,'' *Acta Automatica Sinica*, vol. 42, no. 6, pp. 807–818, 2016.

[9] W. Yih, X. He, and C. Meek, ''Semantic parsing for single-relation question answering,'' in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, Jun. 2014, pp. 643–648.

[10] L. Dong, F. Wei, M. Zhou, and K. Xu, ‘‘Question answering over freebase with multi-column convolutional neural networks,’’ in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, Beijing, China, Jul. 2015, pp. 260–269.

[11] A. Bordes, N. Usunier, S. Chopra, and J. Weston, ‘‘Large-scale simple question answering with memory networks,’’ 2015, *arXiv:1506.02075*. [Online]. Available: https://arxiv.org/abs/1506.02075

[12] A. Fader, L. Zettlemoyer, and O. Etzioni, ‘‘Open question answering over curated and extracted knowledge bases,’’ in *Proc. KDD*, Aug. 2014, pp. 1156–1165.

[13] Abujabal, Abdalghani, ‘‘Automated template generation for question answering over knowledge graphs,’’ in *Proc. Int. Conf. World Wide Web*, Apr. 2017, pp. 1191–1200.

[14] Y. Bengio, A. Courville, and P. Vincent, ‘‘Representation learning: A review and new perspectives,’’ *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[15] A. Bordes, J. Weston, and N. Usunier, ‘‘Open question answering with weakly supervised embedding models,’’ *CoRR*, vol. abs/1404.4326, pp. 165–180, Apr. 2014.

[16] A. Bordes, S. Chopra, and J. Weston, ‘‘Question answering with subgraph embeddings,’’ *CoRR*, vol. abs/1406.3676, pp. 615–620, Sep. 2014.

[17] W. Chen, W. Xiong, X. Yan, and W. Y. Wang, ‘‘Variational knowledge graph reasoning,’’ in *Proc. 32nd AAAI Conf. Artif. Intell.*, Jun. 2018, pp. 1823–1832.

[18] B. Wang, K. Liu, and J. Zhao, ‘‘Inner attention based recurrent neural networks for answer selection,’’ in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, Aug. 2016, pp. 1288–1297.

[19] Y. Zhang, K. Liu, S. He, G. Ji, Z. Liu, H. Wu, and J. Zhao, ‘‘Question answering over knowledge base with neural attention combining global knowledge information,’’ 2016, *arXiv:1606.00979*. [Online]. Available: https://arxiv.org/abs/1606.00979

[20] M. Tan, C. N. dos Santos, B. Xiang, and B. Zhou, ‘‘Improved representation learning for question answer matching,’’ in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Aug. 2016, pp. 464–473.

[21] M. Schuster and K. K. Paliwal, ‘‘Bidirectional recurrent neural networks,’’ *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[22] B. T. Zhou, ‘‘LSTM based question answering for large scale knowledge base,’’ *Beijing Da Xue Xue Bao*, vol. 54, no. 2, pp. 286–292, 2018.

[23] S. Grossberg, ‘‘Recurrent neural networks,’’ *Scholarpedia*, vol. 8, no. 2, p. 1888, 2013.

[24] Y. Lai, Y. Lin, J. Chen, Y. Feng, and D. Zhao, ‘‘Open domain question answering system based on knowledge base,’’ in *Proc. Int. Conf. Comput. Process. Oriental Lang. (ICCPOL)*, vol. 10102, 2016, pp. 722–733, doi: 10.1007/978-3-319-50496-4_65.

[25] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, ‘‘Neural architectures for named entity recognition,’’ in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Jun. 2016, pp. 260–270.

[26] C. Dong, J. Zhang, C. Zong, M. Hattori, and H. Di, ‘‘Character-based LSTM-CRF with radical-level features for Chinese named entity recognition,’’ in *Proc. Int. Conf. Comput. Process. Oriental Lang. (ICCPOL)*, vol. 10102, Dec. 2016, pp. 239–250.

[27] Z. Huang, W. Xu, and K. Yu, ‘‘Bidirectional LSTM-CRF models for sequence tagging,’’ 2015, *arXiv:1508.01991*. [Online]. Available: https://arxiv.org/abs/1508.01991

[28] L. Sun, C. Zong, M. Zhang, and G.-A. Levow, Eds., *Proceedings of the Third CIPS-SIGHAN Joint conference on Chinese language processing*. Wuhan, China: Association for Computational Linguistics, 2014, p. 223.

[29] K. Cho, ‘‘On the properties of neural machine translation: Encoder–decoder approaches,’’ in *Proc. 8th Workshop Syntax, Semantics Struct. Stat. Transl.*, Oct. 2014, pp. 103–111.

[30] N. Duan, ‘‘Overview of the NLPCC-ICCPOL 2016 shared task: Open domain Chinese question answering,’’ in *Proc. Int. Conf. Comput. Process. Oriental Lang.*, Cham, Switzerland: Springer, Dec. 2016, pp. 942–948.

[31] C. Shen, T. Huang, X. Liang, F. Li, and K. Fu, ‘‘Chinese knowledge base question answering by attention-based multi-granularity model,’’ *Information*, vol. 9, no. 4, p. 98, Apr. 2018.

[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, ‘‘Efficient estimation of word representations in vector space,’’ 2013, *arXiv:1301.3781*. [Online]. Available: https://arxiv.org/abs/1301.3781

[33] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, ‘‘Distributed representations of words and phrases and their compositionality,’’ in *Proc. NIPS*. New York, NY, USA: Curran Associates, Inc., 2013, pp. 3111–3119.

[34] M. D. Zeiler, ‘‘ADADELTA: An adaptive learning rate method,’’ 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701

[35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ‘‘BERT: Pre-training of deep bidirectional transformers for language understanding,’’ 2018, *arXiv:1810.04805*. [Online]. Available: https://arxiv.org/abs/1810.04805

[36] C. Xiong, S. Merity, and R. Socher, ‘‘Dynamic memory networks for visual and textual question answering,’’ in *Proc. ICML*, vol. 48, 2016, pp. 2397–2406.

[37] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, ‘‘End-to-end memory networks,’’ in *Proc. NIPS*, 2015, pp. 2440–2448.

[38] Y. Lai, Y. Lin, J. Chen, Y. Feng, and D. Zhao, ‘‘Open domain question answering system based on knowledge base,’’ in *Natural Language Understanding and Intelligent Applications*. Cham, Switzerland: Springer, 2016, pp. 722–733.

[39] F. Yang, L. Gan, A. Li, D. Huang, X. Chou, and H. Liu, ‘‘Combining deep learning with information retrieval for question answering,’’ in *Natural Language Understanding and Intelligent Applications*. Cham, Switzerland: Springer, 2016, pp. 917–925.

[40] Z. Tao, J. Zhen, and L. Tianrui, ‘‘Open-domain question-answering system based on large-scale knowledge base,’’ *CAAI Trans. Intell. Syst.*, vol. 13, no. 4, pp. 557–563, 2018.

[41] S. Cun, H. Tinglei, and X. Liang, ‘‘Knowledge graph question answering based on multi-granularity feature representation,’’ in *Proc. 12th ACM Int. Conf. Web Search Data Mining*, vol. 9, Feb. 2019, pp. 105–113.

[42] L. Wang, Y. Zhang, and T. Liu, ‘‘A deep learning approach for question answering over knowledge base,’’ in *Proc. 24th Int. Conf. Comput. Process. Oriental Lang.*, 2016, pp. 885–892.

LEI SU is currently an Assistant Professor with the School of Information and Automation, Kunming University of Science and Technology, China. His current research interests include machine learning and information retrieval.

TING HE is currently pursuing the master's degree with the Kunming University of Science and Technology, China. Her current research interests include machine learning and question answering.

ZHENGYU FAN is currently pursuing the master's degree with the Kunming University of Science and Technology, China. His current research interests include machine learning and question answering.

YIN ZHANG (SM'16) is currently an Associate Professor with the School of Information and Safety Engineering, Zhongnan University of Economics and Law (ZUEL), China. He is also a Wenlan Distinguished Scholar with ZUEL and a Chutian Distinguished Scholar, Hubei, China. He has published more than 80 prestigious conference and journal articles, including eight ESI highly cited articles. His current research interests include intelligent service computing, big data, and social networks. He received the IEEE Systems Journal Best Paper Award from the IEEE Systems Council, in 2018. He also served as a Track Chair of IEEE CSCN 2017 and a TPC Co-Chair of CloudComp 2015 and TRIDENTCOM 2017. He is a Vice-Chair of the IEEE Computer Society Big Data STC. He serves as an Editor or an Associate Editor for the IEEE Network, IEEE Access, and the *Journal of Information Processing Systems*. He is a Guest Editor of *Future Generation Computer Systems*, the IEEE Internet of Things Journal, ACM/Springer *Mobile Networks & Applications*, *Sensors*, *Neural Computing and Applications*, *Multimedia Tools and Applications*, *Wireless Communications and Mobile Computing*, *Electronic Markets*, *Journal of Medical Systems*, and *New Review of Hypermedia and Multimedia*.

MOHSEN GUIZANI (S'85–M'89–SM'99–F'09) received the B.S. (with distinction) and M.S. degrees in electrical engineering, and the M.S. and Ph.D. degrees in computer engineering from Syracuse University, Syracuse, NY, USA, in 1984, 1986, 1987, and 1990, respectively. He served in different academic and administrative positions with the University of Idaho, Western Michigan University, the University of West Florida, the University of Missouri-Kansas City, the University of Colorado-Boulder, and Syracuse University. He is currently a Professor with the Department of Computer Science and Engineering, Qatar University. He is the author of nine books and more than 500 publications in refereed journals and conferences. His current research interests include wireless communications and mobile computing, computer networks, mobile cloud computing, security, and smart grid. He is a Senior Member of ACM. Throughout his career, he received three teaching awards and four research awards. He also received the 2017 IEEE Communications Society WTC Recognition Award and the 2018 Ad Hoc Technical Committee Recognition Award for his contribution to outstanding research in wireless communications and ad hoc sensor networks. He has also served as a TPC member, Chair, and the General Chair of a number of international conferences. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the Chair of the TAOS Technical Committee. He is currently serves on the editorial boards of several international technical journals, and the Founder and the Editor-in-Chief of the IEEE Network *Wireless Communications and Mobile Computing Journal* (Wiley). He has guest edited a number of special issues in the IEEE journals and magazines. He served as the IEEE Computer Society Distinguished Speaker. He is currently an IEEE ComSoc Distinguished Lecturer.