

# A multi-attention RNN-based relation linking approach for question answering over knowledge base

Li Huiying Zhao Man Yu Wenqi

(School of Computer Science and Engineering, Southeast University, Nanjing 211189, China)

**Abstract:** Aiming at the relation linking task for question answering over knowledge base, especially the multi relation linking task for complex questions, a relation linking approach based on the multi-attention recurrent neural network (RNN) model is proposed, which works for both simple and complex questions. First, the vector representations of questions are learned by the bidirectional long short-term memory (Bi-LSTM) model at the word and character levels, and named entities in questions are labeled by the conditional random field (CRF) model. Candidate entities are generated based on a dictionary, the disambiguation of candidate entities is realized based on predefined rules, and named entities mentioned in questions are linked to entities in knowledge base. Next, questions are classified into simple or complex questions by the machine learning method. Starting from the identified entities, for simple questions, one-hop relations are collected in the knowledge base as candidate relations; for complex questions, two-hop relations are collected as candidates. Finally, the multi-attention Bi-LSTM model is used to encode questions and candidate relations, compare their similarity, and return the candidate relation with the highest similarity as the result of relation linking. It is worth noting that the Bi-LSTM model with one attentions is adopted for simple questions, and the Bi-LSTM model with two attentions is adopted for complex questions. The experimental results show that, based on the effective entity linking method, the Bi-LSTM model with the attention mechanism improves the relation linking effectiveness of both simple and complex questions, which outperforms the existing relation linking methods based on graph algorithm or linguistics understanding.

**Key words:** question answering over knowledge base (KBQA); entity linking; relation linking; multi-attention bidirectional long short-term memory (Bi-LSTM); large-scale complex question answering dataset (LC-QuAD)

**DOI:** 10.3969/j.issn.1003-7985.2020.04.003

Question answering over knowledge base (KBQA) is an active research area that allows answers to be ob-

tained from knowledge base based on natural language input. Some KBQA systems have been intensively studied for simple questions, which can be answered by one fact in the knowledge base. Even though this task is called “simple”, it is actually not simple at all and is far from being solved perfectly. Moreover, the demand for complex question answering has increased quickly. Answering a complex question can require references to multiple related facts in the knowledge base.

An important line of KBQA research is the step-by-step pipeline method based on semantic parsing. Singh et al.<sup>[1]</sup> proposed a KBQA component framework to dynamically choose components to create a complete question answering pipeline. A KBQA pipeline consists of three key components: 1) Entity linking, which links  $n$ -grams to knowledge base entities; 2) Relation linking, which identifies the knowledge base relation(s) to which a question refers; and 3) Query building, which builds the structured query to obtain answers from the knowledge base.

Relation linking, linking the extracted relations from the input question to their knowledge base occurrences, is a core component of the KBQA pipeline. In this work, we propose a multi-attention recurrent neural network (RNN)-based relation linking approach that can deal with relation linking problems for both simple and complex questions. For the input question, entity linking, the basic component of the KBQA pipeline, recognizes the entity mentions and links them to the corresponding entities in the knowledge base first. Then, a classifier is used to judge whether the question is complex or not. Based on the judgement, different relation linking models are used. For a simple question, a one-attention bidirectional long short-term memory (Bi-LSTM) model is applied to detect one relation from the question. For a complex question, a two-attention Bi-LSTM model is applied to detect two relations from the question.

## 1 Related Work

Given a natural language question, entity linking and relation linking are the core tasks for question answering. There is a wide range of tools and research work in the area of entity linking. Mostly, research in this domain targets news corpora, documents and Wikipedia abstracts with long sentences. Based on a vector-space representation of entities, DBpedia Spotlight<sup>[2]</sup> uses the cosine simi-

Received 2020-05-05, Revised 2020-09-16.

**Biography:** Li Huiying (1977—), female, doctor, associate professor, huiyingli@seu.edu.cn.

**Foundation item:** The National Natural Science Foundation of China (No. 61502095).

**Citation:** Li Huiying, Zhao Man, Yu Wenqi. A multi-attention RNN-based relation linking approach for question answering over knowledge base[J]. Journal of Southeast University (English Edition), 2020, 36 (4): 385 – 392. DOI: 10.3969/j.issn.1003-7985.2020.04.003.

larity to rank candidate entities. AIDA<sup>[3]</sup> builds a coherence graph and applies dense subgraph algorithms for entity linking. Babelfy<sup>[4]</sup> uses the random walk and the densest subgraph algorithm to tackle the word sense disambiguation and entity linking tasks jointly. FOX<sup>[5]</sup> deals with named entity recognition based on ensemble learning. However, when these tools are applied to short text in a new domain such as question answering, the performance is limited. Considering short text, TAGME<sup>[6]</sup> recognizes named entities by matching terms with Wikipedia link texts and disambiguates the match using the in-link graph and the page dataset. KEA<sup>[7]</sup> begins with the detection of groups of  $n$ -grams and a lookup of all potential candidate entities for each  $n$ -gram. The disambiguation of candidate entities is based on local and context-related features.

Relation linking is a relatively new research area compared to entity linking. SIBKB<sup>[8]</sup> represents a background knowledge base as a bi-partite and a dynamic index over the relation patterns included in the knowledge base. A relation linking component is proposed based on the semantic index. With the recent progress of deep learning, neural network (NN)-based methods have been introduced to the relation linking task. NN-based methods represent both the questions and the relations as semantic vectors. Then, the relation linking process can be converted into a similarity matching process between an input question and its candidate relations in a semantic space. The candidates with the highest similarity score will be selected as the final relations. The main difference among these approaches lies in the encoder model and the input granularity. The encoder model can be an RNN model or a convolutional neural network (CNN) model. The input granularity can be word level or character level granularity. ReMatch<sup>[9]</sup> characterizes both the properties in knowledge and the relations in a question as comparable triples, then leverages both synonyms and semantic similarity measures based on graph distances from Wordnet (<https://wordnet.princeton.edu/>). Yu et al.<sup>[10]</sup> proposed a hierarchical RNN enhanced by residual learning for rela-

tion detection. Both EARL<sup>[11]</sup> and Falcon<sup>[12]</sup> perform joint entity and relation linking. EARL implements two different solution strategies. The first strategy formalizes the problem as an instance of the generalized traveling salesman (GTSP) problem and solves it with an approximation algorithm. The second strategy uses a machine learning method in order to exploit the connection density between nodes in the knowledge graph. Falcon performs joint entity and relation linking of a short text by using a light-weight linguistic approach. It utilizes an extended knowledge graph created by merging entities and relations from various knowledge sources for candidate generation. Specifically, it leverages several fundamental principles of English morphology (e. g. compounding, headword identification) for candidate ranking.

Although there has been some research on entity and relation linking, solving the task of relation linking is still very challenging, especially the relation linking task for complex questions.

2 Method

In this section, we introduce the overall architecture of our approach, followed by our dictionary-based entity linking approach and the multi-attention RNN based relation linking approach.

2.1 Overview

The general framework of the proposed approach is shown in Fig. 1. Let us consider an example to explain the underlying idea: How many developers make software for Unix-like operating systems? The entity linking component is expected to recognize the mention “Unix-like” as a named entity and link it to the corresponding entity “dbr: Unix-like” in the knowledge base. The relation linking component classifies the question into simple and complex categories. For a simple question, a one-attention Bi-LSTM is applied to detect one relation from the question. For a complex question, a two-attention Bi-LSTM is applied to detect two relations from the question.

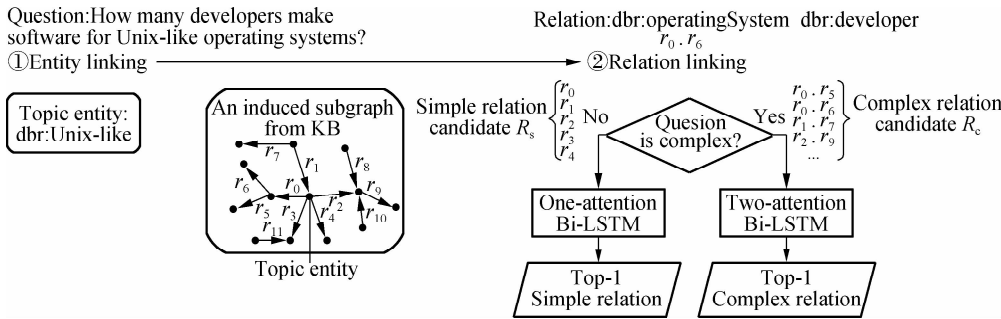


Fig. 1 Overall architecture of the proposed approach

2.2 Entity linking

Entity linking is the first component of the QA pipe-

line, and the results greatly affect the following components. In general, entity linking is a two-step process to obtain the topic entity from the question. The first step is

to recognize the named entity mention. The second step is to disambiguate or link the entity to the knowledge base. The general framework of the proposed entity linking approach is shown in Fig. 2.

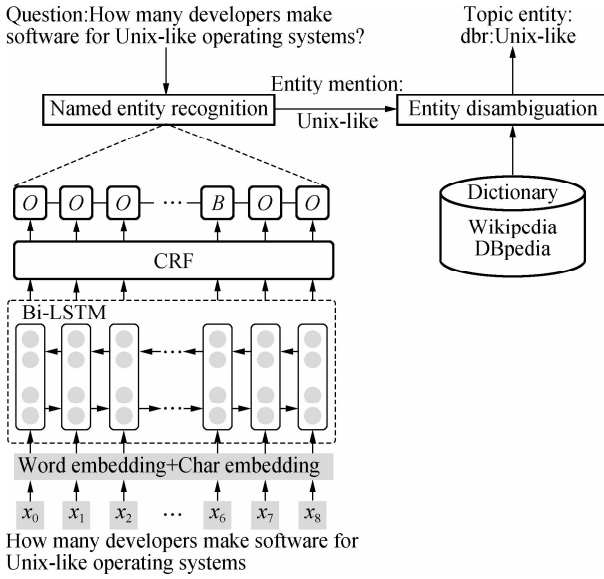


Fig. 2 Architecture of the entity linking

For named entity recognition, we regard it as a sequence annotation task. We encode the question at the word and character levels to provide additional advantages in dealing with unknown or out-of-vocabulary (OOV) words. When a question sequence comes, the pre-trained word embedding and character embedding characters are retrieved to construct a matrix. Then, the Bi-LSTM is applied to generate an abstract representation of the question. Finally, an additional conditional random field (CRF) tagging model is used to detect the named entity mention. For the labeling of the mentions, the BIO tagging scheme (B = beginning of an entity, I = inside of an entity, O = outside of an entity) is used.

For entity disambiguation, we collect a dictionary on the surface forms of entities from entity pages, redirect pages, and disambiguation pages in Wikipedia. We can then generate candidate entities for the given mention on the basis of this dictionary and map these candidate entities in Wikipedia to the entities in the DBpedia knowledge base. For candidate entity ranking, we consider the edit distance and a few linguistic-based rules.

### 2.3 Relation linking

First, a classifier is used to judge whether the question is complex or not. Secondly, simple or complex relation candidates are obtained from the generated subgraph according to the complexity of the question. Finally, different attentive RNN models are applied to encode the given question and relation candidates and score the similarity between them.

The aim of the simple/complex classification is to

build a binary classifier to efficiently predict whether a given question is simple or complex. The input of the classifier is the features of the question, and it contains the length of the question, the number of entities, and word POS tagging. After testing eight classifiers, we select logistic regression which performs outstandingly well to classify the complexity of the question.

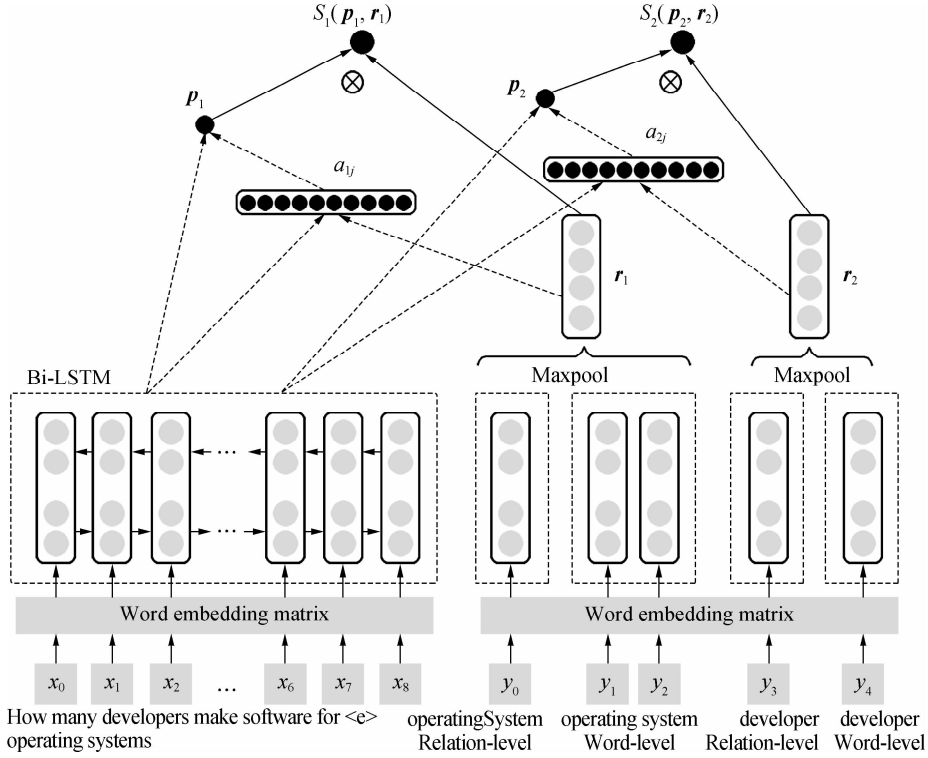
For each question  $q$ , we use the entity linking component to identify the topic entity, which can be simply regarded as the main entity of the question. After obtaining the topic entity, we collect its one-hop or two-hop relations according to the classification of the question. For a simple question, one-hop relations are collected to constitute a simple relation candidate set  $R_s$ . For a complex question, two-hop relations are collected to constitute a complex relation candidate set  $R_c$ .

According to the complexity of the question, we adopt different multi-attention Bi-LSTM models to detect relations. For a simple question, the one-attention Bi-LSTM model is applied to detect one relation from the question. For a complex question, the two-attention Bi-LSTM model is applied to detect two relations from the question.

We take the complex question as an example to explain the process. The topic entity mention is replaced in the question with a token " $\langle e \rangle$ " as in question pattern  $P$ . It helps the model reduce the effect of noise. We use the two-attention Bi-LSTM model (see Fig. 3) to score the similarity between  $P$  and each two-hop relation candidate  $r$  in  $R_c$ .

For each relation  $r$  in  $R_c$ , we describe it from two levels, namely, relation level and word level, and transform each part to its trainable embedding to obtain their vector representations. The running question "How many developers make software for Unix-like operating systems?" is classified as a complex one. Hence, the relation candidate set  $R_c$  contains all two-hop relations connected to the topic entity. Take the two-hop candidate relation "operatingSystem developer" for instance. We deal with the relations "operatingSystem" and "developer" separately and feed them into the two-attention Bi-LSTM model. For each input relation, we describe it from the relation and word levels. For example, the relation "operatingSystem" can be divided into "operating" and "system" at the word level. The relation level of the single word "developer" is the same as the word level. The word embedding is fed into the Bi-LSTM network. Then, we obtain  $r_i (i \in \{1, 2\})$  by max-pooling the relation and word levels, respectively.  $r_1$  represents the first relation aspect in relation  $r$ , and  $r_2$  represents the second relation aspect.

Each word in the question is initialized to its word embedding. Then, the embedding is fed into the Bi-LSTM network to obtain the hidden representations,  $H = [h_1, \dots, h_L]$ .  $L$  is the length of the question. Each vector  $h_j$  is



**Fig. 3** Architecture of the two-attention Bi-LSTM model

the concatenation between forward/backward representations at time  $j$ . We adopt the additive attention mechanism proposed in Ref. [13]. Each aspect  $r_i (i \in \{1, 2\})$  of the relation pays different attention to the question and decides how the question is represented. The extent  $\alpha_{ij}$  of the attention is used as the weight of each word in the question. Thus, for different relation aspects  $r_i$ , the corresponding question pattern representation  $p_i$  is calculated as

$$p_i = \sum_{j=1}^L \alpha_{ij} h_j \quad (1)$$

$$\alpha_{ij} = \frac{\exp(w_{ij})}{\sum_{k=1}^L \exp(w_{ik})} \quad (2)$$

$$w_{ij} = v^T \tanh(W^T [h_j; r_i] + b) \quad (3)$$

where  $\alpha_{ij}$  denotes the attention weight of the  $j$ -th word in the question in terms of relation aspect  $r_i$ . Let the dimensions of  $r_i$  and  $h_j$  be  $m$  and  $n$ , respectively. Then,  $W$  and  $v$  are the parameters to be learned with  $W \in \mathbf{R}^{c \times (m+n)}$ ,  $v \in \mathbf{R}^{1 \times c}$  (where  $c$  is a hyper parameter).  $b$  is the offset.

At this point, we have the representations of question pattern  $p_i$  and relation aspect  $r_i$ . Their similarity is calculated by the following equation:

$$S_i(p_i, r_i) = p_i \otimes r_i \quad i = 1, 2 \quad (4)$$

The operation  $\otimes$  here is the dot product of two vectors.

We compute a matching score  $S(P, r)$  that represents the similarity between the question pattern and the candi-

date relation.

$$S(P, r) = \text{sigmoid}(W^T [S_1(p_1, r_1); S_2(p_2, r_2)] + b) \quad (5)$$

For all  $r \in R_c$ , the final prediction is  $\hat{r}$  with

$$\hat{r} = \underset{r \in R_c}{\operatorname{argmax}} (S(P, r)) \quad (6)$$

The model described above is trained with a ranking loss to maximize the margin between gold relation  $r^+$  and other relations  $r^-$  in the candidate set:

$$\text{loss}(P, r^+, r^-) = \sum_{(P, r^-) \in D} \max(0, \gamma + S(P, r^-) - S(P, r^+)) \quad (7)$$

where  $\gamma$  is a constant parameter.

### 3 Experimental Results

We use the LC-QuAD<sup>[14]</sup> question dataset in our experiments. LC-QuAD (<https://figshare.com/projects/LC-QuAD/21812>) contains 5 000 questions (4 000 for training and 1 000 for testing) with their intended SPARQL queries for DBpedia. DBpedia is an encyclopedic knowledge base, and the types of entities mentioned in the questions for DBpedia exhibit large variations. There are 5 042 entities, 615 relations, and multifarious entity types (for example, book, television show, company, and politician) in the dataset. We supplement the dataset with the entity mention and relation mention annotations (<https://github.com/wds-seu/KBQA-multi-attention-RNN>) to benefit the evaluation of each KBQA pipeline step. All reported experiments are run on the DBpedia 2016-04 version.

### 3.1 Results

We report and analyze the experimental results of the entity linking, relation linking components separately to show their effectiveness.

To evaluate the proposed entity linking component, we compare it against the state-of-the-art tools. We report the experimental results of the systems integrated in Gerbil<sup>[15]</sup> i.e., KEA<sup>[7]</sup>, FOX<sup>[5]</sup>, Babelfy<sup>[4]</sup>, and AIDA<sup>[3]</sup>. Gerbil is a benchmarking framework for entity linking systems. These methods deal with entity linking task based on the graph algorithm or context-related features. We also report the performance of TAGME<sup>[6]</sup>, DBpedia Spotlight<sup>[2]</sup>, EARL<sup>[11]</sup> and Falcon<sup>[12]</sup> for entity linking. Most of these approaches (except Falcon) use state of the art machine learning techniques. It is important to note that Falcon utilizes an extended knowledge graph created by merging entities and relations from various knowledge sources for candidate generation.

Tab.1 lists the macro precision, macro recall, and macro  $F_1$  of all systems on the LC-QuAD dataset. The performance of the compared systems listed in Tab. 1 is obtained directly from Ref. [12]. The word embedding in our approach is initialized using two pre-trained word vectors, GloVe<sup>[16]</sup> and BERT<sup>[17]</sup>. GloVe is an unsupervised learning algorithm for obtaining vector representations for words. BERT, a pre-trained transformer network, is designed to pre-train deep bidirectional representations from unlabeled text. The results demonstrate that Falcon and our approach significantly outperform other systems. Our approach, whether initialized by GloVe or BERT, is superior to other methods in precision. When initializing with BERT, it performs better in both precision and  $F_1$  value. The reason for Falcon's better recall value is that it uses more external knowledge sources to generate candidates.

**Tab.1** Macro precision, recall, and  $F_1$  values of entity linking approaches

Approach	Precision	Recall	$F_1$
KEA	0.001	0.001	0.001
FOX	0.53	0.51	0.51
Babelfy	0.43	0.50	0.44
AIDA	0.50	0.45	0.47
DBpedia Spotlight	0.60	0.65	0.61
TAGME	0.65	0.77	0.68
EARL	0.53	0.55	0.53
Falcon	0.81	<b>0.86</b>	0.83
Our GloVe-based approach	0.83	0.81	0.81
Our BERT-based approach	<b>0.87</b>	0.85	<b>0.86</b>

In the process of relation linking, we evaluate different models to prove that the one-attention Bi-LSTM model is applicable to simple questions and that the two-attention Bi-LSTM model is applicable to complex questions. Among the 1 000 testing questions in LC-QuAD, there

are 366 simple and 634 complex questions, respectively. On the basis of the assumption that the simple/complex classification is completely correct, we evaluate the Bi-LSTM model without attention and with one attention separately on the simple question dataset. We also evaluate the Bi-LSTM model without attention, with one attention, and with two attentions on the complex question dataset. We set the embedding dimension to 300 in training process, and all word embedding is initialized using the pre-trained GloVe here. The comparison results are shown in Tab.2.

**Tab.2** Macro precision, recall, and  $F_1$  values of different models on simple and complex LC-QuAD datasets

Dataset	Model	Precision	Recall	$F_1$
Simple (366)	Without attention Bi-LSTM	0.51	0.51	0.51
	One-attention Bi-LSTM	<b>0.52</b>	<b>0.52</b>	<b>0.52</b>
Complex (634)	Without attention Bi-LSTM	0.50	0.50	0.50
	One-attention Bi-LSTM	0.49	0.49	0.49
	Two-attention Bi-LSTM	<b>0.52</b>	<b>0.52</b>	<b>0.52</b>

We observe that the results with attention are better than those without attention for simple questions. Hence, adopting the attention mechanism in the Bi-LSTM model can improve performance. For complex questions, the results of the two-attention Bi-LSTM model that separates two relations are better than those of the one-attention Bi-LSTM model or the model without attention. It means that, merging two aspects of the relation into one attention or without attention is not appropriate for complex questions.

To illustrate the effectiveness of the attention mechanism clearly, we visualize the attention distribution of the question words in the question encoding process ( see Fig.4). Color depth represents the attention weights, a darker color indicates a higher attention weight. From the simple question example, we can observe that the one-attention model is able to capture the attention properly. For the complex question, each question word learns a different weight for different relation aspects.

To evaluate our relation linking component, we apply the multi-attention Bi-LSTM model on the basis of the results of our simple/complex classification. The results of our proposed relation linking approach in comparison with four baselines are listed in Tab. 3. These baselines are SIBKB<sup>[8]</sup>, ReMatch<sup>[9]</sup>, and the recently released EARL and Falcon systems. Both SIBKB and ReMatch leverage semantic similarity measures to perform relation linking. EARL applies both the graph algorithm and machine learning for relation linking. Falcon utilizes an extended knowledge graph and a linguistic understanding method.

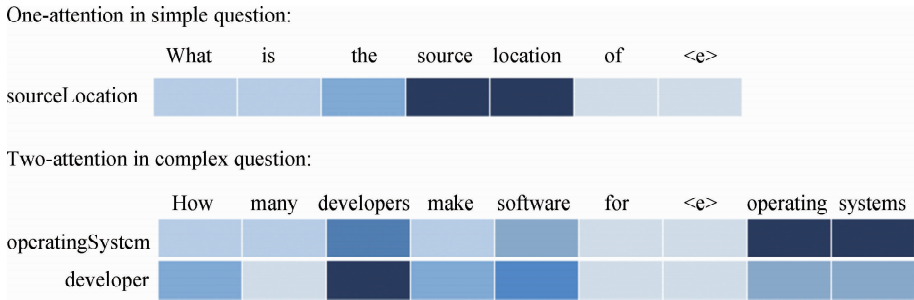


Fig. 4 Visualized attention heat map

Our evaluation metrics are the same as those of Falcon. We list two experimental results of our relation linking component. Our RL (golden EL) employs our relation linking component on the basis of the golden entity linking outputs. Our RL (our EL) employs our relation linking component on the basis of our entity linking (BERT) component outputs. From the results, we observe that our independent relation linking component outperforms the state-of-the-art results by Falcon. Despite the error propagation caused by entity linking, the results of our relation linking component are still promising. The experimental results show that it is difficult to obtain better results by only relying on semantic similarity or graph algorithm. Although Falcon leverages external resources to obtain a better recall value, its accuracy is limited by its linguistic methods. Our multi-attention RNN model can better deal

with the relation linking task.

Tab. 3 Macro precision, recall, and  $F_1$  values of relation linking approaches

Approach	Precision	Recall	$F_1$
SIBKB	0.13	0.15	0.14
ReMatch	0.15	0.17	0.16
EARL	0.17	0.21	0.18
Falcon	0.42	0.44	0.43
Our RL (golden EL)	<b>0.47</b>	<b>0.47</b>	<b>0.47</b>
Our RL (our EL)	0.45	0.45	0.45

3.2 Error analysis

For error analysis, we list some question examples in Tab. 4 (the named entity mentions are underlined), for which our approach does not output correct results.

Tab. 4 Failure analysis results

Reason	Sample question
Entity linking	Q1: What is the serving railway line of <u>Warwick railway station</u> , <u>Perth</u> ?
	Q2: List the <u>Sci-fi</u> TV shows with theme music given by <u>Ron Grainer</u> ?
	Q3: Who started at the pole position in both <u>1997 canadian grand prix</u> and the <u>94 spanish one</u> ?
	Q4: Which Texas based company was founded by <u>Jim Harris</u> ?
S/C classification	Q5: What is the baseball team whose manager is <u>Chip Hale</u> ?
Relation linking	Q6: Give me the home town of all musical artists who uses <u>Guitar</u> as instrument?
	Q7: What magazine companies are of form <u>Limited liability company</u> ?
	Q8: Which universities are also known as the <u>Tulane Green wave</u> ?
	Q9: What was the language of the single which came before <u>To Know Him Is to Love Him</u> ?
	Q10: Who was the voice actor of <u>allen walker</u> also gave voice to <u>kimihiro watanuki</u> ?

The first part of the samples is the failure of the entity linking component, which means that we fail to recognize the correct entity mentioned in the question. Such occurrence is common when the mention is too long or too sketchy, such as the long mention “Warwick railway station, Perth” in Q1, the acronym mention “Sci-fi” in Q2, and the long mention “1997 canadian grand prix” and the omitted mention “94 spanish one” in Q3. There are also failures which links the mention to the entity with the same name but the wrong type. For example, the mention “Jim Harris” is linked to the wrong entity “Jim\_Harris\_(politician)” instead of “Jim\_Harris\_(entrepreneur)” in Q4. The second part of the samples is the failure of the simple or complex classification for the question. For exam-

ple, Q5 is a simple question with a type constraint. However, it is incorrectly recognized as a complex question. The incorrect classification leads to the incorrect results of the following components. The third part of the samples is the failure of the relation linking component, which has the largest proportion. One cause is the error propagation; for example, the wrong entity linking results for Q6 and Q7 lead to the failure of relation linking. Another cause is that implicit relation is difficult to detect. For the simple question Q8, the surface form “are known as” is not identified as relation “nickname” correctly. The surface form “before” in Q9 is not identified as relation “previousWork” correctly. The causes of wrong relation linking for complex ques-

tions are varied because large numbers of two-hop relation candidates compound the difficulties of relation linking. For the complex question Q10, we incorrectly detect the relation “creator · voice” instead of “voice · voice”.

## 4 Conclusions

1) For entity linking, which is an essential component for relation linking, we train a Bi-LSTM-CRF model to recognize entities mentioned in a question and link them to the entities in the knowledge base based on a dictionary.

2) We propose a relation linking approach based on the multi-attention RNN, which can deal with the relation linking problem for both simple and complex questions. We first classify questions as simple or complex questions. On the basis of the classification, we apply different attention-based Bi-LSTM models for relation linking.

3) The experimental results show that our entity and relation linking components achieve the best reported macro precision on the LC-QuAD dataset.

4) In future, we plan to improve the relation linking results for complex questions. We also intend to consider aggregation constraints to generate final structured queries for the questions.

## References

- [1] Singh K, Radhakrishna A S, Both A, et al. Why reinvent the wheel: Let’s build question answering systems together [C]//*Proceedings of the 2018 World Wide Web Conference*. Lyon, France, 2018: 1247 – 1256. DOI: 10.1145/3178876.3186023.
- [2] Mendes P N, Jakob M, García-Silva A, et al. DBpedia spotlight: Shedding light on the web of documents[C]//*Proceedings of the 7th International Conference on Semantic Systems*. Graz, Austria, 2011: 1 – 8. DOI: 10.1145/2063518.2063519.
- [3] Hoffart J, Yosef M A, Bordino I, et al. Robust disambiguation of named entities in text[C]//*Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, UK, 2011: 782 – 792.
- [4] Moro A, Raganato A, Navigli R. Entity linking meets word sense disambiguation: A unified approach [J]. *Transactions of the Association for Computational Linguistics*, 2014, 2: 231 – 244. DOI: 10.1162/tacl\_a\_00179.
- [5] Speck R, Ngonga Ngomo A C. Ensemble learning for named entity recognition[C]//*Proceedings of the 13th International Semantic Web Conference*. Trentino, Italy, 2014: 519 – 534. DOI: 10.1007/978-3-319-11964-9.
- [6] Ferragina P, Scaiella U. TAGME: On-the-fly annotation of short text fragments (by wikipedia entities)[C]//*Proceedings of the 19th ACM Conference on Information and Knowledge Management*. Toronto, Canada, 2010: 1625 – 1628. DOI: 10.1145/1871437.1871689.
- [7] Waitelonis J, Sack H. Named entity linking in #Tweets with KEA [C]//*Proceedings of the 6th Workshop on ‘Making Sense of Microposts’*. Montreal, Canada, 2016: 61 – 63.
- [8] Singh K, Mulang I O, Lytra I, et al. Capturing knowledge in semantically-typed relational patterns to enhance relation linking[C]//*Proceedings of the Knowledge Capture Conference*. Austin, TX, USA, 2017: 1 – 8. DOI: 10.1145/3148011.3148031.
- [9] Mulang I O, Singh K, Orlandi F. Matching natural language relations to knowledge graph properties for question answering [C]//*Proceedings of the 13th International Conference on Semantic Systems*. Amsterdam, the Netherlands, 2017: 89 – 96. DOI: 10.1145/3132218.3132229.
- [10] Yu M, Yin W, Hasan K S, et al. Improved neural relation detection for knowledge base question answering [C]//*Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Vancouver, Canada, 2017: 571 – 581. DOI: 10.18653/v1/p17-1053.
- [11] Dubey M, Banerjee D, Chaudhuri D, et al. EARL: Joint entity and relation linking for question answering over knowledge graphs[C]//*Proceedings of the 17th International Semantic Web Conference*. Monterey, CA, USA, 2018: 108 – 126. DOI: 10.1007/978-3-030-00671-6\_7.
- [12] Sakor A, Mulang I O, Singh K, et al. Old is gold: linguistic driven approach for entity and relation linking of short text[C]//*Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN, USA, 2019: 2336 – 2346. DOI: 10.18653/v1/n19-1243.
- [13] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [C]//*Proceedings of the International Conference on Learning Representations*. San Diego, USA, 2015.
- [14] Trivedi P, Maheshwari G, Dubey M, et al. LC-QuAD: A corpus for complex question answering over knowledge graphs[C]//*Proceedings of the 16th International Semantic Web Conference*. Vienna, Austria, 2017: 210 – 218. DOI: 10.1007/978-3-319-68204-4\_22.
- [15] Usbeck R, Röder M, Ngomo A C N, et al. GERBIL: General entity annotator benchmarking framework [C]//*Proceedings of the 24th International Conference on World Wide Web*. Florence, Italy, 2015: 1133 – 1143. DOI: 10.1145/2736277.2741626.
- [16] Pennington J, Socher R, Manning C D. GloVe: Global vectors for word representation [C]//*Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Doha, Qatar, 2014: 1532 – 1543. DOI: 10.3115/v1/d14-1162.
- [17] Devlin J, Chang M, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [C]//*Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Minneapolis, MN, USA, 2019: 4171 – 4186.

# 面向知识库问答的多注意力 RNN 关系链接方法

李慧颖      赵 满      余文麒

(东南大学计算机科学与工程学院, 南京 211189)

**摘要:**针对知识库问答场景中的关系链接任务,尤其是面向复杂问题的多关系链接任务,提出一种基于多注意力 RNN 模型的关系链接方法,该方法既适用于简单问题也适用于复杂问题. 首先,在字符及词级别基础上通过 Bi-LSTM 模型学习问句的向量表示,利用条件随机场模型标注问句中的命名实体. 基于词典生成候选实体,基于预定义规则实现候选实体消歧,将问句中命名实体指称链接到知识库中实体. 然后,采用机器学习方法将问句分类为简单问题及复杂问题. 从已识别实体出发,对于简单问题,在知识库中收集一跳关系作为候选关系;对于复杂问题,收集二跳关系作为候选. 最后,采用多注意力 Bi-LSTM 模型对问句及候选关系进行编码,比较相似度,返回相似度最高的候选关系作为关系链接的结果. 值得注意的是,对于简单问题采用带有一个注意力的 Bi-LSTM 模型,对于复杂问题则采用带有 2 个注意力的 Bi-LSTM 模型. 实验结果表明:在有效的实体链接方法基础上,引入注意力机制的 Bi-LSTM 关系链接方法对于无论简单问题还是复杂问题效果都有所提升,且优于现有基于图算法或基于语言学的关系链接方法.

**关键词:**知识库问答;实体链接;关系链接;多注意力双向长短时记忆网络;大规模复杂问答数据集

**中图分类号:**TP311