A General Framework for Chinese Domain Knowledge Graph Question Answering Based on TransE

Zhongmin Pei¹, Jie Zhang¹, a, Wei Xiong¹, Li Li², Zhangkai Luo¹

¹Science and Technology on Complex Electronic System Simulation Laboratory, Space Engineering University, Beijing, China
²Gaojing Network Management and Security Center, Beijing, China

Email: misite_J@outlook.com, azhangjie1214@mails.jlu.edu.cn

Abstract: This paper introduces a general question answering (QA) framework for Chinese domain knowledge graphs to serve various industries. The question is analyzed by shielding invalid characters in this framework, and the TransE calculation of entities and relations is used to search answer, rather than traditional logical queries that rely heavily on predefined rules. In this way, the QA framework can be easily applied to different fields with little manual participation, and reduce the migration cost caused by the lack of scalability of the existing QA systems, which are designed only for specific domains. Then, the proposed framework is operated on the space launch knowledge graph and has achieved HITS@1, HITS@3, HITS@5 scores of 60.60%, 80.87%, 85.61% separately, which is competitive with other restricted domain knowledge graphs. In addition, numerical results on the knowledge graph of the Three Kingdoms and the COVID-19-Character verified the scalability of the QA framework.

1. Introduction

Knowledge graph (KG) is a kind of semantic network that reveals the relation between entities, which is able to describe things in the real world and their relations. According to the coverage, KG can be divided into general knowledge graph and domain knowledge graph [1]. General knowledge graphs focus on breadth and the integration of more entities, such as Google Knowledge Graph, FreeBase, and WikiData, which are mainly used for Internet-oriented search, recommendation, question answering (QA) and other business scenarios; in addition, domain knowledge graphs are oriented to specific industries and rely on data in domains, usually have certain depth and completeness, which make sophisticated application analysis as well as decision support possible in kinds of certain domains, such as medical, finance and tourism. Compared with the general knowledge graph, the domain knowledge graph has the advantages of clear subjects, rich entity attributes and data formats, high accuracy, which are suitable to build a QA system for different business scenarios and users.

In recent years, the Chinese domain knowledge graph question answering (CdKGQA) has made great progress, however most QA systems are limited at specific industries and are difficult to expand. Therefore, the design of a general QA framework is imperative. Reference [2] designs an ontology-based aviation QA system, which analyses question through morphological analysis and ontology-based classification, then the question is converted into SPARQL query based on the template for answer extraction; Reference [3] constructs a QA system based on military knowledge graph, which combines shallow syntax analysis and question templates to identify entities and users’ intentions, then query answer on the graph database; Reference [4] introduces a pipeline QA system to knowledge graph of...
primary liver cancer, where BiLSTM-CRF is used for medical entity recognition, TF-IDF and word embedding are used for sentence embedding to select the corresponding question template; Reference [5] combines CRF, template matching and SQL query to design an intelligent QA system based on knowledge graph of Chinese medicine; Reference [6] implements QA system on knowledge graph for Princes Jingjiang Residence, the word substitution and semantic similarity between question and template are used for question parsing, then the Cypher query and Levenshtein distance are used for answer generation.

However, a major problem with the traditional QA system is its weak scalability, which means too much manual participation for different types of questions and logical query templates. Besides, few researches consider the particularity of questions in Chinese scenarios: (1) Unlike English sentences, Chinese sentences are composed of a series of Chinese characters, rather than separated by spaces; (2) Chinese expressions are more diverse, such as terminology, aliases, and abbreviations exist everywhere especially in specific fields; (3) Chinese questions generally conform to the sequential logical syntax of "subject-predicate-object".

Based on the work of previous scholars, this paper designs a general QA framework for Chinese domain knowledge graph, which recognizes entities and relations by shielding invalid characters, then introduces the TransE into the answer extraction module. Experiments and analysis of the QA framework on the knowledge graph of the space launch show the framework is feasible, then the implementations on the knowledge graph of the Three Kingdoms and the COVID-19-Character verify the scalability of the QA framework.

2. Related work
To design the general CdKGQA framework, this paper uses stop words and word embedding to parse question, and uses TransE to reason and search answer. In this section, stop words, word embedding, and TransE embedding are introduced respectively.

For natural language processing tasks, filtering stop words can improve the efficiency and accuracy of text processing, that is, the system could automatically remove certain words. However, no one stop words list is suitable for all tasks. In search systems, stop words are words that appear frequently but are not helpful for the result [7]; in classification systems based on support vector machine, stop words refer to function words without meaning and neutral words with weak category features [8]; in QA systems, stop words would change dynamically considering different questions [9]. As for the QA system in a specific field, the domain stop words list is extracted through probability and content analysis after checking and analysing a large amount of text in this field, include text frequency, word frequency statistics, entropy calculation, CHI Statistics, etc.[10] Aiming at the problem that characters are the basic unit of Chinese sentences, a method for constructing domain stop characters is proposed - called invalid characters in this paper, which is used to segment the question.

Word Embedding transforms the word into a distributed representation of a fixed-length continuous dense vector that contains tremendous information, and the value of each dimension represents a feature with a certain semantic and grammatical interpretation [11]. Therefore, the similarity of text can be expressed by calculating the cosine similarity between word vectors. In [12], the one-hot representation of a word is converted into a low-dimensional vector by CBOW or Skip-Gram, which is used in almost all the natural language processing tasks. However, both character (single Chinese character) and word (word or phrase) is able to use as the language unit for distributed representation in Chinese information processing. Reference [13] prove that the sparse distribution of words leads to problems such as OOV and overfitting, which can be avoid by character vectors, it is also found that the performances of using character vectors always achieve better results in most Chinese tasks.

Inspired by the translation invariance of the word embedding, TransE was proposed in [14], which is a multi-relation-based translation model. Relations in the knowledge graph are regarded as some kind of translation vectors between entities. During the training process, \( v_{head} + v_{relation} = v_{tail} \) is used as the constraint condition, vectors of head entity, relation, and tail entity are constantly adjusted to achieve the embedding of all KG triples. Due to fewer parameters, the computational complexity with TransE is
greatly reduced, and it also has good performance and scalability on large-scale sparse knowledge graphs.

3. QA framework for Chinese domain knowledge graph

3.1. Framework

![Figure 1. Chinese domain knowledge graph QA framework.](image)

In this paper, we propose a general QA framework (Figure 1) for Chinese domain knowledge graph, which consists of three parts: question analysis, answer reasoning, and answer generation. First, the question analysis module obtains question entities and relations by shielding invalid characters. Then, the answer reasoning module combines sequential logic and permutation and combination strategies to calculate the TransE embedding of possible entities. Finally, the answer generation module calculates the cosine similarity of candidate answers and entities in knowledge graph, and returns reasoning paths and answer entities to the user.

3.2. Question Analysis Based on Invalid characters

The question analysis module identifies the entities and relations in natural language questions to better search and generate answers based on the knowledge graph. Aiming at the problem of less domain data, more proper nouns, fixed question types for domain knowledge graph, this module implements word segmentation by shielding invalid characters, then compares the word embedding with the entity and relation vectors in the knowledge graph, so as to identify entities and relations in the question.

The construction of the domain invalid character dictionary is composed of two parts. First, count the character frequency of a large set of Chinese questions, and select the first 100 high-frequency characters as the general question invalid character dictionary. Second, calculate the entropy value of each character for specific domain knowledge graph:

$$H(w) = 1 + \frac{1}{\ln(m)} \sum_{i=1}^{m} P_i(w) \ln [P_i(w)]$$

where $m$ is the number of triples, $P_i(w)$ is the probability that character $w$ appears in the $i$-th triple. The domain invalid character dictionary is obtained by arranging all characters in ascending order of the entropy value and removing the first 1/5 characters which are related to the domain question.

For sequential logical questions, the positions of subject, predicate, and object appear in sequence, and some invalid characters connect them into a complete question. For example, the triple "<mo zi hao wei xing yi hao, shi jian fa she (that is “fa she shi jian”), ?> can be got by analysing the question “mo zi hao wei xing yi hao shi shen me shi jian fa she de?”, where the entities and relations are connected by invalid characters (such as “shi”, “shen”, “me”, “de”, etc.) to form a complete question. Therefore, candidate entities and relations can be obtained by segmenting the question with invalid characters. However, these entities and relations may not be aligned with the entities or relations in the domain knowledge graph. Table 1 shows the problems of head-tail off, mid-segment split, and synonymous expression.
To solve those problems, Word2Vec is introduced to calculate the similarity between segmentation results and entities or relations in the domain knowledge graph to achieve matching. First, calculate the average of character vectors in the word according to $(v_{c1} + v_{c2} + \ldots + v_{cn})/n$, we can get vector representations of segmentation result and entity or relation in the domain knowledge graph separately. Then, the cosine similarity is calculated for the vector of each word segmentation and the vector set $M^r_{dc}$ of all elements in the domain knowledge graph, where $m$ is the total number of entities and relations, $d_c$ is the dimension of character vector. Finally, choose the top 3 most similar entities or relations, and assign probability values $\{1.0, 0.9, 0.8\}$ to the candidate entities and relations in turn.

Table 1. Question segmentation error type.

| Head-Tail off | Mid-segment split | Synonymous expression |
|---------------|-------------------|-----------------------|
| Question      | Zhong xing 6B tong xin wei xing wei yu na ge gui dao? | Qing hua na xing yi hao ke xue shi yan wei xing de fa she di dian shi na li? |
| Ideal entities/relations | zhong xing 6B tong xin wei xing, yun xing gui dao | qing hua na xing yi ke xue shi yan wei xing, fa she di dian |
| Triple        | < zhong xing 6B tong xin wei xing, yun xing gui dao, ? > | < qing hua na xing yi hao ke xue shi yan wei xing, fa she di dian, ? > |

* Italics indicate Chinese Pinyin.

### 3.3. Answer reasoning based TransE

In the reasoning phase of answer, this paper introduces TransE to transform the query of answer into a vector calculation process, which is completely different from the previous generation of SPARQL, Cypher and other logical query statements that rely heavily on manual definitions.

After traversing the question entities and relations in all possible combinations, we can hit on a unary relational form of $\{e_h, r\}$ or multi-relational form of $\{e_h, r_1, r_2, \ldots\}$, which is the answer reasoning path. Since $v_{e_h} + v_r = v_{e_t}$, the TransE vectors corresponding to the entities and relations on the answer reasoning path are added or subtracted to get the TransE of tail entity. Then, calculate the cosine similarities of the vector and TransE vectors of all entities in the domain knowledge graph, and select three entities with the highest similarity as the candidate answers in this path.

### 3.4. Answer generation

Through aforementioned steps, each question gets $3M$ candidate answers, where $M$ is the number of answer reasoning path, and the final answer is determined jointly by probabilities of candidate entities and relations obtained in question analysis module and the generational path of the answer reasoning module. The confidence of each candidate answer is calculated as follows:

$$C(a_j) = \cos_{a_j} \prod_{e \in A_j} \text{prob}_{e,r}$$  \hspace{1cm} (2)

where $A_j$ is the reasoning path corresponding to answer $a_j$, $\text{prob}_{e,r}$ is the probability corresponding to the entity or relation in the reasoning path $A_j$, $\cos_{a_j}$ is the cosine similarity between the TransE vector calculated from the answer reasoning path and the corresponding entity in the knowledge graph. The confidences of all answers are sorted in descending order, and the top $N$ results are returned to the user along with their answer reasoning path.

### 4. Experiment

This chapter implements a QA system on the knowledge graph of space launch based on the above framework, and perform error analysis with the experimental results. Furthermore, QA systems for the knowledge graph of the Three Kingdoms and the COVID-19-Character verify the scalability of the QA framework.
4.1. Dataset

Knowledge Graph  This paper collects the space launch records of China over the years and all space launch records at home and abroad over the past three years from http://www.spaceflightfans.cn/. Then, these raw data are cleaned and structured, manually checked to ensure the accuracy of the acquired knowledge. Finally, the knowledge graph of space launch was constructed, including 4127 triples, 1611 entities, and 16 types of relations.

The knowledge graph of the Three Kingdoms and the COVID-19-Character provided by OpenKG are also used to verify the scalability of the QA framework. Among them, the knowledge graph of the Three Kingdoms contains 10459 triples, 5084 entities, and 33 relations; the knowledge graph of the COVID-19-Character contains 991 triples, 759 entities, and 41 relations.

Invalid Character  The statistics of 5.8 million questions from Baidu Zhidao yielded 12,874 Chinese characters that fit the long-tail distribution, and selected 100 Chinese characters with the highest frequency as the general invalid character dictionary; for domain question, Equation (1) is used to filter the invalid characters. In this experiment, an 80-character domain invalid dictionary is obtained.

Chinese Character Vector  This framework uses an open-source Chinese character embedding project, which uses Skip-Gram to train Chinese Wikipedia and obtains 20,029 300-dimensional character vectors.

TransE  Embedding of the domain knowledge graph is based on RotaeE [15], the relevant parameters are set to Epoch = 40,000, Dim = 1000, Batch = 512.

QA Pair  Taking the space launch knowledge graph as an example, several question templates are generated based on 16 relations. For example, questions “eh de fa she shi jian shi?”, “ni zhi dao eh shi na tian fa she de?” can be generated according to the relation “fa she shi jian”, where eh is the head entity. In this way, a total of 7736 QA pairs are generated. Inspired by EDA (Easy Data Augment) [16], 5650 questions with one answer are expanded through random deletion and synonymous replacement, resulting in 16,950 questions, and 1/3 of them are randomly selected to evaluate the QA system.

4.2. Results of space launch knowledge graph

For each question, the QA system would return five answers with the most confidence and their corresponding answer reasoning paths. HITS@1, HITS@3, HITS@5 (HITS@N means the percentage of top N hits the correct answer) are used to quantify the accuracy of the QA system, with results show in Table 2.

| Table 2. Experimental results of the QA system for space launch knowledge graph. |
|---------------------------------|-----------------|-----------------|
| HITS@1                         | HITS@3          | HITS@5          |
| 60.60                          | 80.87           | 85.61           |

As we can see, although the system achieves HITS@1 of 60.60%, HITS@3 has increased by 20.27%, and HITS@5 has further reached 85.61%. At the same time, the results show that our QA system is competitive with the KGQA system in restricted domains [2, 4], which is able to meet the needs of different industries. In addition, the reasoning paths of the missed answers in top N are highly correlated with the user question, which plays a relevant recommendation role for users of the domain knowledge graph.

Furthermore, this paper analyses the 813 questions which miss answer in top 5, Table 3 reports the types of errors in each part of question analysis and answer reasoning.

| Table 3. Error analysis of each module of the QA system. |
|--------------------------------------------------------|
| Question Analysis                                    | Counts | Percentage |
| Shield invalid characters                             | 254    | 31.24      |
| Calculate cosine similarity of character vector       | 176    | 21.65      |
| Answer reasoning                                      | 240    | 29.52      |
In the question analysis module, there are 254 errors (31.24%) due to the wrong question segmentation caused by invalid characters, and 176 errors (21.65%) due to the mismatch of cosine similarity caused by the proximity of character vectors. As mentioned in section 3.2, there are problems of head-tail off, mid-segment split and synonymous expression, although character vectors can partially solve these problems, some wrong segmentations that are too ridiculous to be correctly matched based on the character vectors; besides, vectors of similar characters indicate the phenomenon of aggregation, such as “shijian shiyi hao 01 xing”, “shijian shiyi hao 04 xing” and “shijian shiyi hao 06 xing”, the correct entity cannot be got by the cosine similarity of character vectors, which would affect subsequent steps. In the answer reasoning module, there are 240 errors (29.52%) due to the wrong reasoning path caused by permutation and combination, and 143 errors (17.59%) due to the mismatch of cosine similarity caused by the proximity of TransE embeddings. Although two kinds of permutation and combination strategies of sequential logic and distinguishing between entity and relation are proposed in this paper, it is not enough for more complex questions or ambiguous questions, which require stronger semantic understanding or reasoning ability to learn the reasoning path; in addition, similar to character vectors, aggregated TransE embeddings cannot rely solely on cosine similarity to accurately match the answer.

### 4.3. Scalability analysis

To verify the scalability of the QA framework in this paper, we use it to implement QA systems for the knowledge graph of the Three Kingdoms and the COVID-19-Character separately. Table 4 reports that the QA framework has good scalability and can meet the needs of QA system for different domain knowledge graphs. The QA system performs well in small and medium knowledge graphs, but the accuracy decreases slightly with the scale of knowledge graph increases. The possible reason is that relations in larger knowledge graphs are more complicated, and there are high-level relation modes such as symmetry or antisymmetry, inversion, and composition, which put higher requirements on the intelligence of the QA system.

| KG of the COVID-19-Character | KG of the space launch | KG of the Three Kingdoms |
|-------------------------------|------------------------|--------------------------|
| KG scale                      | HITS@1                 | HITS@3                   | HITS@5                   |
| Smaller                       | 72.93                  | 86.93                    | 89.61                    |
| Medium                        | 60.60                  | 80.87                    | 85.61                    |
| Larger                        | 65.11                  | 76.84                    | 81.20                    |

### 5. Conclusions

This paper proposes a general QA framework for the Chinese domain knowledge graph, which parses the question with domain invalid characters and queries the answer with TransE embedding. The experiments on different Chinese domain knowledge graphs verify the feasibility and scalability of the framework. In the future, we plan to combine invalid characters and named entity recognition based on pre-trained language models, and apply improved models such as TransH or TransD to solve 1-N and N-N problem in QA pairs.

### Acknowledgements

This work was supported by the Fundamental Research Program of Science and Technology on Complex Electronic System Simulation Laboratory.

### References

[1] Xu Z, Sheng Y, He L, et al. (2016) Review on Knowledge Graph Techniques. Journal of University
of Electronic Science and Technology of China, 45: 589–606.

[2] Zhang K, Li W, Wang H. (2015) Ontology-based Question Answering System for Aviation Domain. Journal of Chinese Information Processing, 29: 192–198.

[3] Dou X, Liu T, Zhang Z. (2018) Question Answering System Based on Military Knowledge Graph. In: The 6th China Conference on Command and Control. Beijing. pp. 537-541.

[4] Cao M, Li Q, Yang Z, et al. (2019) A Question Answering System for Primary Liver Cancer Based on Knowledge Graph. Journal of Chinese Information Processing, 33: 88–93.

[5] Chen C, Zhai J, Qin J, et al. (2018) Design and Development of Intelligent Question Answering System Based on Knowledge Map of Chinese Medicine. China New Telecommunications, 20: 204–207.

[6] Shi Y, Gu T, Bin C, et al. (2018) Question and answer system of tourist attractions based on knowledge graph. Journal of Guilin University of Electronic Technology, 38: 296–302.

[7] Luhn H P. (1957) A Statistical Approach to Mechanized Encoding and Searching of Literary Information. IBM Journal of Research and Development, 1: 309–317.

[8] Zhou Q, Sun B, Wang Y. (2004) Study on New Pretreatment Method for Chinese Text Classification System. Application Research of Computers, No. 2: 89–90.

[9] Xiong W, Song R. (2007) Removal of Stop Word in Users’ Request for Information Retrieval. Computer Engineering, 33: 201–203.

[10] Chen X, Zhang J, Li X, et al. (2011) A Text Classification Method for Chinese Pornographic Web Recognition. Measurement and Control Technology, No. 5: 41-45+54.

[11] Turian J P, Ratinov L A, Bengio Y. (2010) Word Representations: A Simple and General Method for Semi-Supervised Learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala. pp. 384–394.

[12] Mikolov T, Chen K, Corrado G, et al. (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[13] Li X, Meng Y, Sun X, et al. (2019) Is Word Segmentation Necessary for Deep Learning of Chinese Representations? In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence. pp. 3242–3252.

[14] Bordes A, Usunier N, Garcia-Duran A, et al. (2013) Translating embeddings for modeling multi-relational data. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. Nevada. pp. 2787–2795.

[15] Sun Z, Deng Z-H, Nie J-Y, et al. (2019) Rotate: Knowledge graph embedding by relational rotation in complex space. arXiv preprint arXiv:1902.10197.

[16] Wei J W, Zou K. (2019) Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In: International joint conference on natural language processing. Hong Kong. pp. 6381-6387.