Calculating Question Similarity is Enough: A New Method for KBQA Tasks

Hanyu Zhao
Beijing Academy of Artificial Intelligence
Beijing, China
hyzhao@baai.ac.cn

Sha Yuan
Beijing Academy of Artificial Intelligence
Beijing, China
yuansha@baai.ac.cn

Jiahong Leng
Beijing Academy of Artificial Intelligence
Beijing, China
jhleng@baai.ac.cn

Xiang Pan
New York University
New York, America
xiangpan@nyu.edu

Guoqiang Wang
Beijing Academy of Artificial Intelligence
Beijing, China
gqwang@baai.ac.cn

ABSTRACT
Knowledge Base Question Answering (KBQA) aims to answer natural language questions with the help of an external knowledge base. The core idea is to find the link between the internal knowledge behind questions and known triples of the knowledge base. The KBQA task pipeline contains several steps, including entity recognition, relationship extraction, and entity linking. This kind of pipeline method means that errors in any procedure will inevitably propagate to the final prediction. In order to solve the above problem, this paper proposes a Corpus Generation - Retrieve Method (CGRM) with Pre-training Language Model (PLM) and Knowledge Graph (KG). Firstly, based on the mT5 model, we designed two new pre-training tasks: knowledge masked language modeling and question generation based on the paragraph to obtain the knowledge enhanced T5 (kT5) model. Secondly, after preprocessing triples of knowledge graph with a series of heuristic rules, the kT5 model generates natural language QA pairs based on processed triples. Finally, we directly solve the QA by retrieving the synthetic dataset. We test our method on NLPCC-ICCPOL 2016 KBQA dataset, and the results show that our framework improves the performance of KBQA and the straightforward method is competitive with the state-of-the-art.

1 INTRODUCTION
Question Answering System is a long-popular topic in the natural language processing area. Question Answering task requires the model has some knowledge. One direction is to use an existed knowledge base to enrich the model knowledge to get better model performance, which is called Question Answering (QA) over Knowledge Base (KB) or Knowledge Base Question Answering (KBQA). The mainstream of KBQA is to link the current question sentence’s entities into the KBQA entities and use the relation that appeared in the sentence to induct in the Knowledge Graph to get the final answer. At present, KBQA tasks are mainly solved by a set of pipeline processing step by step, as shown in Figure 1, including entity identification, relationship extraction, entity link, and knowledge base query.

Figure 1: Traditional Pipeline Structure

Due to the diversity of natural language expression and errors will appear in every link of such pipeline. In identifying the question’s subject, both the difference of subject and object and the model error will affect the model’s performance. For example, given a question ‘主持人李冀川是哪里人 (where dose host Li Jichuan come from)’, it can be answered by the triple [李冀川, 成都电视台《道听途说》节目主持人 (Li Jicuan, host of Chengdu TV program 'Daotingtushuo')], [籍贯 (native place); 成都市温江区 (Wenjiang District, Chengdu City)]. However, in the process of entity identification, it is possible that ‘李冀川 (Li Jichuan)’ cannot be accurately identified as the name of a person. It is also difficult to match ‘哪里人 (where come from)’ and ‘籍贯 (native place)’ to
each other. Besides, there can be several ‘李冀川 (Li Jichuan)’ in the KB. It means that entity extraction of example question becomes more complex, because the subject ‘李冀川, 成都电视台《道听途说》节目主持人 (Li Jicuan, host of Chengdu TV program ‘Daotingtushuo’)’ needs to be linked correctly at first based on question information. When applying the above pipeline methods, any errors occurred midway will affect all the subsequent pipeline links, thus degrade the whole KBQA system to a great extent.

In order to solve the above problem, we proposed a QA Corpus Generation-Retrieve Method (CGRM). Our procedures are shown in Figure 2. In our method, the Pre-trained model is used to query the processed knowledge triples to generate a huge QA corpus. The final answer of user’s question can be obtained by retrieving similar questions generated in the corpus. Specifically, based on the mT5 [36] model, we designed two new pre-training tasks: knowledge masked language modeling and question generation based on the paragraph to obtain the knowledge enhanced T5 (kT5) model. Compared with the mT5 model, kT5 is enhanced with word knowledge and few-shot learning. In addition, to enhance the language generation ability of the model, kT5 regards every single word as a unit, which can effectively alleviate the problem of exposure bias in generative tasks. Later, the [MASK] symbol is used to replace any entity in the triplet of the knowledge graph to make the entity the answer. Taking the triplet after masking as the input, seq2seq fine-tuning is carried out with the help of the kT5 model to generate the corresponding problem, to build a huge QA corpus. Subsequently, we use the fine-tuned model to generate the exhaustive question-answering pair corpus. Finally, we transform the KBQA task into a retrieval problem. The answer can be retrieved in the generated QA dataset for a user’s question by computing question similarity.

To sum up, our work proposes a new solution for KBQA tasks, eliminating the problem of label propagation error in traditional systems. When deploying our KBQA system, it only needs to retrieve generated dataset with the user’s query, which significantly simplifies the operation process and improves the efficiency of the deployment system. We summarize our main contributions as below:

- We propose the new method CGRM for the KBQA task, improving the pipeline accuracy and deployment efficiency. In addition, our method does not require domain knowledge and handcrafted feature or model structure design.
- Towards the KBQA task, we proposed the pre-trained knowledge-enhanced mT5 (kT5), which shows tremendous few-shot learning. Such few-show learning abilities benefit various domain KBQA deployment without heavy training.
- Based on our pipeline and methods, we propose a QA corpus, which can facilitate the QA research.

2 RELATED WORK

2.1 Knowledge Based Question Answering

KBQA is an essential direction of the research of question-answering tasks. It can be divided into three main directions: Semantic Parsing methods, Information Extraction (IE) methods, and Representation Learning methods.

Semantic Parsing can be divided into two steps: question semantic representation and semantic matching with a knowledge base. Early studies rely on predefined rules or templates to parse questions and obtain logical forms, which finish two tasks together [32, 40]. Methods like bottom-up semantic parser [2], question syntax dependency [35] and staged query graph [38], are applied for question semantic analysis. Later, semantic query graph (SQG) [1, 5, 6, 15, 16, 30] and Graph Neural Network [10, 26] are used to enhance the semantic parsing performance.

Information Extraction collect the entity in the question. By querying the entity in the knowledge base, we can get a subgraph centered on the entity node and treat all the nodes in the subgraphs as candidate answers. Related studies [37] analyzed the question syntax information and extracted features to get the representation.

Representation Learning is similar to information extraction methods. The representation is more learned from neural network rather than from hand crafted rules or dependency information. Knowledge embeddings [25], attention mechanism [11], and variational reasoning network [41] are used to get better representation.
2.2 Question Generation Base on Knowledge Graph

Question Generation (QG) concerns the task of generation from various sources [21]. For the knowledge Graph, we can treat the source as the entity or the triple. For the generating questions from vocabulary, some early studies use WordNet [20], and techniques of distributional similarity [3]. Recently, Reddy et.al. [24] used RNN to get triple-based generation. Recently, many studies investigate the method of generating questions by using graph transformer [4, 12, 13] or sequence labeling [9, 29].

2.3 Pre-trained Language Models

Pre-trained Language Models have achieved outstanding performance in various NLP tasks, including generation and semantic matching. Among these models, BERT [7] and Text-to-Text Transfer Transformer (T5) [23] are two representative ones. BERT is a general pre-trained language model for the various downstream tasks, including the sentence encoding and sentence pair. In this work, we use the BERT-based model for candidate ranking [19].

T5 applies standard encoder-decoder architecture and constructs unsupervised and supervised text generation as pre-training tasks. Model mT5 is the multilingual version of T5, which supports 101 languages. Besides, mT5 uses a linear gating unit with a GELU activation function. As for the embedding layer, the T5 model shares the same embedding matrix in the encoder, decoder, and the softmax layer of the decoder. In contrast, the mT5 model only shares the same parameters of embedding layer in encoder and decoder and uses an independent embedding matrix in the softmax layer of the decoder to predict final probability distribution. Moreover, the dropout layer of mT5 is removed in pre-training to further enhance the model’s performance.

3 METHOD

3.1 kT5

In our paper, we use mT5 as our pre-training model backbone. Moreover, to make the mT5 model generate natural language question-answer pairs from a knowledge graph better, we improve the mT5 model from two aspects of pre-training tasks and vocabulary configuration to obtain knowledge enhanced T5 model (kT5).

3.1.1 Pre-training tasks. According to the target task, we define two pre-training tasks that apply the basic structure and initial weight of the Google mT5, as shown in Figure 3.

- **Question Generation Based on Paragraph** We use the passage and answer pairs of WebQA and SogouQA to train the model to improve its ability of question generation.
- **Knowledge Masked Language Modeling** This task is similar to the unsupervised pre-training task designed for the T5 model, which is the prediction of masked entities in the input text. Here, we use the summary of encyclopedia entries in WuDaoCarpota [39] as the training data. Instead of randomly masking words in a sentence, we mask the linked entity in the summary of encyclopedia entries to enhance the word knowledge of the trained kT5 model.

3.1.2 Vocabulary Configuration. In mT5 model, the text is encoded as WordPiece [33] tokens using SensePiece. For all experiments, mT5 uses a vocabulary of 250 thousand WordPieces. However, the original algorithm of SensePiece is not effective in the Chinese environment because its performance on word segmentation is poor, and the algorithm will forcibly convert some full-angle symbols into half-angle symbols. Thus, we use pkuseg1 to encode text as WordPiece tokens instead. Besides, our kT5 model is designed mainly for Chinese tasks and supports English as an auxiliary language. Nevertheless, the mT5 model uses a huge amount of vocabulary in other languages, which causes the parameter over-occupied of the embedding layer. Therefore, we supplement Chinese vocabulary and delete those words that are not Chinese or English, and the total vocabulary is reduced to 50 thousand.

3.2 Question Generation

We use the knowledge base provided by NLPCC-ICCPOL 2016 KBQA task2. It is a collection of facts that are stored as triples in the form of (subject|||predicate|||object). In order to query more

---

1. [https://github.com/lancopku/pkuseg-python](https://github.com/lancopku/pkuseg-python)
2. [http://tcci.ccf.org.cn/conference/2016/dldoc/evagline2.pdf](http://tcci.ccf.org.cn/conference/2016/dldoc/evagline2.pdf)
efficiently, the knowledge graph is stored in the neo4j database. To finally generate a large QA corpus from the knowledge graph by using a pre-trained model, we follow several vital steps as shown in Figure 4.

First, for each node in the graph, as shown at the top of figure 4, the related knowledge triples are extracted according to their connected relations. Taking entity “徐峥(Xu Zheng)” as an example, we can extract the knowledge related to it, such as [徐峥(Xu Zheng); 国籍(nationality); 中国(China)] and [徐峥(Xu Zheng); 妻子(wife); 陶虹(Tao Hong)].

After knowledge extraction, we use the “?” as the special token to randomly mask one of the entities in knowledge extracted by the above rules. Subsequently, we manually generate questions for a small volume of the above-processed knowledge. Those generated questions and corresponding knowledge tuples together form a large raw QA corpus. Finally, we use heuristic rules to filter the raw QA dataset according to the quality of generated questions. It is worth mentioning that, to avoid the occasionally occurred repetition problem in text generation tasks, we segment every generated text and count the proportion of each word in it. If the proportion of a word in a question exceeds 70%, the corresponding QA pair will be deleted from the dataset. The bottom of Figure 4 gives some examples of generated QA pairs.

3.3 Retrieval

Information Retrieval based methods usually use the questions raised by the user and the questions in the corpus to calculate the similarity and return the answers corresponding to the questions with the highest match scores. In order to prove the effectiveness of the CGRM scheme, in terms of problem similarity calculation, this paper uses the commonly used pre-training model BERT and compares it with other KBQA methods using BERT to prove the effectiveness of the scheme. At the same time, because the number of corpora generated is too large (4300w +), to comprehensively consider the efficiency, this paper uses a two-phase scheme containing two subtasks: Retrieve (Rough-Rank) and Rank (Precise-Rank). The Retrieve part usually uses efficient models to recall Top K candidate data from the corpus. In contrast, the rank part uses a model with semantic representation to reorder the recalled Top K data and select the calculated best match sample as the final result.

3.3.1 Recall Algorithm. Best Match 25 (BM25) [31] is an algorithm to score documents in the information retrieval system according to the proposed query. The algorithm first parses the morpheme of a query to generate the morpheme $q_i$; then, for each search result $D$, calculates the correlation score of each morpheme $q_i$ and $D$; finally, weighted sum the correlation score of $q_i$ relative to $D$ to get the correlation score of query and $D$.

$$
score(D, Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})}
$$

where $Q$ is a collection of documents, and $D$ is a query statement. $IDF(q_i)$ means the IDF value of the word $q_i$ in the document collection $Q$, $f(q_i, D)$ means the TF value of the word $q_i$ in the document $D$, and $k_1$ is the term frequency saturation, which is used to adjust the rate of saturation change. $b$ is the field length reduction, which is used to reduce the length of the document to the average length of all documents. Its value is between 0 and 1, 1 means all reduction, and 0 does not reduce. $|D|$ represents the length of the query statement, and $avgdl$ represents the average length of all text.

3.3.2 Rank Algorithm. In a dialogue system, BERT can be used to select and sort candidate answers (see Figure 5). This paper will query and candidate question are concatenated into a sequence, with [CLS] as the start of the sequence, [SEP] is inserted between query and candidate question as a separator, and the position at
[CLS] is calculated softmax judges whether the answer is reasonable, and sorts according to the probability distribution of the positive example to get the final result.

Figure 5: Rank model based on BERT

4 EXPERIMENT

Table 1: Overall information of NLPCC-ICCPOL 2016 KBQA dataset

| Data             | Train | Test | Entities | Attributes | Triples |
|------------------|-------|------|----------|------------|---------|
| NLPCC-ICCPOL 2016 KBQA | 14609 | 9870 | 6,502,738 | 587,875    | 43063795 |

4.1 Pre-tranining Model Details

The maximum length of kT5 training is set to 512, the batch_size and learning rate are 128 and 10⁻⁴ respectively. In addition, 8 NVIDIA Tesla V100 with 32GB storage are used to train the model for 500 thousand steps.

4.2 Evaluation Metric

4.2.1 QA Evaluation Metric. For QA tasks, if the starting or ending position is close but not the same, the answer is still valuable sometimes. Therefore, we use the F1-score, which can measure the characters’ overlap between the prediction and ground truth. In this paper, if there are successive non-Chinese tokens, they are counted as one unit length rather than segmented into characters.

4.2.2 Question Generation Evaluation Metric. For question generation task, four measures are used for evaluating the quality of output questions. They are Rouge_1, Rouge_2, Rouge_1 and BLEU.

Rouge_n (Recall oriented under study for gisting evaluation) [18] is a set of indicators for evaluating generation tasks such as automatic summarization and machine translation. It measures the recall rate between generated outputs and reference answer. Here, n is the gram-number, which can be regarded as the length of slide window. In majority of evaluation work, n=1 and n=2 are the most frequently used cases. Similarly, we use Rouge_1 and Rouge_2 as two measures in this paper.

Rouge_l is a Rouge-based measure that considers longest common subsequence (LCS) in precision and recall rate calculation between generated answer and reference answer. Generally, the Rouge_l is set to focus more on the recall part.

BLEU (Bilingual Evaluation Understudy) [22] is another measure that focuses more on the precision rate. It is a semantic measure that is first used in translation tasks. BLEU computes the gram-n overlap between generated text and reference text, and adds a penalty factor for short generation.

Human evaluation is also used in this paper because above measures only calculate the sentence similarity, and can not correctly evaluate those outputs that are semantically similar but differently expressed to ground truth. In addition, human evaluation is necessary in the evaluation of fluency and relevance of a given sentence.

4.3 NLPCC 2016

The dataset used in this paper is a Chinese knowledge base (KB), which was presented by a KBQA evaluation task of NLPCC-ICCPOL 2016. It provides a training set with 14,609 question answer (QA) pairs and a test set with 9870 QA pairs. The whole Chinese KB consists of 6,502,738 entities, 587,875 attributes and more than 43M triples in the form of (subject, predicate, object). In addition, the dataset also offers a mention2id library, which can map entities in those questions to their names in the KB. In following experiments, 20% data of training set are randomly selected to form the development set. The overall data information is shown in Table 1.

4.4 KBQA

We compare our system with the baseline system (CDSSM) released in NLPCC-ICCPOL 2016 KBQA task, the state-of-the-art system [19] and some other baseline systems [17, 27, 42, 43]. Table 2 demonstrates the experimental results on NLPCC-ICCPOL 2016 KBQA task. It can be found that our system outperforms all other methods. Compared with systems using pipeline model method and other sophisticated features and hand-craft rules (such as the use of part-of-speech features in the mention detection stage) [14], our framework can still improve the performance on KBQA task, and the straight-forward method is competitive with the state-of-the-arts one.

Table 2: NLPCC-ICCPOL 2016 KBQA task

| Model             | F1   |
|-------------------|------|
| CDSSM(baseline system) | 52.47 |
| InsunKBQA         | 80.97 |
| B. Zhou et al [43]| 81.06 |
| Kai Lei et al    | 80.97 |
| Lei Su et al [27] | 79.41 |
| Lai et al [14]   | 82.47 |
| CTEEM+Combined-DSSM+LMS [34] | 82.43 |
| BB-KBQA [19]     | 84.12 |
| MFSMM [42]       | 80.35 |
| SiameseATT [28]  | 81.81 |
| CGBM              | 85.04 |

Because traditional KBQA method needs to go through several steps such as entity identification, relationship extraction, entity linking and knowledge base query. Some label propagation errors is likely to occur during these steps. In particular, the entity identification and linking mistakes occur with a relatively high probability, which can significantly affect the precision of final answer. For example, in question '你知道知母属于什么纲的吗 (Do you know

\[\text{http://tcci.ccf.org.cn/conference/2016/dldoc/evagline2.pdf}\]
which class does rhizoma anemarrhena belong to)?", the entity '知母 (rhizoma anemarrhena) may be wrongly identified as '知母属 (anemarrhena bungee)'. The former plant belongs to the class of monocotyledonous while the latter one belongs to liliaceae. Once the entity cannot be identified correctly, the output will inevitably give a wrong answer to user’s question. In our KBQA approach, we use kT5 model to generate questions for both rhizoma anemarrhena’s and anemarrhena bungee’s class, obtaining questions '知母属属于什么纲 (Which class does rhizoma anemarrhena belong to)?' and '知母属属于什么纲 (Which class does anemarrhena bunge belong to)?'. By computing the similarity to user’s question, the correct answer is much possibly to be returned.

### 4.5 Question Generation

Table 3 shows the experimental results of four models' question generation measures. Among four models, the parameter scales of UNILM and mT5_small are almost at the same level with our kT5 model. It can be concluded from table that our model has considerable advantages in both four measures compared with similar-scale models, with at least a 5.8%, 0.7%, 5.5% and 4.4% lead in Rouge_1, Rouge_2, Rouge_L and BLUE respectively. Additionally, compared with mT5_base model, which is approximately 2 times larger than kT5, our model can still achieve better performance in all four measures.

| Model          | Param | Rouge_1 | Rouge_2 | Rouge_L | BLUE  |
|----------------|-------|---------|---------|---------|-------|
| UNILM [8]      | 350M  | 67.935  | 50.205  | 66.21   | 53.72 |
| mT5_small [36] | 300M  | 70.944  | 58.959  | 67.638  | 45.352|
| mT5_base [36]  | 580M  | 71.15   | 59.471  | 67.821  | 46.086|
| kT5            | 275M  | 76.756  | 59.716  | 73.259  | 49.814|

Based on above methods, we compare questions generated from triples in test set by kT5 model with those original questions in test set, and randomly select 10 pieces of data, as shown in table 4. It can be seen that questions generated by kT5 model are almost synonymous with those manual annotated questions. Additionally, it is worth mentioning that human annotators usually have some personal habits when generating questions. For example, some annotators are used to start a question with words like `告诉我 (Could you tell me)’ or ‘你知道 (Do you know)’. In comparison, questions generated by kT5 seem to be more standardized, because our model learns from the full training set, which can neutralize habits of different annotators and form a standard query format. Moreover, some typing errors may occur in the manual annotation process, such as typing one more character and typing a wrong character. In table 4, the question No.280 '你知道经纪公司的经纪是哪一个啊? (Do you know which is Ji Xinyu’s brokerage company?)’ is typed wrongly with one more character ‘经’. When generating questions by our model, these kind of mistakes can be avoided.

### 4.6 Few-Shot Learning

To better evaluate our kT5’s transferability to various domains, we compare the few-shot learning ability of our model and baseline model mT5_small (Table 5). It can be noticed that the performance of kT5 model is significantly better than the baseline under the equal model parameter condition. Especially for the case when the number of instances equals 100, kT5 improves baseline with approximate 57%, 43%, 53% and 37% in Rouge_1, Rouge_2, Rouge_L and BLUE respectively. Even in the cases of 200 instances and 300 instances, our model still leads baseline model with approximately 10% to 20% for all four measures.

There are mainly three reasons that the few-shot learning ability of our model in question generation task can far exceed the baseline model.

- Compared to mT5 model, kT5 is pre-trained on larger Chinese corpus, which can improve the ability of few-shot learning on tasks in Chinese language setting.
- Two pre-training tasks is beneficial for knowledge learning, which make our kT5 knowledge-enhanced. Specifically, the paragraph to question pre-training task improves model’s generation ability, and masked entity prediction task helps model understanding entities in a sentence better.
- kT5 enrich Chinese vocabulary with an extra word list, which avoids being affected by some rare Chinese characters and improve the domain transfer ability. For example, the character ‘鳜(pomfret)’ and both two characters of ‘猞猁(lynx)’ are not involved in the vocabulary of mT5 model, resulting in the question generation failure about those entities.

### 4.7 Human Evaluation

To better evaluate the quality of generated questions in semantic aspect, it is necessary to use human evaluation approaches. In this paper, we randomly sample 10 thousand questions to be manually evaluated. In details, two kinds of evaluation criteria are applied to measure questions’ semantic fluency and accuracy respectively. First, questions are rated from the range of 1 to 5 according to the fluency and semantics, where ‘1’ represents poor in both two aspects and ‘5’ means an excellent question that is both fluent and semantically clear. The average evaluation rate of those 10 thousand questions is 4.6, which suggests the high quality of generated questions. Second, questions are evaluated by a binary-measure, which uses ‘1’ to annotate a correctly generated question and ‘0’ in contrast. After human measurement, the final accuracy tare is 96%, approaching the performance of manual question generation.

Furthermore, we analysis those wrong samples in the human evaluation. It can be found that there are mainly two kinds of faults. The first one can be concluded as question generation error. For example, in Table 6, the knowledge tuple [川端康成集(Kawabata Yasunari collection); ISBN; 9787560215709], the relation ‘ISBN’ is due to the high appearance frequency of word ‘is’. The second type error is caused by the error occurred in extracting knowledge tuple from KG. As shown in Table 6, in the tuple [贾加乡 (Jiajia Town); 贾加乡 (Jiajia Town); 中华人民共和国 (the People’s Republic of China)], the extracted relation should be ‘位置(Location)’ instead of ‘贾加乡 (Jiajia Town)’.
Calculating Question Similarity is Enough: A New Method for KBQA Tasks
Conference '17, July 2017, Washington, DC, USA

Table 4: Question Generation Example

| Id | Question | Question Generation |
|----|----------|---------------------|
| 17 |  ‘A Global History’ like? | ‘A Global History’ like? |
| 49 |  where is the location of Liuhua Mountain? | where is the location of Liuhua Mountain? |
| 164 |  Shop Rui Xinyu’s brokerage company is? | Shop Rui Xinyu’s brokerage company is? |
| 280 |  Which brokerage company is Ji Xinyu in? | Which brokerage company is Ji Xinyu in? |

Table 5: Few-shot Learning

| Model | Instances | Rouge_1 | Rouge_2 | Rouge_L | BLUE |
|-------|-----------|---------|---------|---------|------|
| mT5_small | 100 | 15.603 | 11.082 | 45.159 | 46.355 |
| kT5 | 100 | 72.766 | 54.427 | 69.154 | 44.524 |
| mT5_small | 200 | 54.911 | 40.957 | 52.667 | 28.993 |
| kT5 | 200 | 72.188 | 54.542 | 68.996 | 44.8552 |
| mT5_small | 300 | 54.911 | 40.957 | 52.667 | 28.993 |
| kT5 | 300 | 72.188 | 54.542 | 68.996 | 44.8552 |

Table 6: Bad Case

| Type | Example |
|------|---------|
| 1 |  ‘A Global History’ like? | ‘A Global History’ like? |
| 2 |  Which brokerage company is Ji Xinyu in? | Which brokerage company is Ji Xinyu in? |

5 CONCLUSION AND FURTHER RESEARCH

5.1 Conclusion

Most present KBQA methods use pipeline to extract or represent the semantic information from the question and try to match those features with knowledge base to get the final prediction, which may accumulate the error in pipeline links. In addition, those hand-crafted feature representation or model architecture need careful design and hard to be robust to semantic difference.

In order to solve these problems, our work proposes a straightforward solution that utilize both pre-training model and knowledge graph. By using the pre-trained kT5 model, a large QA corpus is generated based on triples recalled from the knowledge graph, which is used for recall in our setting, however, this synthesized corpus can be used for QA research. Based on the corpus, we can directly get the final answer of a user’s question can be obtained by merely retrieving user’s query and questions in that corpus according to their similarity. We have examined that our method on the NLPCB KBQA dataset. Results show that the F1 of our method reaches 96.12%, which far exceeds the performances of other algorithms. Additionally, it is enough to answer user’s questions with just one-round retrieval when running our KBQA system online, which is deployment efficient and can reduce the server load and cost. In addition, we have proved that our KTS’s few shot learning ability can make it easy for various domain deployment without much training.

5.2 Further Research

Currently, we have only used several one-hop templates for corpus generation. In the future, we will study the generation of more complex questions, such as questions with multi-hop relations. For the corpus generation, we are going to design a more question-related generation mechanism, which can make the system more efficient and accurate.

REFERENCES

[1] Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. 2016. Constraint-Based Question Answering with Knowledge Graph. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. The COLING 2016 Organizing Committee, Osaka, Japan, 2503–2514.
[2] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic Parsing on Freebase from Question-Answer Pairs. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 1533–1544.
[3] Jonathan C. Brown, Gwen A. Frishkoff, and Maxine Eshkenazi. 2005. Automatic question generation for vocabulary assessment. In In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 819–826.
[4] Deng Cai and Wai Lam. 2020. Graph transformer for graph-to-sequence learning. In In Proceedings of the AAAI Conference on Artificial Intelligence. 7464–7471.
[5] Yongrui Chen, Huying Li, Yancheng Hua, and GuiLin Qin. 2020. Formal Query Building with Query Structure Prediction for Complex Question Answering over Knowledge Base. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, Yokohama, Japan, 3751–3758. https://doi.org/10.24963/ijcai.2020/519
[6] Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jinnsa Nayak, and Lun-Wei Ku. 2019. UHop: An Unrestricted-Hop Relation Extraction Framework for...
Knowledge-Based Question Answering. arXiv:1904.01246 [cs] (April 2019).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. n.d. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (n.d.), 16.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Lei Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. arXiv preprint arXiv:1905.03197 (2019).

Hady Elsaahar, Christophe Gravier, and Frederique Laforest. 2018. Zero-shot question generation from knowledge graphs for unseen predicates and entity types. In Proceedings of the 2018 Conference of the NAACL. 218–228.

Jiale Han, Bo Cheng, and Xu Wang. 2020. Two-Phase Hypergraph Based Reasoning with Dynamic Relations for Multi-Hop KBQA. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, Yokohama, Japan, 3655–3661. https://doi.org/10.24963/ijcai.2020/500

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Liangming Pan. 2019. Recent Progress of Deterministic Knowledge Graph Reasoning. arXiv:1905.08949 [cs] (May 2019). https://doi.org/10.18653/v1/2020.acl-main.91

Yue Hu and Guangyou Zhou. 2020. Question Generation from Knowledge Base with Graph Transformer. In The 19th China National Conference on Computational Linguistics. Computational linguistics Committee of Chinese society of China, 324–335.

Rik K. Kedziorzski, Dhanush Bekai, Yi Luan, Miirena Lapata, and Hannaneh Hajishirzi. 2019. Text Generation From Knowledge Graph With Graph Transformers. NAACL-HIT 1 (2019), 2284–2293.

Yuxuan Lai, Yang Liu, Jiahua Chen, Yansong Feng, and Dongyan Zhao. 2016. Open domain question answering system based on knowledge base. In Natural Language Understanding and Intelligent Applications. Springer, 722–733.

Yunhui Lan and Jing Jiang. 2020. Query Graph Generation for Answering Multi-hop Complex Questions from Knowledge Bases. In Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, 969–974. https://doi.org/10.18653/v1/2020.acl-main.91

Yunhui Lan, Shouzhong Wang, and Jing Jiang. 2019. Knowledge Base Question Answering with Topic Units. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, Macao, China, 5046–5052. https://doi.org/10.24963/ijcai.2019/701

Kai Lei, Yang Deng, Bing Zhang, and Ying Shen. 2017. Open domain question answering with character-level deep learning models. In 2017 10th International Symposium on Computational Intelligence and Design (ISCID), Vol. 2. IEEE, 30–33.

Chun-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In Text Summarization Branches Out. Association for Computational Linguistics, Barcelona, Spain, 74–81.

Aiting Liu, Ziqi Huang, Hengtong Lu, Xiaobing Liu, and Caixia Yuan. 2019. BB-KBQA: BERT-based knowledge base question answering. In China National Conference on Computational Linguistics. Springer, 81–92.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and David Klein. 1990. Introduction to Word Net: An on-line lexical database. International journal of lexicography 3, 4 (1990), 235–244.

Liangming Pang, Wallu Li, Tai-Seng Chua, and Qin Yin Kan. 2019. Recent Advances in Neural Question Generation. arXiv:1905.08949 [cs] (June 2019). https://doi.org/10.18653/v1/N19-1021

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL’02. Association for Computational Linguistics, Philadelphia, Pennsylvania, 311–318. https://doi.org/10.3115/107308.1073335

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Massively Multilingual Model. arXiv preprint arXiv:2010.11929 (2020).

Luke S. Zettlemoyer and Michael Collins. 2005. Learning to Map Sentences to Logical Forms. In Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence. 466–473.

Xuchen Yao and Benjamin Van Durme. 2014. Information extraction over structured data. In Association for Computational Linguistics (ACL).

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic Models for Knowledge Base Question Answering. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 960–967.

Shou Wang, Richong Zhang, Cheng Xu, and Yongyi Mao. 2018. The APVA-TURBO Approach To Question Answering in Knowledge Base. In Proceedings of the 27th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, 1998–2009.

Hady Elsahar, Christophe Gravier, and Frederique Laforest. 2018. Zero-shot learning with Graph Transformer. In Proceedings of the 2018 Conference on Natural Language Processing. 9066–9075.

Yue Wang, Binglong Zhang, Cheng Xu, and Yongyi Mao. 2018. The APVA-TURBO Approach To Question Answering in Knowledge Base. In Proceedings of the 27th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, 1998–2009.

John S. Whissell and Charles L. A. Clarke. 2011. Improving document clustering using Okapi BM25 feature weighting. Information Retrieval 14 (2011), 466–487.

Yuk W. Wong and Raymond J. Mooney. 2007. Learning Synchronous Grammars for Semantic Parsing with Lambda Calculus. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. 960–967.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Aparna Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Katou, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv:1609.08144 [cs] (Oct. 2016). https://arxiv.org/abs/1609.08144

Z. Xie, Z. Zhao, G. Zhou, and W. Wang. 2017. Topic enhanced deep structured semantic models for knowledge base question answering. Science China 11 (2017), 110103.

Kun Xu, Siva Reddy, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2016. Question Answering on Freebase via Relation Extraction and Textual Evidence. arXiv:1603.00957 [cs] (June 2016). arXiv:1603.00957

Lixing Xue, Noah Constant, Adam Roberts, Milare Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mT5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11924 (2020).

Xucheng Yao and Benjamin Van Derme. 2014. Information extraction over structured data: Question answering with freebase. In In Association for Computational Linguistics (ACL).

Wen-tai Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic Parsing by Staged Query Graph Generation: Question Answering with Knowledge Base. In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics. 1321–1331.

Shuang Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, Xiao Liu, Yukuo Cen, Xu Zou, Zhihui Yang, and Je Tang. 2021. WuDaoCorpora: A super large-scale Chinese corpora for pre-training language models. AI Open 2 (2021), 65–68. https://doi.org/10.1007/s42976-021-00090-4

Luke S. Zettlemoyer and Michael Collins. 2005. Learning to Map Sentences to Logical Form: Structured Classification with Probabilistic Categorical Grammars. In Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence. 456–466.

Yuyu Zhang, Junhan Dai, Zornitsa Kozareva, Alexander J. Smola, and Le Song. 2017. Variational Reasoning for Question Answering with Knowledge Graph. arXiv:1709.04071 [cs] (Nov. 2017). arXiv:1709.04071

Y. Zhang, G. Xu, Xingyu Fu, L. Jin, and T. Huang. 2020. Adversarial Training Improved Multi-Path Multi-Scale Relation Detector for Knowledge Base Question Answering. IEEE Access 8 (2020), 63310–63319.

B. Zhou, C. Sun, L. Lin, and B. Liu. 2018. LSTM Based Question Answering for Large Scale Knowledge Base. Acta Scientiarum Naturalium Universitatis Pekinensis (2018).