A Survey on Complex Knowledge Base Question Answering: Methods, Challenges and Solutions

Yunshi Lan[1*], **Gaole He**[2,3*], **Jinhao Jiang**[4], **Jing Jiang**[1], **Wayne Xin Zhao**[3,4†] and **Ji-Rong Wen**[2,3,4]
[1]School of Computing and Information Systems, Singapore Management University
[2]School of Information, Renmin University of China
[3]Beijing Key Laboratory of Big Data Management and Analysis Methods
[4]Gaoling School of Artificial Intelligence, Renmin University of China
{yslan, jingjiang}@smu.edu.sg, {hegaole, jrwen}@ruc.edu.cn, {batmanfly, jiangjinhaonlp}@gmail.com

Abstract

Knowledge base question answering (KBQA) aims to answer a question over a knowledge base (KB). Recently, a large number of studies focus on semantically or syntactically complicated questions. In this paper, we elaborately summarize the typical challenges and solutions for complex KBQA. We begin with introducing the background about the KBQA task. Next, we present the two mainstream categories of methods for complex KBQA, namely semantic parsing-based (SP-based) methods and information retrieval-based (IR-based) methods. We then review the advanced methods comprehensively from the perspective of the two categories. Specifically, we explicate their solutions to the typical challenges. Finally, we conclude and discuss some promising directions for future research.

1 Introduction

A knowledge base (KB) is a structured database that contains a collection of facts in the form (*subject*, *relation*, *object*). Large-scale KBs, such as Freebase [Bollacker *et al.*, 2008], DBPedia [Lehmann *et al.*, 2015] and Wikidata [Tanon *et al.*, 2016], have been constructed to serve many downstream tasks. Based on available KBs, knowledge base question answering (KBQA) is a task that aims to answer natural language questions with KBs as its knowledge source. Early work on KBQA [Bordes *et al.*, 2015; Dong *et al.*, 2015; Hu *et al.*, 2018a; Lan *et al.*, 2019b; Lan *et al.*, 2019a] focuses on answering a simple question, where only a single fact is involved. For example, "*Where was JK Rowling born?*" is a simple question which can be answered using just the fact "*(J.K. Rowling, birthplace, United Kingdom)*".

Recently, researchers start paying more attention to answering *complex questions* over KBs, i.e., the complex KBQA task [Hu *et al.*, 2018b; Luo *et al.*, 2018]. Complex questions usually contain multiple subjects, express compound relations and include numerical operations. Take the question in Figure 1 as an example. This example question starts with the subject "*The Jeff Probst Show*". Instead of querying a single fact, the question requires the composition of two relations, namely, "*nominee*" and "*spouse*". This query is also associated with an entity type constraint "*(Jeff Probst, is a, TV producer)*". The final answer should be further aggregated by selecting the possible candidates with the earliest marriage date. Generally, complex questions are questions involving *multi-hop* reasoning, *constrained* relations, *numerical* operations, or some combination of the above.

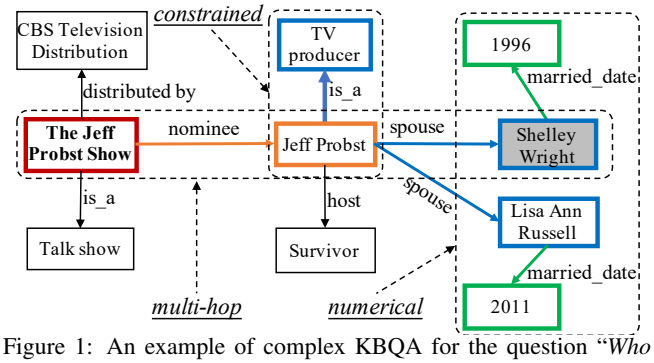

Figure 1: An example of complex KBQA for the question "*Who is the first wife of TV producer that was nominated for The Jeff Probst Show?*". We present the related KB subgraph for this question. The ground truth path to answer this question is annotated with colored borders. The topic entity and the answer entity are shown in the bold font and shaded box respectively. "multi-hop" reasoning, "constrained" relations and "numerical" operation are highlighted in black dotted box. We use different colors to indicate different reasoning hops to reach each entity from the topic entity.

Tracing back to the solutions for simple KBQA, a number of studies from two mainstream approaches have been proposed. These two approaches first recognize the subject in a question and link it to an entity in the KB (referred to as the *topic entity*). Then they derive the answers within the neighborhood of the topic entity by either executing a parsed logic form or reasoning in a question-specific graph extracted from the KB. The two categories of methods are commonly known as *semantic parsing-based methods* (SP-based methods) and *information retrieval-based meth-*

[*] Equal contribution.
[†] Corresponding author.

ods (IR-based methods) in prior work [Bordes *et al.*, 2015; Dong *et al.*, 2015; Hu *et al.*, 2018a; Gu *et al.*, 2020]. They include different working mechanisms to solve the KBQA task. The former approach represents a question by a symbolic logic form and then executes it against the KB and obtains the final answers. The latter approach constructs a question-specific graph delivering the comprehensive information related to the question and ranks all the entities in the extracted graph based on their relevance to the question.

However, when applying the two mainstream approaches to the complex KBQA task, complex questions bring in challenges on different parts of the approaches. We identify the main challenges as follows:

- Parsers used in existing SP-based methods are difficult to cover diverse complex queries (*e.g.,* multi-hop reasoning, constrained relations and numerical operations). Similarly, previous IR-based methods may fail to answer a complex query, as their ranking is performed over small-scope entities without traceable reasoning.
- More relations and subjects in complex questions indicate a larger search space of potential logic forms for parsing, which will dramatically increase the computational cost. Meanwhile, more relations and subjects could prevent IR-based methods from retrieving all relevant entities for ranking.
- Both approaches treat question understanding as a primary step. When questions become complicated in both semantic and syntactic aspects, models are required to have strong capabilities of natural language understanding and generalization.
- It is expensive to label the ground truth paths to the answers (see the example in Figure 1) for complex questions. Generally, only question-answer pairs are provided. This indicates SP-based methods and IR-based methods have to be trained without the annotation of correct logic forms and reasoning paths, respectively. Such weak supervision signals bring difficulties to both approaches.

Regarding the related surveys, we observe Wu *et al.* [2019] and Chakraborty *et al.* [2019] reviewed the existing work on simple KBQA. Furthermore, Fu *et al.* [2020] investigated the current advances on complex KBQA. They provided a general view of advanced methods only from the perspective of techniques and focused more on application scenarios in the e-commerce domain. Different from these surveys, our work tries to identify the challenges encountered in previous studies and extensively discusses existing solutions in a comprehensive and well-organized manner. Specifically, we categorize the methods for complex KBQA into two mainstream approaches based on their working mechanisms. We decompose the overall procedure of the two approaches into a series of modules and analyze the challenges in each module. We believe that this way is particularly helpful for readers to understand the challenges and how they are addressed in existing solutions to complex KBQA. Furthermore, we provide a thorough outlook on several promising research directions on complex KBQA.

Datasets	KB	Size	LF	NL
WebQuestions [Berant *et al.*, 2013]	Freebase	5,810	No	No
ComplexQuestions [Bao *et al.*, 2016]	Freebase	2,100	No	No
WebQuestionsSP [Yih *et al.*, 2016]	Freebase	4,737	Yes	Yes
ComplexWebQuestions [Talmor and Berant, 2018]	Freebase	34,689	Yes	Yes
QALD series [Lopez *et al.*, 2013]	DBpedia	-	Yes	Yes
LC-QuAD [Trivedi *et al.*, 2017]	DBpedia	5,000	Yes	Yes
LC-QuAD 2.0 [Dubey *et al.*, 2019]	DBpedia, Wikidata	30,000	Yes	Yes
MetaQA Vanilla [Zhang *et al.*, 2018]	WikiMovies	400k	No	No
CFQ [Keysers *et al.*, 2020]	Freebase	239,357	Yes	No
GrailQA [Gu *et al.*, 2020]	Freebase	64,331	Yes	Yes
KQA Pro [Shi *et al.*, 2020]	Wikidata	117,970	Yes	Yes

Table 1: Several complex KBQA benchmark datasets. "**LF**" denotes whether the dataset provides **L**ogic **F**orms, and "**NL**" denotes whether the dataset incorporates crowd workers to rewrite questions in **N**atural **L**anguage.

The remainder of this survey is organized as follows. We will first introduce the preliminary knowledge about the task formulation, multiple available datasets and evaluation protocol in Section 2. Next, we introduce the two mainstream categories of methods for complex KBQA in Section 3. Then following the categorization, we figure out typical challenges and solutions to these challenges in Section 4. Finally, we conclude and discuss some future research directions in Section 5.

2 Background

In this section, we first give a task definition about complex KBQA, and then introduce available datasets and evaluation protocol for this task.

Task. For the task of complex KBQA, a KB consisting of a set of facts is given as input, where the subject and object are connected by their relation. All the subjects and objects in the facts form the entity set of a KB. Given the available KB, this task aims to answer complex natural language questions in the format of a sequence of tokens. Specially, we assume the correct answers come from the entity set of the KB. Unlike answers of simple KBQA, which are entities directly connected to the topic entity, the answers of the complex KBQA task are entities multiple hops away from the topic entities or even some aggregation of them.

Datasets. Generally, the answers of the questions should be provided to train a complex KBQA system. For this purpose, many efforts have been devoted to constructing datasets for complex KBQA. We list the available complex KBQA datasets in Table 1. Overall, these datasets are constructed with the following steps. Given a topic entity in a KB as question subject, simple questions are first created with diverse templates. Based on simple questions and the neighborhood of a topic entity in a KB, complex questions are further generated with predefined templates, and some work [Shi *et al.*, 2020] also generates executable logic forms with templates. Meanwhile, answers are extracted with corresponding rules. In some cases, crowd workers are hired to paraphrase the template queries into natural language questions and refine the generated logic forms, making the question expressions more diverse and fluent. In order to serve realistic applications,

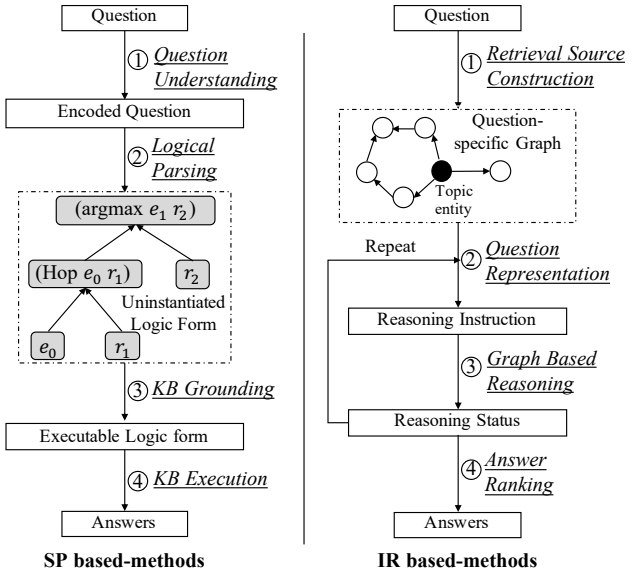

Figure 2: Illustration of two mainstream approaches for complex KBQA.

these datasets typically create questions which require multiple KB facts to reason. Moreover, they might include numerical operations (*e.g.,* counting, ranking operations for comparative or superlative questions) and constraints (*e.g.,* entity, temporal keywords), which further increase the difficulty in reasoning the answers from KBs.

Evaluation Protocol. The KBQA system usually predicts entities with the top confidence score to form the answer set. Note that there can be more than one answer to a question. In previous studies, there are some classical evaluation metrics such as *precision*, *recall*, F_1 and *Hits@1*. Some studies [Yih *et al.*, 2015; Liang *et al.*, 2017; Abujabal *et al.*, 2017] use the *precision*, *recall*, F_1 score to evaluate the prediction. *Precision* indicates the ratio of the correct answers over all the predicted answers. *Recall* is the ratio of the correct predicted answers over all the ground truth. And F_1 score considers *precision* and *recall* simultaneously. Some other methods [Miller *et al.*, 2016; Sun *et al.*, 2018; Xiong *et al.*, 2019; He *et al.*, 2021] use *Hits@1* to assess the fraction that the correct answers rank higher than other entities.

3 Two Mainstream Approaches

As introduced in Section 1, SP-based and IR-based methods are two mainstream approaches to solving complex KBQA task. SP-based methods parse a question into a logic form and execute it against KBs for finding the answers. IR-based methods retrieve a question-specific graph and apply some ranking algorithms to select entities from top positions. To summarize, the two approaches follow either a *parse-then-execute* paradigm or a *retrieval-and-rank* paradigm, which are illustrated in Figure 2.

Semantic Parsing-based Methods. This category of methods aims at parsing a natural language utterance into a logic form [Berant and Liang, 2014; Reddy *et al.*, 2014]. They predict answers via the following steps: (1) They fully understand a question via a *question understanding* module, which is to conduct the semantic and syntactic analysis and obtain an encoded question for the subsequent parsing step. (2) A *logical parsing* module is utilized to transfer the encoded question into an uninstantiated logic form. The uninstantiated logic form is a syntactic representation of the question without the grounding of entities and relations. The grammar and constituents of logic forms could be different according to specific designs of a system. (3) To execute against KBs, the logic form is further instantiated and validated by conducting some semantic alignments to structured KBs via *KB grounding*. Note that, in some work [Yih *et al.*, 2015; Liang *et al.*, 2017], the logical parsing and KB grounding are simultaneously performed, where logic forms are validated in KBs while partially parsed. (4) Eventually, the parsed logic form is executed against KBs to generate predicted answers via a *KB execution* module.

Information Retrieval-based Methods. As another mainstream approach, IR-based methods directly retrieve and rank answers from the KBs considering the information conveyed in the questions [Bordes *et al.*, 2015; Dong *et al.*, 2015]. They consist of the following steps: (1) Starting from the topic entity, the system first extracts a question-specific graph from KBs. Ideally, this graph includes all question related entities and relations as nodes and edges, respectively. Without explicitly generating an executable logic form, IR-based methods perform reasoning over the graph and then rank entities in the graph. (2) Next, the system encodes input questions via a *question representation* module. This module analyzes the semantics of the question and outputs reasoning instructions, which are usually represented as vectors. (3) A *graph-based reasoning* module conducts semantic matching via vector-based computation to propagate and then aggregate the information along the neighboring entities within the graph. The reasoning status, which has diverse definitions in different methods (*e.g.,* distributions of predicted entities, representations of relations), is updated based on the reasoning instruction. Recently, several studies [Jain, 2016; Chen *et al.*, 2019] repeat Step (2) and (3) for multiple times to perform the reasoning. (4) An *answer ranking* module is utilized to rank the entities in the graph according to the reasoning status at the end of reasoning. The top-ranked entities are predicted as the answers to the question.

Pros and Cons. Overall, SP-based methods can produce a more interpretable reasoning process by generating expressive logic forms. However, they heavily rely on the design of the logic form and parsing algorithm, which turns out to be the bottleneck of performance improvement. As a comparison, IR-based methods conduct complex reasoning on graph structure and perform semantic matching. Such a paradigm naturally fits into popular end-to-end training and makes the IR-based methods easier to train. However, the blackbox style of the reasoning model makes the intermediate reasoning less interpretable.

Categories	Modules	Challenges	Solutions
SP-based Methods	Question understanding	Understanding complex semantics and syntax	Adopt structural properties (*e.g.,* dependency parsing [Abujabal *et al.*, 2017; Abujabal *et al.*, 2018; Luo *et al.*, 2018], AMR [Kapanipathi *et al.*, 2020]) augmented parsing, skeleton-based parsing [Sun *et al.*, 2020] or structural properties based matching [Maheshwari *et al.*, 2019; Zhu *et al.*, 2020; Chen *et al.*, 2020].
	Logical parsing	Parsing complex queries	Develop expressive targets for parsing, such as: template based queries [Bast and Haussmann, 2015], query graph [Yih *et al.*, 2015; Abujabal *et al.*, 2017; Hu *et al.*, 2018b], and so on.
	KB grounding	Grounding with large search space	Narrow down search space by decompose-execute-join strategy [Zheng *et al.*, 2018b; Bhutani *et al.*, 2019] or expand-and-rank strategy [Chen *et al.*, 2019; Lan *et al.*, 2019c; Lan and Jiang, 2020].
	Training procedure	Training under weak supervision signals	Adopt reward shaping strategy to strengthen training signal [Saha *et al.*, 2019; Hua *et al.*, 2020b; Qiu *et al.*, 2020b], conduct pre-training to initialize the model [Qiu *et al.*, 2020b] or iterative maximum-likelihood training[Liang *et al.*, 2017].
IR-based Methods	Retrieval source construction	Reasoning under incomplete KB	Supplement KB with extra corpus [Sun *et al.*, 2018; Sun *et al.*, 2019], fuse extra textual information into entity representations [Xiong *et al.*, 2019; Han *et al.*, 2020a] or leverage KB embeddings [Saxena *et al.*, 2020].
	Question representation	Understanding complex semantics	Update with reasoned information [Miller *et al.*, 2016; Zhou *et al.*, 2018; Xu *et al.*, 2019], dynamic attention over the question [He *et al.*, 2021] or enrich the question representation with contextual information of graph [Sun *et al.*, 2018].
	Graph based reasoning	Uninterpretable reasoning	Provide traceable reasoning path [Zhou *et al.*, 2018; Xu *et al.*, 2019] or hyperedge based reasoning [Han *et al.*, 2020b].
	Training procedure	Training under weak supervision signals	Provide shaped reward as intermediate feedback [Qiu *et al.*, 2020a], augment intermediate supervision signals with bidirectional search algorithm [He *et al.*, 2021] or adopt variational algorithm to train entity linking module [Zhang *et al.*, 2018].

Table 2: Summary of the existing studies on complex KBQA. We categorize them into two mainstream approaches *w.r.t.* key modules and solutions according to different challenges.

4 Challenges and Solutions

Since the aforementioned approaches are developed based on different paradigms, we describe the challenges and corresponding solutions for complex KBQA with respect to the two mainstream approaches. A summary of these challenges and solutions is presented in Table 2.

4.1 Semantic Parsing-based Methods

In this part, we discuss the challenges and solutions for semantic parsing-based methods.

Overview. As introduced in Section 3, SP-based methods follow a parse-then-execute procedure via a series of modules, namely question understanding, logical parsing, KB grounding and KB execution. These modules will encounter different challenges for complex KBQA. Firstly, question understanding becomes more difficult when the questions are complicated in both semantic and syntactic aspects. Secondly, logical parsing has to cover diverse query types of complex questions. Moreover, a complex question involving more relations and subjects will dramatically increase the possible search space for parsing, which makes the parsing less effective. Thirdly, the manual annotation of logic forms are both expensive and labor-intensive, and it is challenging to train a SP-based method with weak supervision signals (*i.e.,* question-answer pairs). Next, we will introduce how prior studies deal with these challenges.

Understanding Complex Semantics and Syntax. As the first step of SP-based methods, question understanding module converts unstructured text into encoded question (*i.e.,* structural representation), which benefits the downstream parsing. Compared with simple questions, complex questions are featured with more complex query types and compositional semantics, which increases the difficulty in linguistic analysis. To better understand complex natural language questions, many existing methods rely on syntactic parsing, such as dependencies [Abujabal *et al.*, 2017; Abujabal *et al.*, 2018; Luo *et al.*, 2018] and Abstract Meaning Representation (AMR) [Kapanipathi *et al.*, 2020], to provide better alignment between question constituents and logic form elements (*e.g.,* entity, relation, entity types and attributes). However, the accuracy of producing syntactic parsing is still not satisfying on complex questions, especially for those with long-distance dependency. To alleviate error propagation from syntactic parsing to downstream semantic parsing, Sun *et al.* [2020] leveraged the skeleton-based parsing to obtain the trunk of a complex question, which is a simple question with several branches (*i.e.,* pivot word of original text-spans) to be expanded. Another line of work focuses on leveraging structural properties (such as tree structure or graph structure) of logic forms for ranking candidate parsing. They try to improve the matching between logic forms and questions by incorporating structure-aware feature encoder [Zhu *et al.*, 2020], applying fine-grained slot matching [Maheshwari *et al.*, 2019], and adding constraints about query structure to filter noisy queries out [Chen *et al.*, 2020].

Parsing Complex Queries. During parsing, traditional semantic parses (*e.g.,* CCG [Cai and Yates, 2013; Kwiatkowski *et al.*, 2013; Reddy *et al.*, 2014]), which are developed without considering KB schemas, have shown their potential in parsing simple questions. However, these methods could be sub-optimal for complex questions due to the ontology mismatching problem [Yih *et al.*, 2015]. Thus, it is necessary to leverage the structure of KBs for more accurate parsing.

To satisfy the compositionality of the complex questions, researchers have developed diverse expressive logic forms as parsing targets. Bast and Haussmann [2015] designed three query templates as the parsing targets, which could cover questions querying 1-hop, 2-hop relations and single constraint involved relations. Although this piece of work can successfully parse several types of complex questions, it suffers from the limited coverage issue. Yih *et al.* [2015] proposed *query graph* as the expressive parsing target. A query graph is a logic form in graph structure which closely matches with the KB schemas. Such query graphs have shown strong expression capability in complex KBQA task. However, they are restrictedly generated with predefined manual rules, which is inapplicable to large-scale datasets and long-tail complex question types. The follow-up work tried to improve the formulation of query graphs. To generalize to unseen and long-tail question types, Ding *et al.* [2019] proposed to leverage frequent query substructure for formal query generation. Abujabal *et al.* [2017] utilized syntactic annotation to enhance the structural complexity of the query graph. Hu *et al.* [2018b] applied more aggregation operators (*e.g.,* "merging") to fit complex questions, and conducted coreference resolution.

Grounding with Large Search Space. To obtain executable logic forms, KB grounding module instantiates possible logic forms with a KB. As one entity in the KB could be linked to hundreds or even thousands of relations, it would be unaffordable to explore and ground all the possible logic forms for a complex question considering both computational resource and time complexity. Recently, researchers proposed multiple approaches to solving the problem. Zheng *et al.* [2018b] proposed to decompose a complex question into multiple simple questions, where each question was parsed into a simple logic form. Next, intermediate answers are generated via these simple logic forms and final answers are jointly obtained. This *decompose-execute-join* strategy could effectively narrow down the search space. A similar approach was studied by Bhutani *et al.* [2019] and they reduced human annotations by leveraging dependency structure. Meanwhile, a number of studies adopted the *expand-and-rank* strategies to reduce the search space by searching the logic forms with beams. Chen *et al.* [2019] first adopted the hopwise greedy search strategy to expand the most likely query graphs and stop until the best query graph was obtained. Lan *et al.* [2019c] proposed an iterative matching module to parse the questions without revisiting the generated query graphs at each searching step. Such a sequential expansion process is only effective in answering multi-hop questions, while helpless for questions with constraints or numerical operations. Lan and Jiang [2020] defined more operations to support three typical complex queries, which can largely reduce the search space.

Training under Weak Supervision Signals. To deal with the issue of limited or insufficient training data, Reinforcement Learning (RL) based optimization has been adopted to maximize the expected reward [Liang *et al.*, 2017; Qiu *et al.*, 2020b]. In such a way, SP-based methods can only receive the feedback after the execution of the complete parsed logical form, which leads to severe sparse positive rewards and data inefficiency issues. To tackle these issues, some research work adopted *reward shaping* strategies for parsing evaluation. Saha *et al.* [2019] rewarded the model by the additional feedback when the predicted answers are the same type as the ground truth. Hua *et al.* [2020b] adopted a similar idea to evaluate the generated logic form by comparing it with the high-reward logic forms stored in the memory buffer. Besides rewards for the whole procedure, intermediate rewards during the semantic parsing process may also help address this challenge. Recently, Qiu *et al.* [2020b] formulated query graph generation as a hierarchical decision problem, and proposed a framework based on hierarchical RL with intrinsic motivations to provide intermediate rewards. To accelerate and stablize the training process, Qiu *et al.* [Qiu *et al.*, 2020b] *pre-trained* model with pseudo-gold programs (*i.e.,* high-reward logic forms generated by hand-crafted rules). As pseudo-gold programs can be also generated from the model, Liang *et al.* [2017] proposed to maintain pseudo-gold programs found by an iterative maximum-likelihood training process to bootstrap training.

4.2 Information Retrieval-based Methods

Here, we summarize the main challenges brought by complex questions for different modules of IR-based methods.

Overview. The overall procedure typically consists of the modules of retrieval source construction, question representation, graph based reasoning and answer ranking. These modules will encounter different challenges for complex KBQA. Firstly, the retrieval source construction module extracts a question-specific graph from KBs, which covers a wide range of relevant facts for each question. Due to unneglectable incompleteness of source KBs [Min *et al.*, 2013], the correct reasoning paths may be absent from the extracted graph. This issue is more likely to occur in the case of complex questions. Secondly, question representation module understands the question and generates instructions to guide the reasoning process. This step becomes challenging when the question is complicated. After that, reasoning on graph is conducted through semantic matching. When dealing with complex questions, such methods rank answers through semantic similarity without traceable reasoning in the graph, which hinders reasoning analysis and failure diagnosis. Eventually, this system encounters the same training challenge under weak supervision signals (*i.e.,* question-answer pairs). The following parts illustrate how prior work deal with these challenges.

Reasoning under Incomplete KB. IR-based methods first extract a question-specific graph from KBs, and conduct subsequent reasoning on it. Since simple questions only require 1-hop reasoning on the neighborhood of topic entity in KBs, IR-based methods are less likely to suffer from the inherent incompleteness of KBs [Min *et al.*, 2013]. In comparison, it may be a severe problem for complex questions, where the correct reasoning path may be absent from the question-specific graph. Furthermore, this incompleteness reduces the neighborhood information used for encoding entities, which poses additional challenges for effective reasoning. To tackle this challenge, researchers utilize auxiliary information to enrich the knowledge source. Intuitively, large question-related

text corpus retrieved from Wikipedia can provide a wide range of unstructured knowledge as supplementary evidence. Sun *et al.* [2018] and Sun *et al.* [2019] proposed to complement the subgraph extracted from incomplete KBs with extra question-related text sentences to form a heterogeneous graph and conduct reasoning on it. Instead of directly complementing sentences to question-specific graph as nodes, Xiong *et al.* [2019] and Han *et al.* [2020a] proposed to fuse extra textual information into the entity representation to supplement knowledge. They first encoded sentences and entities conditioned on questions, and then supplemented the incomplete KB by aggregating representations of sentences to enhance corresponding entity representations. Besides extra text corpus, knowledge base embeddings have been adopted to alleviate the sparsity of KB by performing missing link prediction. Inspired by KB completion task, Saxena *et al.* [2020] utilized pre-trained knowledge base embeddings to enrich the learned entity representations and address incomplete KB issue.

Understanding Complex Semantics. In general, IR-based methods generate reasoning instructions by directly encoding questions as low-dimensional vectors through neural network (*e.g.,* LSTM). Static reasoning instructions obtained through above approaches cannot effectively represent the compositional semantics of complex questions. In order to comprehensively understand questions, recent work dynamically updated the reasoning instructions during the reasoning process. To focus on the currently unanalyzed part of question, Miller *et al.* [2016], Zhou *et al.* [2018] and Xu *et al.* [2019] proposed to update the reasoning instruction with information retrieved along the reasoning process. Besides updating the instruction representation with the reasoned information, He *et al.* [2021] proposed to focus on different parts of the question with dynamic attention mechanism. This dynamic attention mechanism can promote the model to attend to other information conveyed by the question and provide proper guidance for subsequent reasoning steps. Instead of decomposing the semantics of questions, Sun *et al.* [2018] proposed to augment the representation of the question with contextual information from graph. They updated the reasoning instruction through aggregating information from the topic entity after every reasoning step.

Uninterpretable Reasoning. Traditional IR-based methods rank answers by calculating a single semantic similarity between questions and entities in the graph, which is less interpretable at the intermediate steps. As the complex questions usually query multiple facts, the system is supposed to accurately predict answers over the graph based on a traceable and observable reasoning process. Even though some work repeated reasoning steps for multiple times, they cannot reason along a traceable path in the graph. To derive a more interpretable reasoning process, multi-hop reasoning is introduced. Specifically, Zhou *et al.* [2018] and Xu *et al.* [2019] proposed to make the relation or entity predicted at each hop traceable and observable. They output intermediate predictions (*i.e.,* matched relations or entities) from predefined memory as the interpretable reasoning path. Nevertheless, it can not fully utilize the semantic relation information to reason edge by edge. Thus, Han *et al.* [2020b] constructed a denser hypergraph by pinpointing a group of entities connected via same relation, which simulated human's hopwise relational reasoning and output a sequential relation path to make the reasoning interpretable.

Training under Weak Supervision Signals. Similar to the SP-based methods, it is difficult for IR-based methods to reason the correct answers without any annotations at intermediate steps, since the model cannot receive any feedback until the end of reasoning. It is found that this case may lead to spurious reasoning [He *et al.*, 2021]. To mitigate such issues, Qiu *et al.* [2020a] formulated the reasoning process over KBs as expanding the reasoning path on KB and adopted reward shaping strategy to provide intermediate rewards. To evaluate reasoning paths at intermediate steps, they utilized semantic similarity between the question and the reasoning path to provide feedback. Besides evaluating the reasoning path at intermediate steps, a more intuitive idea is to infer pseudo intermediate status and augment model training with such inferred signals. Inspired by bidirectional search algorithm on graph, He *et al.* [2021] proposed to learn the intermediate reasoning entity distributions by synchronizing bidirectional reasoning process. While most of existing work focused on enhancing the supervision signals at intermediate steps, few work paid attention to the entity linking step. Researchers utilized off-the-shelf tools to locate the topic entity in question, which may cause error propagation to subsequent reasoning. In order to accurately locate the topic entity without annotations, Zhang *et al.* [2018] proposed to train entity linking module through a variational learning algorithm which jointly modeled topic entity recognition and subsequent reasoning over KBs.

5 Conclusion and Future Directions

This paper attempted to provide an overview of typical challenges and corresponding solutions on complex KBQA. We introduced commonly used datasets and summarized the widely employed SP-based methods as well as IR-based methods. Existing complex KBQA methods are generally summarized into these two categories. Besides them, some other methods [Talmor and Berant, 2018] may not fall into these two categories. For example, Talmor and Berant [2018] proposed to transform a complex question to a composition of simple questions through rule-based decomposition, which focused on question decomposition instead of KB based reasoning or logic form generation. We believe that complex KBQA will continue to be an active and promising research area with wide applications, such as natural language understanding, compositional generalization, multi-hop reasoning. Many challenges presented in this survey are still open and under-explored.

Considering the challenges summarized in this paper, we point out several promising future directions for complex KBQA task:

Evolutionary KBQA. As we can see, existing methods for complex KBQA task are usually learned on offline training datasets and then deployed online to answer user questions. Due to such clear separation, most of existing KBQA systems fail to catch up with the rapid growth of world knowl-

edge and answer new questions. However, user feedback may provide deployed KBQA systems an opportunity to improve themselves. Based on this observation, Abujabal *et al.* [2018] leveraged the user feedback to rectify answers generated by the KBQA system and made further improvement. Despite verifying the correctness of system prediction, users may also play a more active role in the question answering process. Zheng *et al.* [2018a] designed an interactive method to engage users in the question parsing process of the KBQA system directly. In the future, an evolutionary KBQA system is imperative to get continuous improvement after online deployment.

Robust and Interpretable Models. While existing methods have achieved promising results on benchmark datasets where *i.i.d* assumption is held, they may easily fail to deal with an out-of-distribution case. Few-shot setting is a scenario where the training data is limited. A few previous studies [Hua *et al.*, 2020a; He *et al.*, 2021] discussed related topics, but they are still far from comprehensive in terms of challenge anslysis and problem solving. Compositional generalization is another scenario where the novel combinations of component items seen in training should be inferred during testing. To support more research on such issue, Gu *et al.* [2020] and Keysers *et al.* [2020] have introduced related datasets, namely GrailQA and CFQ. The models are supposed to handle out-of-distribution questions and obtain explainable reasoning process. Designing methods for KBQA with good interpretability and robustness may be a challenging but promising topic for future research.

More General Knowledge Base. Due to KB incompleteness, researchers incorporated extra information (such as text [Sun *et al.*, 2018], images [Xie *et al.*, 2017] and human interactions [He *et al.*, 2020]) to complement the knowledge base, which would further improve the complex KBQA performance. There are also some tasks (*e.g.,* visual question answering and commonsense knowledge reasoning), which can be formulated as question answering based on specific KBs. For example, in visual question answering, the scene graph extracted from an image can be regarded as a special KB [Hudson and Manning, 2019]. Despite explicitly representing relational knowledge as the structural KB, some researchers proposed to reason on implicit "KB". Petroni *et al.* [2019] analyzed the relational knowledge in a wide range of pretrained language models, and some follow-up work [Bouraoui *et al.*, 2020; Jiang *et al.*, 2020] further demonstrated its effectiveness to answer "fill-in-the-blank" cloze statements. While most of existing work focused on traditional structured KBs, a more general definition about KBs and flexible usage of KBs may help KBQA research topic show greater impact.

Acknowledgements

This work is partially supported by the National Research Foundation, Singapore under its International Research Centres in Singapore Funding Initiative, the National Natural Science Foundation of China under Grant No. 61872369 and 61832017, Beijing Academy of Artificial Intelligence (BAAI) under Grant No. BAAI2020ZJ0301 and Beijing Outstanding Young Scientist Program under Grant No. BJJWZYJH012019100020098. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore. Wayne Xin Zhao is the corresponding author.

References

[Abujabal *et al.*, 2017] Abdalghani Abujabal, Mohamed Yahya, Mirek Riedewald, and Gerhard Weikum. Automated template generation for question answering over knowledge graphs. In *WWW*, 2017.

[Abujabal *et al.*, 2018] Abdalghani Abujabal, Rishiraj Saha Roy, Mohamed Yahya, and Gerhard Weikum. Never-ending learning for open-domain question answering over knowledge bases. In *WWW*, 2018.

[Bao *et al.*, 2016] Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. Constraint-based question answering with knowledge graph. In *COLING*, 2016.

[Bast and Haussmann, 2015] Hannah Bast and Elmar Haussmann. More accurate question answering on freebase. In *CIKM*, 2015.

[Berant and Liang, 2014] Jonathan Berant and Percy Liang. Semantic parsing via paraphrasing. In *ACL*, 2014.

[Berant *et al.*, 2013] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *EMNLP*, 2013.

[Bhutani *et al.*, 2019] Nikita Bhutani, Xinyi Zheng, and H. V. Jagadish. Learning to answer complex questions over knowledge bases with query composition. In *CIKM*, 2019.

[Bollacker *et al.*, 2008] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: A collaboratively created graph database for structuring human knowledge. In *SIGMOD*, 2008.

[Bordes *et al.*, 2015] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *arXiv*, 2015.

[Bouraoui *et al.*, 2020] Zied Bouraoui, José Camacho-Collados, and Steven Schockaert. Inducing relational knowledge from BERT. In *AAAI*, 2020.

[Cai and Yates, 2013] Qingqing Cai and Alexander Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *ACL*, 2013.

[Chakraborty *et al.*, 2019] Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. Introduction to neural network based approaches for question answering over knowledge graphs. *arXiv*, 2019.

[Chen *et al.*, 2019] Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jijnasa Nayak, and Lun-Wei Ku. UHop: An unrestricted-hop relation extraction framework for knowledge-based question answering. In *NAACL*, 2019.

[Chen *et al.*, 2020] Yongrui Chen, Huiying Li, Yuncheng Hua, and Guilin Qi. Formal query building with query structure prediction for complex question answering over knowledge base. In *IJCAI*, 2020.

[Ding *et al.*, 2019] Jiwei Ding, Wei Hu, Qixin Xu, and Yuzhong Qu. Leveraging frequent query substructures to generate formal queries for complex question answering. In *EMNLP*, 2019.

[Dong *et al.*, 2015] Li Dong, Furu Wei, Ming Zhou, and Ke Xu. Question answering over Freebase with multi-column convolutional neural networks. In *ACL*, 2015.

[Dubey *et al.*, 2019] Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia. In *ISWC*, 2019.

[Fu *et al.*, 2020] Bin Fu, Yunqi Qiu, Chengguang Tang, Yang Li, Haiyang Yu, and Jian Sun. A survey on complex question answering over knowledge base: Recent advances and challenges. *arXiv*, 2020.

[Gu *et al.*, 2020] Yu Gu, Sue Kase, Michelle Vanni, Brian M. Sadler, Percy Liang, Xifeng Yan, and Yu Su. Beyond I.I.D.: three levels of generalization for question answering on knowledge bases. In *WWW*, 2020.

[Han *et al.*, 2020a] Jiale Han, Bo Cheng, and Xu Wang. Open domain question answering based on text enhanced knowledge graph with hyperedge infusion. In *EMNLP*, 2020.

[Han *et al.*, 2020b] Jiale Han, Bo Cheng, and Xu Wang. Two-phase hypergraph based reasoning with dynamic relations for multi-hop kbqa. In *IJCAI*, 2020.

[He *et al.*, 2020] Gaole He, Junyi Li, Wayne Xin Zhao, Peiju Liu, and Ji-Rong Wen. Mining implicit entity preference from user-item interaction data for knowledge graph completion via adversarial learning. In *WWW*, 2020.

[He *et al.*, 2021] Gaole He, Yunshi Lan, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *WSDM*, 2021.

[Hu *et al.*, 2018a] Sen Hu, Lei Zou, Jeffrey Xu Yu, Hai Xun Wang, and Donyan Zhao. Answering natural language questions by subgraph matching over knowledge graphs. *TKDE*, 2018.

[Hu *et al.*, 2018b] Sen Hu, Lei Zou, and Xinbo Zhang. A state-transition framework to answer complex questions over knowledge base. In *EMNLP*, 2018.

[Hua *et al.*, 2020a] Yuncheng Hua, Yuan-Fang Li, Gholamreza Haffari, Guilin Qi, and Wei Wu. Retrieve, program, repeat: Complex knowledge base question answering via alternate meta-learning. In *IJCAI*, 2020.

[Hua *et al.*, 2020b] Yuncheng Hua, Yuan-Fang Li, Guilin Qi, Wei Wu, Jingyao Zhang, and Daiqing Qi. Less is more: Data-efficient complex question answering over knowledge bases. *J. Web Semant.*, 2020.

[Hudson and Manning, 2019] Drew A. Hudson and Christopher D. Manning. Learning by abstraction: The neural state machine. In *NeurIPS*, 2019.

[Jain, 2016] Sarthak Jain. Question answering over knowledge base using factual memory networks. In *NAACL*, 2016.

[Jiang *et al.*, 2020] Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know. *TACL*, 2020.

[Kapanipathi *et al.*, 2020] Pavan Kapanipathi, Ibrahim Abdelaziz, Srinivas Ravishankar, Salim Roukos, Alexander G. Gray, Ramón Fernandez Astudillo, Maria Chang, Cristina Cornelio, Saswati Dana, Achille Fokoue, Dinesh Garg, Alfio Gliozzo, Sairam Gurajada, Hima Karanam, Naweed Khan, Dinesh Khandelwal, Young-Suk Lee, Yunyao Li, Francois P. S. Luus, Ndivhuwo Makondo, Nandana Mihindukulasooriya, Tahira Naseem, Sumit Neelam, Lucian Popa, Revanth Gangi Reddy, Ryan Riegel, Gaetano Rossiello, Udit Sharma, G. P. Shrivatsa Bhargav, and Mo Yu. Question answering over knowledge bases by leveraging semantic parsing and neuro-symbolic reasoning. In *AAAI*, 2020.

[Keysers *et al.*, 2020] Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, Dmitry Tsarkov, Xiao Wang, Marc van Zee, and Olivier Bousquet. Measuring compositional generalization: A comprehensive method on realistic data. In *ICLR*, 2020.

[Kwiatkowski *et al.*, 2013] Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. Scaling semantic parsers with on-the-fly ontology matching. In *EMNLP*, 2013.

[Lan and Jiang, 2020] Yunshi Lan and Jing Jiang. Query graph generation for answering multi-hop complex questions from knowledge bases. In *ACL*, 2020.

[Lan *et al.*, 2019a] Yunshi Lan, Shuohang Wang, and Jing Jiang. Knowledge base question answering with a matching-aggregation model and question-specific contextual relations. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 27:1629–1638, 2019.

[Lan *et al.*, 2019b] Yunshi Lan, Shuohang Wang, and Jing Jiang. Knowledge base question answering with topic units. In *IJCAI*, 2019.

[Lan *et al.*, 2019c] Yunshi Lan, Shuohang Wang, and Jing Jiang. Multi-hop knowledge base question answering with an iterative sequence matching model. In *ICDM*, 2019.

[Lehmann *et al.*, 2015] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 2015.

[Liang *et al.*, 2017] Chen Liang, Jonathan Berant, Quoc Le, Kenneth D. Forbus, and Ni Lao. Neural symbolic ma-

chines: Learning semantic parsers on Freebase with weak supervision. In *ACL*, 2017.

[Lopez *et al.*, 2013] Vanessa Lopez, Christina Unger, Philipp Cimiano, and Enrico Motta. Evaluating question answering over linked data. *Web Semantics Science Services And Agents On The World Wide Web*, 2013.

[Luo *et al.*, 2018] Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Q. Zhu. Knowledge base question answering via encoding of complex query graphs. In *EMNLP*, 2018.

[Maheshwari *et al.*, 2019] Gaurav Maheshwari, Priyansh Trivedi, Denis Lukovnikov, Nilesh Chakraborty, Asja Fischer, and Jens Lehmann. Learning to rank query graphs for complex question answering over knowledge graphs. In *ISWC*, 2019.

[Miller *et al.*, 2016] Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. In *EMNLP*, 2016.

[Min *et al.*, 2013] Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. Distant supervision for relation extraction with an incomplete knowledge base. In *NAACL-HLT*, 2013.

[Petroni *et al.*, 2019] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. Language models as knowledge bases? In *EMNLP*, 2019.

[Qiu *et al.*, 2020a] Yunqi Qiu, Yuanzhuo Wang, Xiaolong Jin, and Kun Zhang. Stepwise reasoning for multi-relation question answering over knowledge graph with weak supervision. In *WSDM*, 2020.

[Qiu *et al.*, 2020b] Yunqi Qiu, Kun Zhang, Yuanzhuo Wang, Xiaolong Jin, Long Bai, Saiping Guan, and Xueqi Cheng. Hierarchical query graph generation for complex question answering over knowledge graph. In *CIKM*, 2020.

[Reddy *et al.*, 2014] Siva Reddy, Mirella Lapata, and Mark Steedman. Large-scale semantic parsing without question-answer pairs. *TACL*, 2014.

[Saha *et al.*, 2019] Amrita Saha, Ghulam Ahmed Ansari, Abhishek Laddha, Karthik Sankaranarayanan, and Soumen Chakrabarti. Complex program induction for querying knowledge bases in the absence of gold programs. *TACL*, 2019.

[Saxena *et al.*, 2020] Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *ACL*, 2020.

[Shi *et al.*, 2020] Jiaxin Shi, Shulin Cao, Liangming Pan, Yutong Xiang, Lei Hou, Juanzi Li, Hanwang Zhang, and Bin He. Kqa pro: A large diagnostic dataset for complex question answering over knowledge base. *arXiv*, 2020.

[Sun *et al.*, 2018] Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. Open domain question answering using early fusion of knowledge bases and text. In *EMNLP*, 2018.

[Sun *et al.*, 2019] Haitian Sun, Tania Bedrax-Weiss, and William Cohen. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. In *EMNLP*, 2019.

[Sun *et al.*, 2020] Yawei Sun, Lingling Zhang, Gong Cheng, and Yuzhong Qu. SPARQA: skeleton-based semantic parsing for complex questions over knowledge bases. In *AAAI*, 2020.

[Talmor and Berant, 2018] Alon Talmor and Jonathan Berant. The web as a knowledge-base for answering complex questions. In *NAACL-HLT*, 2018.

[Tanon *et al.*, 2016] Thomas Pellissier Tanon, Denny Vrandečić, Sebastian Schaffert, Thomas Steiner, and Lydia Pintscher. From freebase to wikidata: The great migration. In *WWW*, 2016.

[Trivedi *et al.*, 2017] Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. Lc-quad: A corpus for complex question answering over knowledge graphs. In *ISWC*, 2017.

[Wu *et al.*, 2019] Peiyun Wu, Xiaowang Zhang, and Zhiyong Feng. A survey of question answering over knowledge base. In *CCKS*, 2019.

[Xie *et al.*, 2017] Ruobing Xie, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. Image-embodied knowledge representation learning. In *IJCAI*, pages 3140–3146, 2017.

[Xiong *et al.*, 2019] Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. Improving question answering over incomplete kbs with knowledge-aware reader. In *ACL*, 2019.

[Xu *et al.*, 2019] Kun Xu, Yuxuan Lai, Yansong Feng, and Zhiguo Wang. Enhancing key-value memory neural networks for knowledge based question answering. In *NAACL-HLT*, 2019.

[Yih *et al.*, 2015] Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *ACL*, 2015.

[Yih *et al.*, 2016] Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In *ACL*, 2016.

[Zhang *et al.*, 2018] Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander J Smola, and Le Song. Variational reasoning for question answering with knowledge graph. In *AAAI*, 2018.

[Zheng *et al.*, 2018a] Weiguo Zheng, Hong Cheng, Jeffrey Xu Yu, Lei Zou, and Kangfei Zhao. Never-ending learning for open-domain question answering over knowledge bases. In *InfoScience*, 2018.

[Zheng *et al.*, 2018b] Weiguo Zheng, Jeffrey Xu Yu, Lei Zou, and Hong Cheng. Question answering over knowledge graphs: Question understanding via template decomposition. In *VLDB Endow.*, 2018.

[Zhou *et al.*, 2018] Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. An interpretable reasoning network for multi-relation question answering. In *COLING*, 2018.

[Zhu *et al.*, 2020] Shuguang Zhu, Xiang Cheng, and Sen Su. Knowledge-based question answering by tree-to-sequence learning. *Neurocomputing*, 2020.