Knowledge Transfer between Structured and Unstructured Sources for Complex Question Answering

Lingbo Mo,[*] Zhen Wang[*], Jie Zhao, Huan Sun
The Ohio State University
{mo.169,wang.9215,zhao.1359,sun.397}@osu.edu

Abstract

Multi-hop question answering (QA) combines multiple pieces of evidence to search for the correct answer. Reasoning over a text corpus (TextQA) and/or a knowledge base (KBQA) has been extensively studied and led to distinct system architectures. However, knowledge transfer between such two QA systems has been under-explored. Research questions like what knowledge is transferred or whether the transferred knowledge can help answer over one source using another one, are yet to be answered. In this paper, therefore, we study the knowledge transfer of multi-hop reasoning between structured and unstructured sources. We first propose a unified QA framework named SIMULTQA to enable knowledge transfer and bridge the distinct supervisions from KB and text sources. Then, we conduct extensive analyses to explore how knowledge is transferred by leveraging the pre-training and fine-tuning paradigm. We focus on the low-resource fine-tuning to show that pre-training SIMULTQA on one source can substantially improve its performance on the other source. More fine-grained analyses on transfer behaviors reveal the types of transferred knowledge and transfer patterns. We conclude with insights into how to construct better QA datasets and systems to exploit knowledge transfer for future work.[1]

1 Introduction

Structured knowledge source, such as Knowledge Base (KB) and unstructured knowledge source, such as text corpus, are arguably the most popular sources for complex question answering (CQA). Multi-hop KB based question answering (KBQA) systems translate questions to logical forms to be executed over a KB for finding answers (Talmor and Berant, 2018; Maheshwari et al., 2019; Lan and Jiang, 2020; Gu et al., 2020; Das et al., 2021; Ye et al., 2021), while text based QA (TextQA) systems leverage large text corpora to retrieve paragraphs and extract answer spans for complex questions (Yang et al., 2018; Qi et al., 2019; Asai et al., 2020; Dhingra et al., 2020; Zhu et al., 2021).

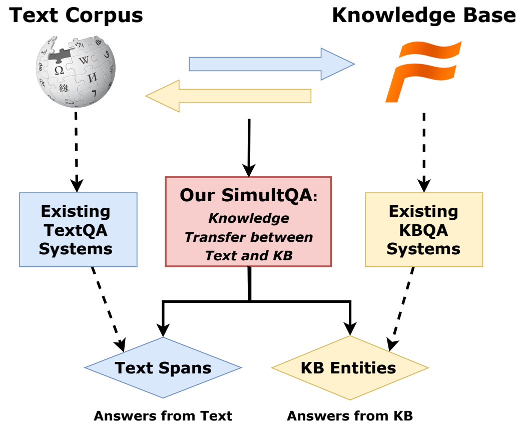

Figure 1: To facilitate knowledge transfer between structured and unstructured sources, we develop a unified framework SIMULTQA that can leverage supervisions from both sources to answer complex questions.

However, despite the impressive performance of separate KBQA and TextQA systems, it is not quite clear to the community whether a system trained on one source can be transferred and beneficial to question answering over another source. Inspired by the general transfer learning in NLP by pre-trained language models (PLMs) (Radford et al., 2018; Devlin et al., 2019; Raffel et al., 2020), it is important to study this research problem systematically and thoroughly for the following reasons. First, given the heterogeneity of structured and unstructured sources, it is desirable to build a unified reasoning module to work on both text and KB and combine different source-specific supervisions. Second, transfer learning has shown to boost the performance on low-resource domains, and it would be practically useful to leverage annotated datasets on one source for CQA on the other source, especially when human annotations are expensive to create new multi-hop QA datasets. Third, it is

[*] Equal contribution
[1] Code and data are available at https://github.com/Stefan1220/SimultQA

also critical to investigate what kind of knowledge can be transferred, which can inspire future CQA dataset creation and system design.

One major obstacle in such an investigation for knowledge transfer between structured and unstructured sources is the disparity of them and their specifically designed QA systems as we mentioned earlier. For instance, KB is highly structured and curated where complex query functions can be executed, while text data is unstructured and noisier, leading to quite distinct QA systems. One relevant line of research is HybridQA that tries to leverage multiple sources for QA (Mihaylov and Frank, 2018; Sun et al., 2018, 2019; Min et al., 2019; Oguz et al., 2020; Shi et al., 2021). To operate their single model on both KB and text, these methods primarily convert distinct sources into similar data format, e.g., merge text and KB by entity linking, which sacrifices unique characteristics of each source to some extent and makes it harder to investigate knowledge transfer as sources are entangled together. Thus, typical HybridQA methods are not suitable for studying knowledge transfer problem.

In this paper, *our first contribution* is proposing a unified CQA framework to enable knowledge transfer between structured and unstructured sources. The proposed framework, SIMULTQA, could perform multi-hop reasoning over text and KB simultaneously by collecting reasoning paths from either text or KB, then rerank paths to select the best one for generating the answer. There are several new and desirable properties of SIMULTQA. First, SIMULTQA unifies the recent advanced KBQA (Luo et al., 2018; Lan and Jiang, 2020) and TextQA (Chen et al., 2017; Asai et al., 2020) systems seamlessly, which preserves their unique strengths maximally to handle various reasoning types. Second, SIMULTQA can utilize distinct supervisions from both sources, which has the potential to combine both KBQA and TextQA datasets for a unified training. Last but not least, since SIMULTQA can be applied to any source, we can pre-train it on KB and fine-tune it on text and vice versa, which makes it easier to quantify transfer effect brought by the pre-training on a different source. In summary, despite the framework design looks straightforward, we are the first to unify two seemingly distinct CQA systems and study knowledge transfer between two sources for CQA.

With SIMULTQA that enables knowledge transfer, *our second contribution* is to systematically analyze the transfer behavior to help us deeply understand the nature of the multi-hop reasoning process in KB and Text. We apply our methodology to CWQ (Talmor and Berant, 2018) and HotpotQA (Yang et al., 2018), which are arguably the most popular dataset in KB and text source, and are representative enough to cover most of the reasoning types on KB and text. We first show that pre-training on one source can consistently improve the fine-tuning performance on the other one in the low-resource setting, indicating future data-hungry QA systems can be boosted by pre-training on another disparate source, especially when human annotation is expensive. More interestingly, further fine-grained analyses attempt to reveal sources of performance gain and find out what knowledge is transferred. We mainly investigate three aspects, reasoning types, reasoning hops and question similarity. We find that despite KB and text sources are quite disparate, SIMULTQA still find ways to transfer knowledge by learning a shared semantic space for the reasoning and a high-level understanding beyond distinct surface forms of reasoning paths. In addition, we study a more challenging transfer setting where we seek to use text reasoning to answer KB-based questions[2] and vice versa. Promising results are obtained by using text knowledge to help KB questions highlighting the expressiveness of text corpus. We conclude that knowledge transfer between structured and unstructured sources is an intriguing direction to combine the strengths of KBQA and TextQA systems and to use data from one source to boost QA on the other. To the best of our knowledge, this paper is the first to study knowledge transfer between KB- and text-based CQA in a quantitative and systematic manner.

2 Related Work

Complex Question Answering. There has been a long history of QA models to answer simple questions (Berant et al., 2013; Rajpurkar et al., 2016; Chen et al., 2017; Wang et al., 2018; Lee et al., 2018; Yang et al., 2019; Karpukhin et al., 2020). More recent attention has focused on answering complex questions, which requires a multi-hop reasoning process (Yang et al., 2018; Fang et al., 2020). For example, some of them target questions that can be answered using multiple text paragraphs as evidences (Das et al., 2018; Qi et al., 2019; Feldman and El-Yaniv, 2019; Asai et al., 2020), while

[2] We refer to questions originally from KBQA/TextQA datasets as KB-based/text-based questions in this paper.

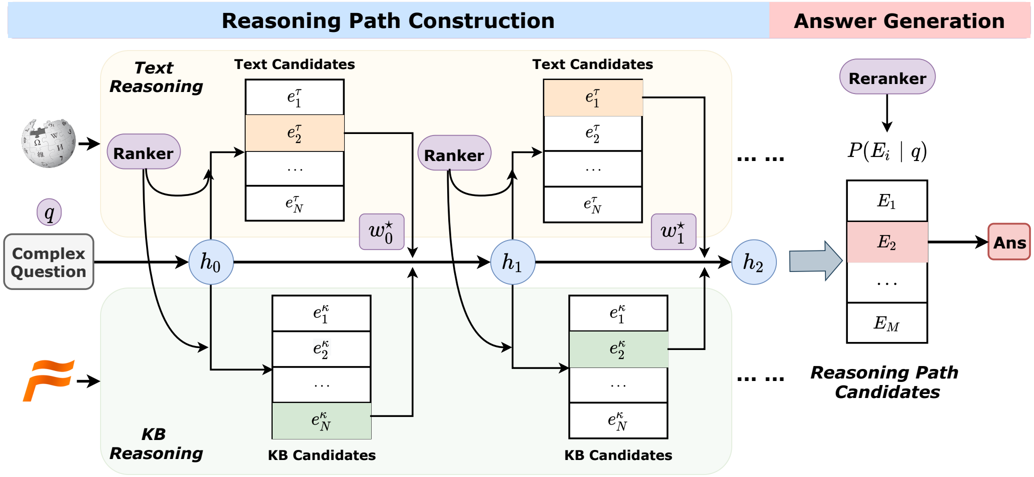

Figure 2: Overview of SIMULTQA Framework. There are two stages including constructing reasoning path from either text or KB, and path reranking for the answer generation. In the inference time, the reasoning can be performed simultaneously over text and KB source to find the final answer.

some existing KBQA works (Bao et al., 2016; Luo et al., 2018; Chen et al., 2019; Lan et al., 2019; Lan and Jiang, 2020) studied how to answer questions by iteratively chaining multiple knowledge base relations into the evidence path. Our proposed framework unifies these two recent trends of CQA frameworks in text and KB to study knowledge transfer between them.

Hybrid Question Answering. HybridQA is a line of QA research that also studies different knowledge sources (e.g., text articles, Web tables, knowledge bases) for answering questions (Mihaylov and Frank, 2018; Sun et al., 2018, 2019; Xiong et al., 2019; Min et al., 2019; Oguz et al., 2020; Chen et al., 2020a,b). This line of work typically requires extra human efforts to merge hybrid data for later complex modeling, for example, linking text paragraphs to KB by entity linking or universal schema (Das et al., 2017; Sun et al., 2018, 2019) or converting KB edges to plain text (Oguz et al., 2020), which is not needed in SIMULTQA. Their major motivation is to unify data formats for text and KB and construct a more comprehensive knowledge space, which is orthogonal to our motivation of studying knowledge transfer between intact knowledge space of text and KB.

Transfer Learning in NLP. In the last few years, NLP has witnessed the emergence of several transfer learning techniques, and their effectiveness of constantly improving state-of-the-art on a wide range of NLP tasks. Traditional transfer learning techniques (Pan and Yang, 2009) include multi-task learning, domain adaptation, etc (Liu et al., 2019; Clark et al., 2019; Ruder et al., 2019). More recently, fine-tuning PLMs has become the de facto standard for transferring knowledge among NLP tasks (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019; Raffel et al., 2020). In this paper, we study knowledge transfer between structured and unstructured sources in CQA task and use BERT models as the backbone of our approach.

3 SIMULTQA Framework

SIMULTQA is a unified framework for multi-hop reasoning to incorporate both KB and text sources. It consists of two stages, iteratively reasoning and final reranking, which can be trained with supervisions from both sources.

3.1 Reasoning Path Construction

CQA requires a multi-hop reasoning process to derive the answer. For KBQA, the reasoning is to traverse the knowledge graph for multi-steps based on generated queries from the question, while for TextQA, it is to collect multiple documents from a text corpus. We consolidate both by iteratively searching for evidence from each source and construct the reasoning path at the end. The key formulation is we treat each step as a ranking problem and train the model to select the most appropriate document/ KB query graph from text corpus/ knowledge graph that can answer the complex question.

Formally, at time step $t, (t \geq 1)$, we are given the complex question q, a pool of candidate evidences, $e_i \in \{e_1, ..., e_N\}$, and the hidden state h_{t-1} from previous step. We first encode them by the BERT [CLS] token representation to get the contextual embedding $\mathbf{w}_i$ for each candidate e_i. Then, we calculate the probability of e_i to be selected in current step by feeding $\mathbf{w}_i$ to a fully-connected layer. We denote text evidence as $e_i^{\mathcal{T}}$ which is a sequence of tokens from a document in the text corpus. For KB evidence, following previous work (Lan and Jiang, 2020), each candidate is "serialized" into a sequence of relation tokens and denoted as e_i^{κ}. The scoring process at t-th step is defined as follows:

$$\mathbf{w}_i^{\mathcal{T}} = \text{BERT}_{[\text{CLS}]}([q; e_i^{\mathcal{T}}]), \tag{1}$$

$$\mathbf{w}_i^{\kappa} = \text{BERT}_{[\text{CLS}]}([q; e_i^{\kappa}]), \tag{2}$$

$$P_t^{\mathcal{T}}(e_i^{\mathcal{T}}|q) = \text{FC}(\mathbf{w}_i^{\mathcal{T}}, \mathbf{h}_t) \in [0, 1], \tag{3}$$

$$P_t^{\kappa}(e_i^{\kappa}|q) = \text{FC}(\mathbf{w}_i^{\kappa}, \mathbf{h}_t) \in [0, 1], \tag{4}$$

where $[q; e_i]$ represents the concatenation of the question and evidence separated by [SEP] token. We simply choose a Recurrent Neural Network (RNN), and h_t is calculated to model the sequential multi-hop reasoning process as follows:

$$\mathbf{h}_t = \text{RNN}(\mathbf{h}_{t-1}, \mathbf{w}_{t-1}^*) \in \mathbb{R}^d \tag{5}$$

where $\mathbf{w}_{t-1}^*$ encodes the ground-truth evidence in previous step for $t > 1$ during training and $\mathbf{h}_0$ will be a free parameterized vector to be initialized randomly, when $t = 1$. During inference, evidences will be dynamically inferred based on the results from previous step. To encourage knowledge transfer, we share the parameters for the recurrent module and BERT model (as well as the answer generation module that will be introduced later) for KB and text source, which will be jointly optimized. We next introduce how to generate high-quality candidate pools for each step.

Generate Text Candidates. Following previous methods (Chen et al., 2017), for a given complex question and a large text corpus (e.g., Wikipedia), we leverage TF-IDF based methods to retrieve top-K documents with the tri-gram hashing techniques. For the iterative process, we reuse TF-IDF method to retrieve candidates in next step combining the complex question and the previous retrieved document. Moreover, since TF-IDF methods mainly consider the lexical matching, there are several advanced approaches that can be explored to extend the reasoning path, such as meta-info based (e.g., entity links, hyperlinks (Nie et al., 2019; Asai et al., 2020)), search engine (Qi et al., 2019, 2020), dense retrieval (Xiong et al., 2021). We consider hyperlinks (Asai et al., 2020) in this work and leave more sophisticated methods to future work.

Generate KB Candidates. We follow recent advanced staged query generation methods (Yih et al., 2015; Luo et al., 2018; Lan and Jiang, 2020) to generate candidates and perform KB reasoning. As shown in Figure 2, the KB module starts from a grounded entity in the complex question and identifies core relation paths[3] as candidates with necessary constraints. We iteratively generate and rank candidate query graphs in each step based on the topic entity or the entity from the last step.

With the iterative ranking in each step, we can establish the reasoning chain as a sequence of documents, $E^{\mathcal{T}} = [e_1^{\mathcal{T}}, ..., e_k^{\mathcal{T}}]$ for TextQA and a sequence of query graphs, $E^{\kappa} = [e_1^{\kappa}, ..., e_k^{\kappa}]$ for KBQA. We score each path by the multiplication of probability of each selected evidence as $P(e_1|q) \cdot ... \cdot P(e_k|q)$ and use beam search to produce top-M reasoning paths $\{E_1, ..., E_M\}$ for the final answer generation.

3.2 Reranking and Answer Generation

Given a complex question q and several reasoning paths $\{E_1, ...E_M\}$ from the previous component, we rerank the paths based on how likely they can answer the question. We use another BERT [CLS] token representation to encode the reasoning path E_i with a fully connected model to output the probability of choosing E_i as follows:

$$\mathbf{v}_i = \text{BERT}_{[\text{CLS}]}([q, \{e_{i1}, ..., e_{ik}\}]), \tag{6}$$

$$P(E_i|q) = \text{FC}(\mathbf{v}_i) \in [0, 1] \tag{7}$$

After the reasoning path reranking, our system allows the KB reasoning path and text reasoning path to be handled differently. This reflects the advantage of our system to combine the strength of both KBQA and textQA as discussed earlier. Since KB is structured, we can directly execute the complete query graph in the knowledge graph to get the answers. For question answering with textual evidence chains in particular, another *reader* component is employed to select the text spans that are the final answer based on the top-ranked path.

3.3 Training and Inference

We leverage the annotated document labels from HotpotQA dataset to train the reasoning path construction and reranking modules. For CWQ dataset, we split the golden complex logic form into sub-queries by defining the sub-query to be composed of head/tail entities along with one relation or two relations with CVT type node. Constraint relations are also added to the connected sub-queries. The

[3] As in (Lan and Jiang, 2020), we allow the relation to be a single predicate or two predicates connected through a CVT node designed for a multi-argument relation.

sub-queries are treated as supervisions in each reasoning step as well as the path reranking module. Note that it is now the standard way to train robust CQA systems by leveraging full supervision in each hop. We leave utilizing distantly weak supervisions for training to future work. In each step of reasoning module, the loss functions for KB and text are defined as follows:

$$\begin{aligned} L_t^{\mathcal{T}} &= -\log P(e_t^{\mathcal{T}}|q) \\ &- \textstyle\sum_{\widetilde{e^{\mathcal{T}}}\in C_t^{\mathcal{T}}} \log(1-P(\widetilde{e^{\mathcal{T}}}|q)) \end{aligned} \tag{8}$$

$$\begin{aligned} L_t^{\mathcal{K}} &= -\log P(e_t^{\mathcal{K}}|q) \\ &- \textstyle\sum_{\widetilde{e^{\mathcal{K}}}\in C_t^{\mathcal{K}}} \log(1-P(\widetilde{e^{\mathcal{K}}}|q)) \end{aligned} \tag{9}$$

where $C_t^{\mathcal{T}}$ and $C_t^{\mathcal{K}}$ are negative samples. For text, we follow previous work (Asai et al., 2020) to generate lexically and semantically similar negative samples based on TF-IDF as well as hyperlinks. For KB, we treat all query graphs other than the golden one in the same step as negative samples.

In terms of reranking reasoning paths for KB and text, we reuse the previous supervisions to train a ranker model (Eqn. 7) for selecting the correct path with the loss function as follows:

$$L_{\text{rank}}^{\mathcal{T}} = -\textstyle\sum_i y_i^{\mathcal{T}} \cdot \log(P(E_i^{\mathcal{T}}|q)) \tag{10}$$

$$L_{\text{rank}}^{\mathcal{K}} = -\textstyle\sum_i y_i^{\mathcal{K}} \cdot \log(P(E_i^{\mathcal{K}}|q)) \tag{11}$$

where $y_i^{\mathcal{T}}$ and $y_i^{\mathcal{K}}$ are the assigned labels for the golden path of i-th sample from two sources. We also design negative samples for reasoning paths by replacing the golden evidence in one of k hops.

4 Knowledge Transfer Experiments

We focus on investigating knowledge transfer between structured and unstructured sources in this paper, though the proposed SIMULTQA can be applied to any open-domain CQA datasets. We seek to answer three research questions (RQs):

- **RQ1**: Can the knowledge learned on one source help the QA performance on another one? (§4.2)
- **RQ2**: What kind of knowledge has been transferred between KB and text? (§4.3)
- **RQ3**: Can knowledge transfer help answer questions by both sources? (§4.4)

4.1 Experimental Setup

Choice of Datasets. Investigating knowledge transfer between text and KB requires at least one dataset from each source. Without losing the generality, we choose Wikipedia and Freebase as the source for text and KB respectively, and select their arguably the most representative CQA dataset to cover the majority of reasoning types. We leave applying SIMULTQA to other domain-specific sources and datasets as future work.

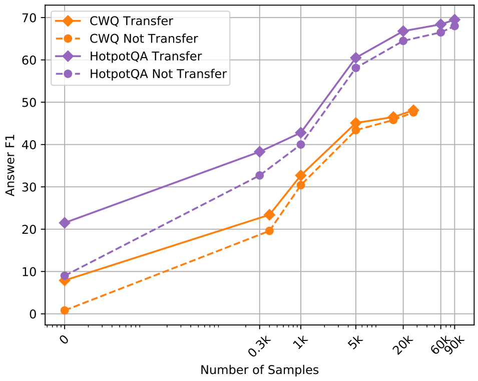

Figure 3: Pre-training and fine-tuning experiments on CWQ and HotpotQA datasets. We first pre-train SIMULTQA on one source with the full dataset, then fine-tune it on another one with various sizes of samples.

The selected large-scale KB dataset is Complex WebQuestions (CWQ) (Talmor and Berant, 2018) that consists of around 27K/3.5K/3.5K samples for train/dev/test. The text dataset is HotpotQA (Yang et al., 2018) that consists of around 90K/7.4K/7.4K samples for train/dev/test. For both datasets, we focus on the most practical setting, which is the open-domain QA, meaning that the model needs to reason over the entire knowledge space for answering the question.

Implementation Details. We adopt pre-trained BERT models (Devlin et al., 2019) using the uncased base configuration (768-hidden) for our reasoning path construction and reranking module. We follow Graph Retriever (Asai et al., 2020) and use their pre-trained whole word masking uncased large configuration (1024-hidden) for the reader. During the process of reasoning path construction, we set the number of negative examples along with the gold example as 30, set the number of hops as 2, and use beam search when doing the inference. Beam size is set as 5 for CWQ and 9 for HotpotQA.

4.2 RQ1: Quantitative measurement

Pre-training and Fine-tuning. A straightforward way to investigate the effect of knowledge transfer between text and KB is to leverage the pre-training and fine-tuning paradigm, where we first pre-train SIMULTQA on one source and fine-tune it on another one. The transfer effect then can be measured by the performance difference with and without the pre-training stage. Furthermore, to demonstrate the transfer effect carefully, we focus on the low-

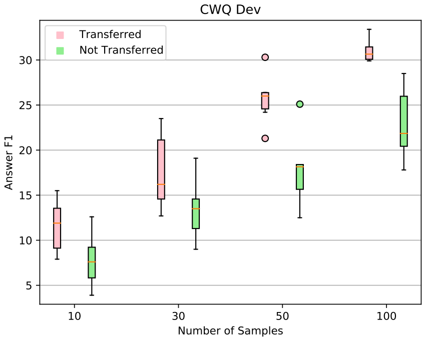

Figure 4: Few-shot experiments on CWQ dataset. Boxes extends from the first quartile to the third quartile of the samples, and lines inside boxes mark the medians.

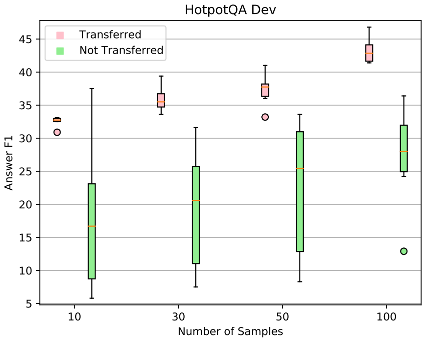

Figure 5: Few-shot experiments on HotpotQA dataset.

resource setting where we increasingly add more samples for the fine-tuning. Note that we only pre-train and fine-tune the first stage of SIMULTQA, which is the retriever, because this is the most important module for multi-hop reasoning.

Transfer Text Knowledge to KB. We show the fine-tuning performance in Figure 3, where we can see that pre-training SIMULTQA on text dataset can consistently improve the performance on KB dataset, especially when the fine-tuning data is limited. Specifically, when there is no fine-tuning data for KB (zero-shot transfer), text pre-training achieves about 8 F1 score on CWQ already, meaning that text knowledge can greatly help the QA model on KB. We also notice that when a large number of KB samples are available, the transfer effect becomes less prominent, possibly due to the model begins overfitting KB-specific features.

To further demonstrate the transfer effect on low-resource setting, we conduct few-shot experiments by randomly sampling only a handful of samples for fine-tuning. We sample five times to reduce the randomness of few-shot samples and results are shown in Figure 4. We can see the transfer effect from text to KB more clearly, and this finding can be leveraged to boost the performance of KBQA in low-data region when human annotations are expensive to collect over domain-specific KBs.

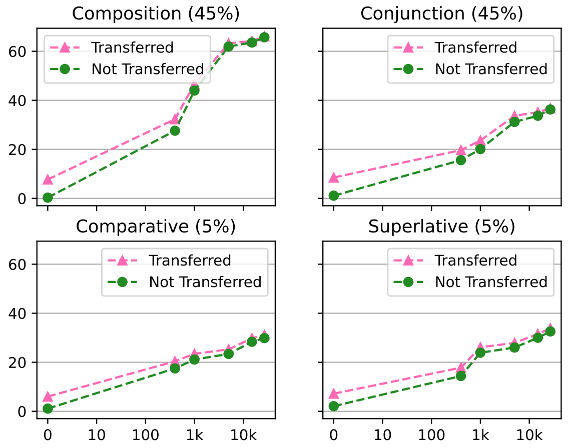

Figure 6: Analysis of reasoning types in CWQ. Numbers in parentheses are percentages of types.

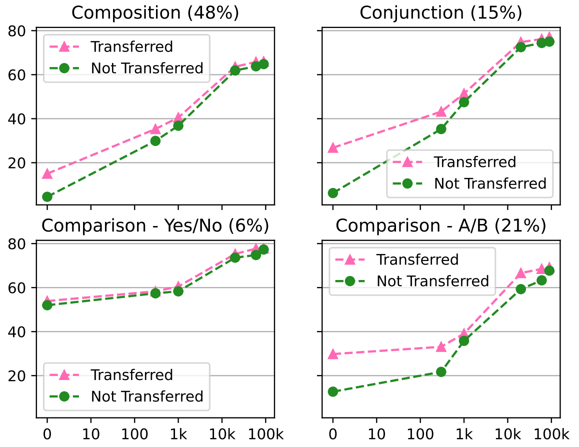

Figure 7: Analysis of reasoning types in HotpotQA. Numbers in parentheses are percentages of types.

Transfer KB Knowledge to Text. Figure 3 shows the pre-training on KB also provides performance boost for fine-tuning on text domain in the low-resource setting. In zero-shot transfer, pre-training on KB brings about 12.5 F1 improvement, which verifies that KB knowledge can also help answer text-based questions. Moreover, few-shot experiments in Figure 5 demonstrate the transfer effect when < 100 text-based samples are available. We notice that the variance of few-shot experiments is greatly reduced by the pre-training, indicating another potential useful transfer effect may be to help reduce the instability in the few-shot learning. Meanwhile, we conduct error analysis for both CWQ and HotpotQA respectively in Table 2.

4.3 RQ2: What has been transferred?

We further conduct fine-grained analyses under previous experiment settings trying to answer what knowledge is transferred between structured and

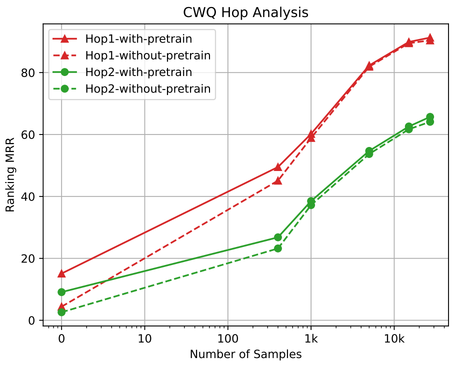

Figure 8: Hop Analysis on the CWQ dataset.

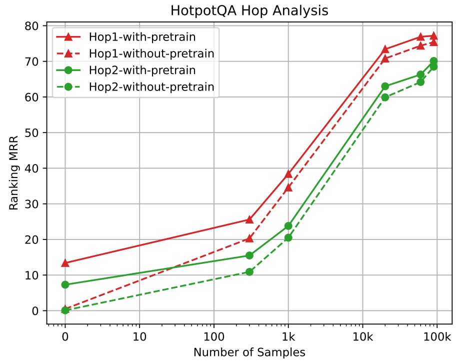

Figure 9: Hop Analysis on the HotpotQA dataset.

unstructured sources. We hypothesize three major factors that may influence the transfer effect and test their correlations with performance changes.

Reasoning types play a central role in answering complex questions. SIMULTQA is expected to learn similar reasoning processes from structured/unstructured sources if the knowledge about certain reasoning types is transferred. We analyze the transfer effect w.r.t. various reasoning types defined in both datasets (we refer to the original papers (Talmor and Berant, 2018; Yang et al., 2018) for their detailed definitions). As shown in Figure 6 and 7, the most shared two types in both text and KB, composition (i.e., infer the bridge entity) and conjunction (i.e., checking multiple properties) questions are benefited from knowledge transfer the most (especially in the zero-shot transfer), which suggests that SIMULTQA is able to transfer similar reasoning processes between disparate sources regardless of their distinct surface forms.

Another interesting observation is for the Comparison - A/B on HotpotQA (e.g., *Who is older, A or B?*) that has a larger F1 score gain under the zero-shot setting. This type asks a two-choice question which can be answered by locating an entity as the final answer through iteratively retrieving two evidences, which is similar to the chain reasoning in Composition and Conjunction. Although this specific reasoning type is not shared by both sources, the similarity between the reasoning processes makes it benefited from knowledge transfer.

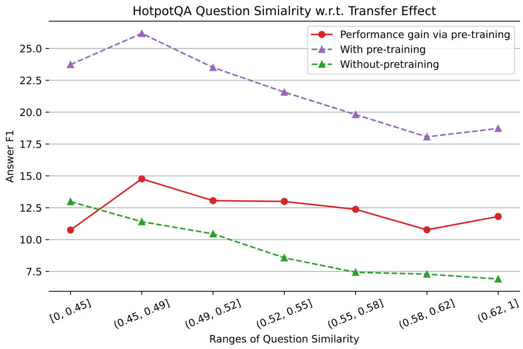

Figure 10: Relationship between question similarity and performance gain.

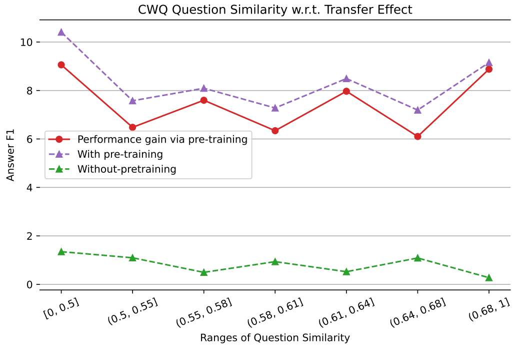

Figure 11: Relationship between question similarity and performance gain on CWQ.

Reasoning hops correspond to decomposed subquestions from a complex question and we are interested in whether the transfer effect varies according to different hops. In both KBQA and TextQA, the first hop sub-question tends to closely connect with a topic entity or phrase mentioned in the question, while the subsequent (second) hops require more semantic inference to answer the sub-question. As shown in Figure 8, the first hop in CWQ dataset usually gets higher retrieval performance and can be transferred from the other source, which indicates that the knowledge of finding the topic entity in the question is transferred. We also show the hop analysis for HotpotQA in Figure 9. Similar to the observation on CWQ, it shows that the first hop in HotpotQA gets higher retrieval performance and can be transferred from the other source, which further validate that the knowledge of finding the topic entity mentioned in the question is transferred.

Question similarity measures the semantic similarities between questions in testing and training. We hypothesize that the transfer might be easier for testing questions if some similar ones appear in the training. We investigate the zero-shot trans-

Complex question: What is European Union country used the Hungarian forint as its main currency?
Gold KB reasoning path: European Union $\xrightarrow{members}$ y1(CVT) $\xrightarrow{member}$ Hungary $\xleftarrow{currency\_used}$ Forint
Reasoning paths from text source: 1. *(first passage)* The currency of Hungary is the Hungarian forint since 1 August 1946 ... *(second passage)* As a member of the European Union, Hungarian government ... replace the forint with the euro. 2. *(first passage)* The forint is the currency of Hungary. ... and the forint has been declared fully convertible. *(second passage)* As a member of the European Union, the long-term of aim of the Hungarian government ... 3. *(first passage)* The Gulden or forint was the currency ... and the Austro-Hungarian Monarchy ... *(second passage)* In Hungary, the forint was divided into ... for the unit and subunit.

Table 1: Case Study. The question comes from CWQ dataset and is originally answered by a KB reasoning path.

fer to study the influence of pre-training questions more directly. Specifically, for a CWQ question in testing set, we calculate its semantic similarities with all HotpotQA questions in pre-training and take the average of top 5 similarities. We then split CWQ testing questions into several chunks based on this averaged similarity and aggregate their QA performance before and after the pre-training. We do the same thing for the other direction of transfer. We present the relationship between question similarity and performance in HotpotQA on Figure 10. Interestingly, we observe that question similarity is not correlated with transfer effect, i.e., higher similar testing questions are not necessarily to obtain larger performance gain. This finding implies that SIMULTQA transfers the reasoning process in a high-level semantic space rather than through low-level lexical features. We show questions similarity for CWQ in Figure 11, where we also find question similarity is not correlated with the transfer effect.

	Type	%
CWQ	Questions with constraints	50
	Questions with aggregation functions	25
	Others	25
HotpotQA	Relations not covered in KB	45
	Not satisfy chain reasoning	35
	Others	20

Table 2: We manually analyze 20 questions with wrong predicted answers respectively from CWQ and HotpotQA and categorize them.

Error analysis is conducted under the full dataset fine-tuning setting to further understand the transfer behaviors by manually checking errors and categorizing them. As is shown in Table 2, 75% of wrongly answered questions sampled from CWQ contain additional constraints or arithmetic operations which are hard to be supported by text corpus. 45% questions sampled from HotpotQA contain semantic knowledge or relations which cannot be covered in Knowledge Base. 35% of them don't follow the chain reasoning process and are not suitable to be decomposed to answer step by step like KBQA. The other remaining questions are related to errors in retrieval, re-ranking or span extraction process. These unshared knowledge between CWQ and HotpotQA make it reasonable that those wrongly answered questions in one data source cannot be contributed from the other data source.

4.4 RQ3: Answering complex questions by both knowledge sources

To directly measure the transfer effect, in previous sections, the reasoning is always performed on the same knowledge source as where the question is from, e.g., a text-based question is answered by the text reasoning path. Now, we ask whether questions can be better answered by considering both sources. Note that this is a more challenging setting because questions in both datasets only have supervisions from one source, which thus requires stronger transfer signal. Moreover, we can utilize this setting to test how complementary two knowledge sources are, regarding how much they can help each other. Specifically, in addition to the annotated reasoning paths, we collect candidate paths from the other source, i.e., KB paths for text-based questions and text paths for KB-based questions. The final reranking will select the best path from both KB and text paths for all questions. We refer to this setting as the hybrid evaluation.

Our preliminary experiments show that pre-training on one source and then fine-tuning on the other tends to forget the knowledge of the first source, leading to less satisfactory results. Therefore, we jointly train SIMULTQA by iteratively sampling batches from both sources to expose the model to both sources equally in the training time. We then compare the hybrid evaluation with the single-source evaluation in Table 4. For CWQ

(HotpotQA) In the television series Green Hornet, which actor played the role of Kato?
Gold reasoning path from text source: *(first passage)* The Green Hornet is a television series on ABC ... starring Van Williams and Bruce Lee ... *(second passage)* Kato is a fictional character ... was portrayed by Bruce Lee.
Reasoning paths from KB source: 1. Green Hornet $\xleftarrow{series}$ y1(CVT) $\xleftarrow{starring\_roles}$ Bruce Lee $\xleftarrow{actor}$ y2(CVT) $\xleftarrow{appear\_in\_tv\_program}$ Kato 2. Green Hornet $\xleftarrow{film}$ y1(CVT) $\xleftarrow{character}$ AI Hodge $\xleftarrow{notable\_types}$ TV Actor 3. Green Hornet $\xleftarrow{film}$ y1(CVT) $\xleftarrow{character}$ Seth Rogen $\xleftarrow{appeared\_on}$ y2(CVT) $\xleftarrow{appearance\_type}$ Host

Table 3: Case study. The question comes from HotpotQA and is originally answered by a textual reasoning path.

CWQ	**F1**	**Hit@1**
SimultQA- KB	46.7	47.7
SimultQA- Hybrid	48.5	49.8
HotpotQA	**F1**	**EM**
SimultQA- Text	71.7	58.8
SimultQA- Hybrid	71.2	58.4

Table 4: Comparing single and hybrid evaluations.

dataset, SimultQA- Hybrid achieves 1.8 F1 score gains after incorporating text paths for the inference, while the performance of HotpotQA is not influenced in hybrid evaluation after incorporating KB paths. This shows that text knowledge is easier to be transferred to help KB-based questions.

We also conduct case studies by retrieving top-ranked reasoning paths in hybrid evaluation. Table 1 presents a CWQ question and shows that top-ranked text paths are closely related to the golden KB path, indicating that linguistics variants of text knowledge can greatly help KB reasoning. On the other hand, KB knowledge seems to be less helpful to answer text-based questions based on the overall QA performance in Table 4, partially due to the incompatibility between TextQA and KBQA dataset, e.g., entities and relations that cannot be mapped to KB, reasoning types that cannot be answered by KB (see Section 4.3), etc. However, we still find cases in HotpotQA in Table 3 to show KB can somehow contribute to textual reasoning as well.

5 Discussion for future directions

Based on our findings of knowledge transfer for CQA in this paper, we discuss the following directions for future CQA datasets and systems.

Knowledge transfer for efficient CQA dataset annotations. When annotating new CQA datasets whether on text or KB, it would be beneficial to leverage pre-trained SimultQA on other sources to discover high-quality reasoning paths for further annotating, which will save much annotation cost.

Diversity of reasoning types. Both text and KB sources are dominant by relatively easy reasoning types, e.g., composition and conjunction. Future CQA datasets should pursue more diverse and harder reasoning types, e.g., types with constraints and arithmetic operations (Dua et al., 2019).

A universal reasoning module. Investigating knowledge transfer between text and KB in this paper suggests that despite the discrepancy of surface forms in different sources, their underlying reasoning processes could be shared. This points out the possibility of learning a universal reasoning process from multiple sources and it is strongly desired to modularize such a reasoning process, which can be injected it into future QA systems.

6 Conclusion

In this paper, we study CQA over structured and unstructured knowledge sources (i.e., KB and text particularly), and focus on studying the knowledge transfer between different knowledge sources. To facilitate the transfer, we first propose a unified CQA framework, SimultQA to bridge KBQA and TextQA systems. Empirical results show that knowledge transfer enables substantial improvements on low-resource domains. More importantly, we conduct fine-grained analyses to shed more light on how knowledge is transferred to inspire future research on knowledge transfer between sources, and we conclude the paper with insights for future CQA datasets and systems.

Acknowledgments

The authors would like to thank the reviewers and OSU NLP group members for their thoughtful comments. This research was sponsored in part by Google Faculty Award, NSF IIS-1815674, NSF CAREER #1942980, NSF OAC-2112606.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *International Conference on Learning Representations*.

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*, pages 2503–2514.

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*, pages 1533–1544.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Wang, and William W Cohen. 2020a. Open question answering over tables and text. *arXiv preprint arXiv:2010.10439*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020b. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1026–1036.

Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jijnasa Nayak, and Lun-Wei Ku. 2019. Uhop: An unrestricted-hop relation extraction framework for knowledge-based question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 345–356.

Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D Manning, and Quoc Le. 2019. Bam! born-again multi-task networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2018. Multi-step retriever-reader interaction for scalable open-domain question answering. In *International Conference on Learning Representations*.

Rajarshi Das, Manzil Zaheer, Siva Reddy, and Andrew McCallum. 2017. Question answering on knowledge bases and text using universal schema and memory networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 358–365.

Rajarshi Das, Manzil Zaheer, Dung Thai, Ameya Godbole, Ethan Perez, Jay Yoon Lee, Lizhen Tan, Lazaros Polymenakos, and Andrew McCallum. 2021. Case-based reasoning for natural language queries over knowledge bases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9594–9611.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Bhuwan Dhingra, Manzil Zaheer, Vidhisha Balachandran, Graham Neubig, Ruslan Salakhutdinov, and William W. Cohen. 2020. Differentiable reasoning over a virtual knowledge base. In *International Conference on Learning Representations*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of NAACL-HLT*, pages 2368–2378.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8823–8838.

Yair Feldman and Ran El-Yaniv. 2019. Multi-hop paragraph retrieval for open-domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2296–2309.

Yu Gu, Sue Kase, Michelle Vanni, Brian Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2020. Beyond iid: Three levels of generalization for question answering on knowledge bases. *arXiv e-prints*, pages arXiv–2011.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Yunshi Lan and Jing Jiang. 2020. Query graph generation for answering multi-hop complex questions from knowledge bases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 969–974.

Yunshi Lan, Shuohang Wang, and Jing Jiang. 2019. Multi-hop knowledge base question answering with an iterative sequence matching model. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 359–368. IEEE.

Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. 2018. Ranking paragraphs for improving answer recall in open-domain question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 565–569.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2185–2194.

Gaurav Maheshwari, Priyansh Trivedi, Denis Lukovnikov, Nilesh Chakraborty, Asja Fischer, and Jens Lehmann. 2019. Learning to rank query graphs for complex question answering over knowledge graphs. In *International semantic web conference*, pages 487–504. Springer.

Todor Mihaylov and Anette Frank. 2018. Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 821–832.

Sewon Min, Danqi Chen, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Knowledge guided text retrieval and reading for open domain question answering. *arXiv preprint arXiv:1911.03868*.

Yixin Nie, Songhe Wang, and Mohit Bansal. 2019. Revealing the importance of semantic retrieval for machine reading at scale. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2553–2566.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2020. Unified open-domain question answering with structured and unstructured knowledge. *arXiv preprint arXiv:2012.14610*.

Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Peng Qi, Haejun Lee, Oghenetegiri Sido, Christopher D Manning, et al. 2020. Retrieve, rerank, read, then iterate: Answering open-domain questions of arbitrary complexity from text. *arXiv preprint arXiv:2010.12527*.

Peng Qi, Xiaowen Lin, Leo Mehr, Zijian Wang, and Christopher D Manning. 2019. Answering complex open-domain questions through iterative query generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2590–2602.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training (2018).

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 unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Tutorials*, pages 15–18.

Jiaxin Shi, Shulin Cao, Lei Hou, Juanzi Li, and Hanwang Zhang. 2021. Transfernet: An effective and transparent framework for multi-hop question answering over relation graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4149–4158.

Haitian Sun, Tania Bedrax-Weiss, and William Cohen. 2019. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2380–2390.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4231–4242.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 641–651.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2018. R3: Reinforced ranker-reader for open-domain question answering.

Wenhan Xiong, Xiang Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oguz. 2021. Answering complex open-domain questions with multi-hop dense retrieval. In *International Conference on Learning Representations*.

Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2019. Improving question answering over incomplete kbs with knowledge-aware reader. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4258–4264.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with bertserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

Xi Ye, Semih Yavuz, Kazuma Hashimoto, Yingbo Zhou, and Caiming Xiong. 2021. Rng-kbqa: Generation augmented iterative ranking for knowledge base question answering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1321–1331.

Yunchang Zhu, Liang Pang, Yanyan Lan, Huawei Shen, and Xueqi Cheng. 2021. Adaptive information seeking for open-domain question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3615–3626.