UniK-QA: Unified Representations of Structured and Unstructured Knowledge for Open-Domain Question Answering

Anonymous ACL submission

Abstract

We study open-domain question answering with *structured, unstructured* and *semi-structured* knowledge sources, including text, tables, lists and knowledge bases. Departing from prior work, we propose a unifying approach that homogenizes all sources by reducing them to text and applies the retriever-reader model which has so far been limited to text sources only. Our approach greatly improves the results on knowledge-base QA tasks by 11 points, compared to latest graph-based methods. More importantly, we demonstrate that our *unified knowledge* (`UniK-QA`) model is a simple and yet effective way to combine heterogeneous sources of knowledge, advancing the state-of-the-art results on two popular question answering benchmarks, NaturalQuestions and WebQuestions, by 3.5 and 2.6 points, respectively.

1 Introduction

Answering factual questions has long been an inspirational challenge to information retrieval and artificial intelligence researchers (Voorhees, 1999; Lopez et al., 2011). In its most general form, users can ask about *any* topic and the answer may be found in *any* information source. Defined as such, the challenge of *open domain question answering* is extremely broad and complex. Though there have been successful undertakings which embrace this complexity (notably Ferrucci, 2012), most recent works make simplifying assumptions as to the source of answers, which fall largely in two categories: **structured data** and **unstructured text**.

A long line of research aims to answer user questions using a structured *knowledge base* (KB) (Berant et al., 2013; Yih et al., 2015), known as **KBQA**. Typically, a KB can be viewed as a knowledge graph consisting of entities, properties, and a predefined set of relations between them. A question can be answered, provided that it can be expressed within the language of relations and objects present in the knowledge graph. With a high-quality, carefully curated KB, answers can be extracted with fairly high precision. KBQA, however, struggles with low answer coverage due to the cost of curating an extensive KB, as well as the fact that many questions simply cannot be answered using a KB if the answers are not entities.

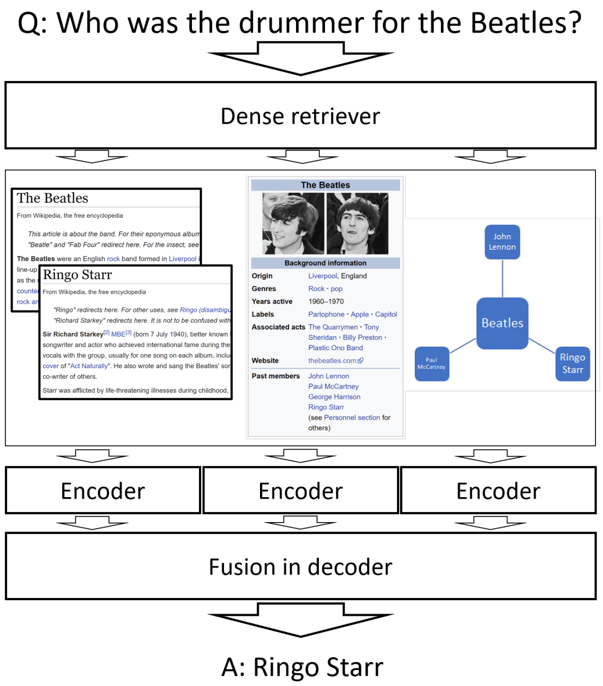

Figure 1: Illustration of `UniK-QA`'s workflow for unified-knowledge question answering: dense index retrieves Wikipedia passages, tables and knowledge base relations. Heterogeneous contexts are encoded independently through the encoder, then processed jointly in the decoder to generate the answer.

A second line of work targets a large collection of unstructured text (such as Wikipedia) (Chen et al., 2017) as the source of answers. Thanks to the latest advances in machine reading comprehension and text retrieval, substantial progress has been made for open-domain question answering from text (**TextQA**) in just the past couple years (Yang et al., 2019; Lee et al., 2019; Karpukhin et al., 2020; Guu et al., 2020; Izacard and Grave, 2020). On the other hand, semi-structured tables and structured KBs can be valuable knowledge sources, yet TextQA methods are restricted in taking only unstructured text as input, missing the opportunity of using these complementary sources of information to answer more questions.

When it comes to answering questions using both structured and unstructured information, a straightforward solution is combining specialized TextQA and KBQA systems. The input question is sent to multiple sub-systems, and one of them is selected to output the final answer. While this approach may take advantage of the state-of-the-art models designed for different information sources, the whole end-to-end system, however, becomes fairly complex. It is also difficult to handle questions that can only be answered when reasoning with information from multiple sources is required.

Having a more integrated system design that covers heterogeneous information sources has proven to be difficult. One main reason is that techniques used for KBQA and TextQA are drastically different. The former exploits the graph structure and/or semantic parsing to convert the question into a structured query, while TextQA has mostly settled on the retriever-reader architecture powered by pre-trained transformers. Recent work on multi-source QA has tried to incorporate free text into graph nodes (Sun et al., 2018; Lu et al., 2019) to make texts amenable to KBQA methods, but the performance remains unconvincing.

In this work, we propose a novel *unified knowledge representation* (`UniK-QA`) approach for open-domain question answering with heterogeneous information sources. Instead of having multiple specialized sub-systems or incorporating text into knowledge graphs, we *flatten* the structured data and apply TextQA methods. Our main motivation for doing so is to make the powerful machinary of pre-trained transformers available for structured QA. In addition, this approach opens the door to a simple and unified architecture. We can easily support semi-structured sources such as lists and tables, as well as fully structured knowledge bases. Moreover, there is no need to specially handle the schema or ontology that defines the structure of the KB, making it straightforward to support multiple KBs. Our `UniK-QA` model incorporates some 27 million passages composed of text and lists, 455,907 Wikipedia tables, and 3 billion relations from two knowledge bases in a single, unified open-domain QA model.

We first validate our approach by modeling KBQA as a pure TextQA task. We represent all relations in the KB with their textual surface form, and train a *retriever-reader* model on them as if they were text documents. This simple approach works incredibly well, improving the exact match score on the WebQSP dataset by 11% over previous state of the art. This result further justifies our choice of unifying multi-source QA under the TextQA framework as it can improve KBQA performance *per se*.

For our multi-source QA experiments, we consider lists, tables, and knowledge bases as sources of structured information, and convert each of them to text using simple heuristics. We model various combinations of structured sources with text, and evaluate on four popular open-domain QA datasets, ranging from entity-heavy KBQA benchmarks to those targeting free-form text sources. Our results indicate that while the best single source of information varies for each dataset as expected, our multi-source model improves over strong TextQA baselines in all cases. We obtain new state-of-the-art results for two datasets, advancing the published art on NaturalQuestions by 3.5 points and on WebQuestions by 2.6 points.

In addition, we consider the realistic setting in which the source of questions is not known *a priori*, as would be the case for a practical system. We train a single *multi-dataset* model on a combined dataset from several benchmarks, and show that it outperforms all single-source baselines across this diverse set of questions.

2 Background & Related Work

2.1 Knowledge-base question answering (KBQA)

A knowledge base (KB) considered in this work is a collection of facts, represented as a set of subject-predicate-object *triples*. Each triple (e_1, p, e_2) denotes a binary relationship between the subject entity e_1 and the object e_2 (e.g., places, persons, dates or numbers), as well as their relation type, or predicate p (e.g., *capital_of*, *married_to*, etc.).

Modern large-scale KBs, such as Freebase (Bollacker et al., 2008), DBPedia (Auer et al., 2007) and Wikidata (Vrandečić and Krötzsch, 2014) can contain billions of triples that describe relations between millions of entities, making them great sources of answers to open-domain questions. The prevailing approach for knowledge-base question answering (KBQA) is semantic parsing (Berant et al., 2013; Yih et al., 2015), where a natural language question is converted into a logical form that can be used to query the knowledge base. Such methods are tailored to the specific graph structure of the KB and are usually not directly applicable to other knowledge sources.

2.2 Open-domain question answering from text (TextQA)

KBQA is ultimately limited in its coverage of facts and the types of questions it can answer. On the other hand, large collections of text such as Wikipedia or CommonCrawl promise to be a richer source of knowledge for truly open domain question answering systems. This line of work (which we will refer to as TextQA) has been popularized by the TREC QA tracks (Voorhees, 1999), and has seen explosive growth with the advent of neural machine reading (MRC) (Rajpurkar et al., 2018) models. In the neural era, Chen et al. (2017) were the first to combine MRC with retrieval for end-to-end QA. Subsequent work cemented this *retriever-reader* paradigm, with improved reader models (Yang et al., 2019; Izacard and Grave, 2020) and neural retrievers (Lee et al., 2019; Guu et al., 2020; Karpukhin et al., 2020). Despite impressive advances, TextQA systems can still underperform KBQA, especially on benchmarks originally created for KBs such as WebQuestions. Furthermore, they also fall short of universal coverage, due to the exclusion of other (semi-)structured information sources such as tables.

2.3 Question answering from tables

Large amounts of authoritative data such as national statistics are often available in the form of tables. Even for simple, natural questions asked by users of a search engine, a significant fraction of them can be answered from tables (Kwiatkowski et al., 2019). While KBQA and TextQA have enjoyed increasing popularity, tables as a source of information has surprisingly escaped the attention of the community save for a few recent works.

Working with web tables can be challenging, due to the lack of formal schema, inconsistent formatting and ambiguous cell values (e.g., entity names). In contrast to relational databases and KBs, tables can at best be described as *semi-structured* information. Sun et al. (2016) considered open domain QA from web tables, however made no use of unstructured text. Some recent work investigated MRC with tables without a retrieval component (Pasupat and Liang, 2015; Yin et al., 2020; Chen et al., 2020b). In addition, Chen et al. (2020a,c) investigated open domain QA using tables and text. While they are in a similar direction, these works focus on complex, crowd-sourced questions requiring more specialized methods, while we target the case of simple, natural questions and investigate if popular TextQA and KBQA benchmarks can be further improved with the addition of tables.

2.4 Fusion of text and knowledge-base

As discussed, KBQA and TextQA are intuitively complementary, and several attempts have been made to merge them to get the benefits of both. An early example is (Ferrucci, 2012), which combines multiple expert systems and re-ranks them to produce the answer. More recent work attempts to enrich the KB by extracting structure from text. One way to accomplish this is using OpenIE triplets (Fader et al., 2014; Xu et al., 2016), thus staying completely within the semantic parsing paradigm. Somewhat closer to our approach are UniversalSchemas (Riedel et al., 2013; Das et al., 2017), which embed KB relations and textual relations in a common space. Yet, UniversalSchemas are also constrained to an entity-relation structure. The latest in this line are the works of (Sun et al., 2018, 2019), which augments the knowledge graph with text nodes and applies graph methods to identify candidate answers.

By retaining structure, previous work was able to take advantage of KBQA methods, but also failed to capture the full richness of TextQA. We depart radically in our approach, by foregoing all structure, and directly applying TextQA methods based on the more general *retriever-reader* architecture. We also evaluate on a more diverse benchmark set composed of natural open domain datasets, as well as those originally meant for KBQA, and demonstrate strong improvements in this truly open-domain setting. Concurrent work (Agarwal et al., 2020) proposed a similar idea for language model pretraining and also evaluated on open-domain QA. Our work differs in that (1) we have a more comprehensive treatment of sources (including tables, lists and multiple KBs) and ODQA datasets, (2) we compare against and improve on much stronger state-of-the-art baselines, and (3) we also evaluate in a more realistic multi-dataset setting with all datasets handled by a single model.

3 Modeling

3.1 `UniK-QA` architecture

We use a retriever-reader architecture, with *dense passage retriever* (DPR) (Karpukhin et al., 2020) as retriever and *fusion-in-decoder* (FiD) (Izacard and Grave, 2020) as our reader. Structured knowledge such as tables, lists and KB relations are converted to text with simple heuristics (§3.2, §3.3), and we generalize DPR to retrieve from these heterogeneous documents as well as regular text passages. Each retrieved document is concatenated with the question, then independently encoded by the reader encoder. Fusion of information happens in the decoder, which computes full attention over the entire concatenated input representations. The overall architecture is illustrated in Figure 1.

Retriever The DPR retriever consists of a dense document encoder and a question encoder, trained such that positive documents have embeddings closer to the question embedding in dot product space. We follow the original DPR implementation closely, starting from BERT-base (Devlin et al., 2019) encoders, using 100-token text passages, a single negative document per question while training with the same hyper-parameters. We further include tables, lists and KB relations in the index. The details of how these are processed into documents and merged are in the subsequent sections.

One improvement we make to the training process is iterative training, where better hard negatives are mined at each step using the model at the previous step, similar to (Xiong et al., 2020a). All models including our text-only baselines benefit from this change. We find 2 iterations sufficient.

Reader The FiD reader has demonstrated strong performance in the text-only setting and effective in fusing information from a large number of documents (Izacard and Grave, 2020). We thus find it a natural candidate for fusing knowledge from various sources. We use the FiD model with T5-large (Raffel et al., 2019), 100 context documents, and the original hyper-parameters for all experiments.

3.2 Unified representations for KBs

In order to apply our retriever-reader model, we first convert KB relations into text using simple

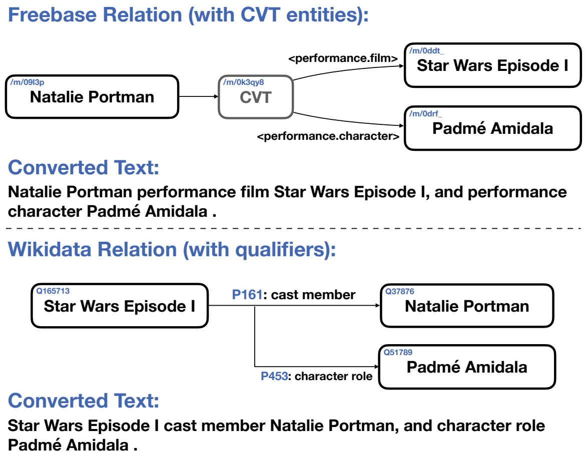

Figure 2: Converting Freebase and Wikidata relations to text.

heuristics. For a relation triple $\langle subj, pred, obj \rangle$, where $subj$, $pred$ and obj are the subject, predicate and object of the relation respectively, we serialize it by concatenating the text surface forms of $subj$, $pred$ and obj.

More complex (n-ary) relations involve multiple predicates and objects, such as *Natalie Portman played the character Padmé Amidala in the movie Star Wars*, and can be expressed differently depending on the KB. In particular, Freebase uses *compound value types* (CVTs) to convert an n-ary relation into multiple standard triples, while Wikidata allows a predicate to have *qualifiers* to express additional properties (Tanon et al., 2016). In this work, we convert an n-ary relation into a single sentence by forming a comma-separated clause for each predicate (Figure 2). A side benefit of this approach is that these complex relations are now represented as a single piece of text, whereas they would normally be considered multi-hop and require more complex methods (Fu et al., 2020) if using traditional graph-based KBQA models.

Once converted to text, relations can be indexed and retrieved using DPR. We index individual relations to best leverage the power of DPR for retrieving the most relevant relations for a given question[1]. Unlike most existing KBQA works, our approach can also seamlessly incorporate multiple KBs by storing all relations into a joint index and retrieving from it (see §5.4).

Directly indexing billions of relations in the entire KB can bring additional engineering challenges.

[1]Indexing at a coarser granularity (such as creating a document for each entity) also has practical challenges because certain entities (e.g., United States) may have hundreds of thousands of relations, resulting in extremely long documents.

To avoid these, we implement retrieval of relations in two steps, where an entity linking system is used in the first step to narrow down the search to a 2-hop neighborhood of the retrieved entities for each question (We use STAGG (Yih et al., 2015) in the case of Freebase and ELQ (Li et al., 2020) for Wikidata). We then use DPR to retrieve the top K relations from this reduced set. To be consistent with text input, we combine retrieved relations into documents of at most 100 tokens, after which they are fed to the FiD reader in the same way as text paragraphs.

3.3 Unified representations for tables

English Wikipedia contains more than 3 million tables ('classical' tables embedded in text as well as specialized tables like info-boxes), which are a huge source of factual knowledge by themselves and can substantially increase the coverage of open-domain QA systems. For instance, the answer to approximately a quarter of the questions in the NaturalQuestions (NQ) dataset can be found in Wikipedia tables (Kwiatkowski et al., 2019). These tables, however, have largely been ignored in recent open-domain QA work since it usually requires a dedicated model to reason over table structure. In contrast, we propose a simple approach to serialize tables and incorporate them into our `UniK-QA` framework like KB relations.

We start from a large subset of Wikipedia tables extracted and released as part of the NaturalQuestions dataset. We include all candidate documents which are part of the training set, extract nested tables into independent units, and filter out single-row tables as well as 'service' tables. This results in a corpus of 455,907 tables, which are used in our experiments.

As with KB relations, semi-structured content in tables need to be 'linearized' into text for the *retriever-reader* model to work. There are many ways to do such linearization (see Yin et al., 2020; Chen et al., 2020b). We tried two types of tables linearization: 'template'-like encoding used in recent literature (Chen et al., 2020b) and a simpler one which we find works the best in our experiments (see Table 4). In particular, we concatenate cell values on the same row, separated by commas, to form the text representation, and multiple rows are then combined into longer documents delimited by newlines.

As with TextQA, we divide linearized tables into 100-token chunks for indexing and retrieval. We take the first non-empty table row as the header and include it in every table chunk. This heuristic to select the first non-empty row as header is crucial and adds 4-6 points to top-20 passage accuracy.

Model	**Hits@1**
GraftNet (Sun et al., 2018)	67.8
PullNet (Sun et al., 2019)	68.1
EmQL (Sun et al., 2020)	*75.5**
Our KBQA (T5-base)	76.7
Our KBQA (T5-large)	**79.1**

Table 1: Hits@1 on WebQSP dataset using Freebase. (*)EmQL uses oracle entities, hence is not directly comparable with the others.

4 KBQA as TextQA: A Motivating Experiment

In this section, we present a motivating experiment showing that our `UniK-QA` approach not only provides a natural pathway to multi-source open-domain QA, but also improves KBQA per se. In particular, we evaluate our approach on a widely-used KBQA dataset, WebQSP (Yih et al., 2016), in the single-source setting.

We use Freebase as the knowledge source, and re-use pre-computed STAGG entity linking results and 2-hop neighborhoods as provided by Sun et al. (2018) for fair comparisons. We convert KB relations in the 2-hop neighborhood into text, retrieve the most relevant ones using DPR to form 100 context passages, and feed them into the T5 FiD reader as described in Section 3.2. The results are shown in Table 1, where the numbers represent *Hits@1*, or the percentage of the model's top-predicted answer being a "hit" (exact match) against one of the gold-standard answers.

We see that our KBQA method outperforms previous state-of-the-art methods by a wide margin, improving exact match accuracy to 79.1%. Since we adopt the exact same KB setup and pre-processing procedure from previous work, this improvement can be attributed purely to our `UniK-QA` model. We take this result as strong evidence for our claim that powerful TextQA methods generalize well to structured data, and offer a natural new framework for unifying structured and unstructured information sources.

5 Multi-Source QA Experiments

We now present our main experiments on unified multi-source question answering.

5.1 Datasets

For our main experiments, we use the same datasets that have recently become somewhat standard for evaluating open-domain QA (Lee et al., 2019):
NaturalQuestions (NQ) (Kwiatkowski et al., 2019) consists of questions mined from real Google search queries and Wikipedia articles with answer spans annotated. While the answer spans are usually on the regular, free-form text, some span annotations are in tables.
WebQuestions (WebQ) (Berant et al., 2013) targets Freebase as the source of answers, with questions coming from Google Suggest API.
TriviaQA (Trivia) (Joshi et al., 2017) contains a set of trivia questions with answers originally scraped from the Web.
CuratedTREC (TREC) (Baudiš and Šedivỳ, 2015) is a collection of questions from TREC QA tracks and various Web sources, intended to benchmark open-domain QA on unstructured text.

5.2 Combinations of sources

We compare 5 variations of our model, each with a different combination of information sources. We have *Text*-only, *Tables*-only and *KB*-only variants as single-source baselines. Next, the *Text + tables* model makes use of the entire Wikipedia dump, including lists and tables. Finally we add the KBs resulting in the *Text + tables + KB* model.

The *Text + tables* model uses a unified dense index, where text passages and table chunks are jointly indexed. For the *Text + tables + KB* model, since KB relations cannot be naturally chunked into 100-token documents for retrieval, we index them separately and then merge results with a fixed quota for KB relations. This quota is determined by maximizing retrieval recall on the development set. We also experiment with combining multiple KBs, which is straightforward with our approach, despite differences in structure.

5.3 A multi-dataset model

In a realistic setting, the best knowledge source to answer a given question is unknown *a priori* to the system, but most open-domain QA datasets are collected with respect to a specific information source (e.g., Wikipedia for NQ and Freebase for WebQ). To better simulate the real-world scenario, we also experiment with a setting where we train a single model on the combination of all 4 datasets and evaluate without any input to the model as to the source of questions.[2] We refer to this as the *multi-dataset* setting. We train multi-dataset models for all 5 variants described above. The smaller datasets, WebQ and TREC, are upsampled 5 and 8 times respectively while training.

5.4 Results

Main results are presented in Table 2. In the first set of experiments, we train a reader model independently for each dataset, as typically done in previous work. We use Freebase as knowledge base for WebQuestions as intended, and use Wikidata for all others. The multi-dataset model uses Wikidata.

The results highlight the limitation of current state-of-the-art open-domain QA models which use texts as the only information source. On WebQ, for instance, the KB-only model performs 5% better than the text-only one, and previous state of the art is also achieved by the KBQA model. Moreover, adding structured information sources significantly improves the performance over text-only models on *all* datasets, obtaining state-of-the-art results for NQ, WebQ and TREC. This indicates that KBs and tables contain valuable knowledge which is either absent in the unstructured texts or harder to extract from them (see also §6).

In the *multi-dataset* setting, we also observe clear improvements from combining sources, with the *Text + tables + KB* model outperforming the *Text*-only baseline by 5.4 points on average in this realistic setting. The performance is generally lower than the per-dataset models, especially for the small datasets (WebQ and TREC), which may be due to the fact that each of these datasets was collected on a single information source and the multi-dataset model is less likely to exploit this implicit prior knowledge.

Multiple KBs We also experiment with combining *both* Wikidata and Freebase. We see substantial improvements on all datasets in the KB-only setting over using a single KB, as well as significant gains over our best numbers for NQ and TriviaQA in the *Text+tables+KB* setting (Table 3).

[2] We normalize the questions by removing question marks and by presenting them in lowercase.

Model	NQ	WebQ	Trivia	TREC	Avg.
SoTA	51.4[1]	55.1[3]	67.6[1]	55.3[2]	57.3
Retrieval-free	28.5[4]	30.6[4]	28.7[4]	-	-
Per-dataset models					
Text	49.0	50.6	64.0	54.3	54.5
Tables	36.0	41.0	34.5	32.7	36.1
KB	27.9	55.6	35.4	32.4	37.8
Text + tables	**54.1**	50.2	**65.1**	53.9	55.8
Text + tables + KB	54.0	**57.8**	64.1	**55.3**	**57.8**
Multi-dataset model					
Text	50.3	45.0	62.6	45.7	50.9
Tables	34.2	38.4	33.7	31.1	34.4
KB	25.9	43.3	34.2	38.0	35.4
Text + tables	**54.6**	44.3	**64.0**	48.7	52.9
Text + tables + KB	53.7	**56.9**	63.4	**51.3**	**56.3**

Table 2: Exact match results on the test set. SoTA numbers are from (Izacard and Grave, 2020)[1], (Iyer et al., 2020)[2] which are TextQA approaches, and (Jain, 2016)[3], which is a KBQA method. (Jain, 2016) reports another metric; however, their predictions are available from which we calculated the EM score. Retrieval-free numbers refer to closed-book results from Roberts et al. (2020)[4] with the same T5 model.

Source(s)	NQ	WebQ	Trivia	TREC
KB-only (1 KB)	27.9	55.6	35.4	32.4
KB-only (2 KBs)	30.9	56.7	41.5	36.0
All (1 KB)	54.0	**57.8**	64.1	**55.3**
All (2 KBs)	**54.9**	57.7	**65.5**	54.0

Table 3: Results for combining Freebase and Wikidata.

Model	**R@20**	**R@100**
Text-only	80.0	85.9
w/ lists	82.7	89.6
w/ tables	83.1	91.0
w/ lists + tables	85.0	92.2
w/ lists + tables + KB	83.4	92.8
Tables-only		
simple linearization	86.3	94.3
template linearization	60.8	69.4

Table 4: Retrieval recall on the NQ dev set with different settings. Tables only results are for the NQ dev subset which has answers in tables.

6 Analysis

Having demonstrated that combining information sources does improve answer accuracy, we now provide more analysis on *how* this is achieved by inspecting both retriever and reader closely.

Retriever One natural assumption is that adding more data increases the coverage of relevant contexts that can be used to answer the input questions, thereby improving the end-to-end performance. We verify this by examining the retrieval results of different models using the NQ development set, where a context is considered relevant if it contains the correct answer string. When more knowledge sources are added, our system is able to improve retrieval *recall* (Table 4, top half), which may correlate with the end-to-end answer accuracy shown in Table 2.

Reader Although including additional information sources improves the chance of retrieving relevant contexts, it is not guaranteed that the reader can leverage those contexts and output the correct answers. For instance, reader model training may benefit from diverse sources of contexts, and the end-to-end improvement of answer accuracy may simply be attributed to a reader model that performs better on contexts from regular text. Due to the nature of the FiD generative reader, however, it is non-trivial to ascertain which input context(s) contribute the answer. As a proxy, we look at the correlation between the source of *positive* contexts (those which contain a correct answer string) feeding into the reader model and the performance change in the outcome.

Suppose we are comparing two reader models

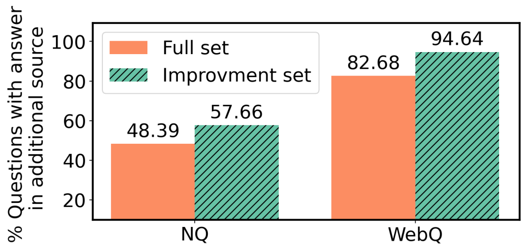

Figure 3: Percentage of questions with answers in additional sources. For NQ the additional sources are list and tables. For WebQ the additional source is KB.

M_u and M_t, where M_u uses additional sources of information compared to M_t (e.g., M_t uses text only and M_u uses text and KB). Let Q be all the questions in our development set, $Q_u \subseteq Q$ and $Q_t \subseteq Q$ the subsets of questions answered correctly by M_u and M_t, respectively. The *improvement* set $Q' = Q_u - Q_t$ is thus the questions that M_u manages to improve upon M_t. Examining the source of the positive contexts for the questions in Q' can help shed some light on how M_u performs better. For example, if more positive contexts are from KB rather than text, then the improvement is more likely due to additional information present at inference time. Figure 3 plots the percentages of positive contexts originating from the additional sources for the questions in the full development set (Q) vs those in the improvement set (Q') in two cases. The first one compares a baseline *text*-only model to a model with lists and tables added on NQ, and the second compares a *text+tables* model with *text+tables+KB* on WebQ. In both cases, answers retrieved from the additional source correlate with a better outcome.

To examine the effects of other indirect factors, such as the change of overall model quality due to the inclusion of varied sources or more training samples from the tables, we evaluate the *text + tables* model with text-only input. We find that this achieves a similar performance (48.7 EM) on the NQ test set compared to a *text*-only model on the same input, suggesting that these other factors are not a major contributor and that the improved performance is primarily due to the added knowledge from structured sources.

7 Discussion

We demonstrated a powerful new approach, `UniK-QA`, for unifying structured and unstructured information sources for open-domain question answering. We adopt the simple and general *retriever-reader* framework and show not only that it works for structured sources, but improves over traditional KBQA approaches by a wide margin. By combining sources in this way, we achieved new state-of-the-art results for two popular open-domain QA benchmarks.

However, our model also has several shortcomings in its current form. As a result of flattening all sources into text, we lose some desirable features of structured knowledge bases: the ability to return *all* answers corresponding to a query, and the ability to infer multi-hop paths to answer more complex questions. In this work we have side-stepped the first issue by focusing on the exact match metric (equivalent to Hits@1), which is standard in the open-domain QA literature, but largely ignores multiple answers. We were also able to ignore the second issue, since the datasets we evaluated on, while standard, are composed mostly of simple, natural user questions which can be answered from a single piece of information.

We do believe these are important details and they can be addressed within the framework described here. For instance, outgoing edges of an entity with the same relation can easily be merged, thus encoding all answer entities into a single text representation. It is also possible to simply generate multiple answer candidates from the reader's decoder. For multi-hop question answering, there is recent work (Xiong et al., 2020b) successfully extending dense retrieval to the multi-hop setting, which could naturally be applied within our framework. It remains to be seen how these approaches would compare to more traditional structured methods, and we leave this for future work.

References

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2020. Large scale knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Petr Baudiš and Jan Šedivỳ. 2015. Modeling of the question answering task in the yodaqa system. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 222–228. Springer.

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, Seattle, Washington, USA. Association for Computational Linguistics.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

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, Vancouver, Canada. Association for Computational Linguistics.

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, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020b. Tabfact: A large-scale dataset for table-based fact verification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020c. HybridQA: A dataset of multi-hop question answering over tabular and textual data. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1026–1036, Online. Association for Computational Linguistics.

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, Vancouver, Canada. Association for Computational Linguistics.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 1156–1165. ACM.

David A Ferrucci. 2012. Introduction to “This is Watson". *IBM Journal of Research and Development*, 56(3.4):1–1.

Bin Fu, Yunqi Qiu, Chengguang Tang, Yang Li, Haiyang Yu, and Jian Sun. 2020. A survey on complex question answering over knowledge base: Recent advances and challenges. *arXiv preprint arXiv:2007.13069*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.

Srinivasan Iyer, Sewon Min, Yashar Mehdad, and Wen-tau Yih. 2020. ReConsider: Re-ranking using span-focused cross-attention for open domain question answering. *arXiv preprint arXiv:2010.10757*.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Sarthak Jain. 2016. Question answering over knowledge base using factual memory networks. In *Proceedings of the NAACL Student Research Workshop*, pages 109–115, San Diego, California. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the*

57th Annual Meeting of the Association for Computational Linguistics, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Belinda Z. Li, Sewon Min, Srinivasan Iyer, Yashar Mehdad, and Wen-tau Yih. 2020. Efficient one-pass end-to-end entity linking for questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6433–6441, Online. Association for Computational Linguistics.

Vanessa Lopez, Victoria Uren, Marta Sabou, and Enrico Motta. 2011. Is question answering fit for the Semantic Web?: A survey. *Semantic web*, 2(2):125–155.

Xiaolu Lu, Soumajit Pramanik, Rishiraj Saha Roy, Abdalghani Abujabal, Yafang Wang, and Gerhard Weikum. 2019. Answering complex questions by joining multi-document evidence with quasi knowledge graphs. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 105–114. ACM.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. 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 1470–1480, Beijing, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84, Atlanta, Georgia. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Haitian Sun, Andrew O. Arnold, Tania Bedrax-Weiss, Fernando Pereira, and William W. Cohen. 2020. Faithful embeddings for knowledge base queries.

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, Hong Kong, China. Association for Computational Linguistics.

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, Brussels, Belgium. Association for Computational Linguistics.

Huan Sun, Hao Ma, Xiaodong He, Wen-tau Yih, Yu Su, and Xifeng Yan. 2016. Table cell search for question answering. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*, pages 771–782. ACM.

Thomas Pellissier Tanon, Denny Vrandecic, Sebastian Schaffert, Thomas Steiner, and Lydia Pintscher. 2016. From freebase to wikidata: The great migration. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*, pages 1419–1428. ACM.

Ellen M Voorhees. 1999. The TREC-8 question answering track report. In *TREC*, volume 99, pages 77–82.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020a. Approximate nearest neighbor negative contrastive learning for dense text retrieval.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, et al. 2020b. Answering complex open-domain questions with multi-hop dense retrieval. *arXiv preprint arXiv:2009.12756*.

Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2016. Hybrid question answering over knowledge base and free text. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2397–2407, Osaka, Japan. The COLING 2016 Organizing Committee.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Beijing, China. Association for Computational Linguistics.

Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, Berlin, Germany. Association for Computational Linguistics.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.