The Value of Semantic Parse Labeling for Knowledge Base Question Answering

Wen-tau Yih Matthew Richardson Christopher Meek Ming-Wei Chang Jina Suh
Microsoft Research
Redmond, WA 98052, USA
{scottyih,mattri,meek,minchang,jinsuh}@microsoft.com

Abstract

We demonstrate the value of collecting semantic parse labels for knowledge base question answering. In particular, (1) unlike previous studies on small-scale datasets, we show that learning from labeled semantic parses significantly improves overall performance, resulting in absolute 5 point gain compared to learning from answers, (2) we show that with an appropriate user interface, one can obtain semantic parses with high accuracy and at a cost comparable or lower than obtaining just answers, and (3) we have created and shared the largest semantic-parse labeled dataset to date in order to advance research in question answering.

1 Introduction

Semantic parsing is the mapping of text to a meaning representation. Early work on learning to build semantic parsers made use of datasets of questions and their associated semantic parses (Zelle and Mooney, 1996; Zettlemoyer and Collins, 2005; Wong and Mooney, 2007). Recent work on semantic parsing for knowledge base question-answering (KBQA) has called into question the value of collecting such semantic parse labels, with most recent KBQA semantic parsing systems being trained using only question-answer pairs instead of question-parse pairs. In fact, there is evidence that using only question-answer pairs can yield improved performance as compared with approaches based on semantic parse labels (Liang et al., 2013). It is also widely believed that collecting semantic parse labels can be a "difficult, time consuming task" (Clarke et al., 2010) even for domain experts. Furthermore, recent focus has been more on the final task-specific performance of a system (i.e., did it get the right answer for a question) as opposed to agreement on intermediate representations (Berant et al., 2013; Kwiatkowski et al., 2013), which allows for KBQA datasets to be built with only the answers to each question.

In this work, we re-examine the value of semantic parse labeling and demonstrate that semantic parse labels can provide substantial value for knowledge base question-answering. We focus on the task of question-answering on Freebase, using the WEBQUESTIONS dataset (Berant et al., 2013).

Our first contribution is the construction of the largest semantic parse dataset for KB question-answering to date. In order to evaluate the costs and benefits of gathering semantic parse labels, we created the WEBQUESTIONSSP dataset[1], which contains semantic parses for the questions from WEBQUESTIONS that are answerable using Freebase. In particular, we provide SPARQL queries for 4,737 questions. The remaining 18.5% of the original WEBQUESTIONS questions are labeled as "not answerable". This is due to a number of factors including the use of a more stringent assessment of "answerable", namely that the question be answerable via SPARQL rather than by returning or extracting information from textual descriptions. Compared to the previous semantic parse dataset on Freebase, Free917 (Cai and Yates, 2013), our WEBQUESTIONSSP is not only substantially larger, but also provides the semantic parses in SPARQL with standard Freebase entity identifiers, which are directly executable on Freebase.

Our second contribution is a demonstration that semantic parses can be collected at low cost. We employ a staged labeling paradigm that enables efficient labeling of semantic parses and improves the accuracy, consistency and efficiency of ob-

[1] Available at http://aka.ms/WebQSP.

taining answers. In fact, in a simple comparison with using a web browser to extract answers from `freebase.com`, we show that we can collect semantic parse labels at a comparable or even faster rate than simply collecting answers.

Our third contribution is an empirical demonstration that we can leverage the semantic parse labels to increase the accuracy of a state-of-the-art question-answering system. We use a system that currently achieves state-of-the-art performance on KBQA and show that augmenting its training with semantic parse labels leads to an absolute 5-point increase in average F_1.

Our work demonstrates that semantic parse labels can provide additional value over answer labels while, with the right labeling tools, being comparable in cost to collect. Besides accuracy gains, semantic parses also have further benefits in yielding answers that are more accurate and consistent, as well as being updatable if the knowledge base changes (for example, as facts are added or revised).

2 Collecting Semantic Parses

In order to verify the benefits of having labeled semantic parses, we completely re-annotated the WEBQUESTIONS dataset (Berant et al., 2013) such that it contains both semantic parses and the derived answers. We chose to annotate the questions with the full semantic parses in SPARQL, based on the schema and data of the latest and last version of Freebase (2015-08-09).

Labeling interface Writing SPARQL queries for natural language questions using a text editor is obviously not an efficient way to provide semantic parses even for experts. Therefore, we designed a staged, dialog-like user interface (UI) to improve the labeling efficiency. Our UI breaks the potentially complicated structured-labeling task into separate, but inter-dependent sub-tasks. Given a question, the UI first presents entities detected in the questions using an entity linking system (Yang and Chang, 2015), and asks the user to pick an entity in the question as the *topic entity* that could lead to the answers. The user can also suggest a new entity if none of the candidates returned by the entity linking system is correct. Once the entity is selected, the system then requests the user to pick the Freebase predicate that represents the *relationship* between the answers and this topic entity. Finally, additional *filters* can be added to further constrain the answers. One key advantage of our UI design is that the annotator only needs to focus on one particular sub-task during each stage. All of the choices made by the labeler are used to automatically construct a coherent semantic parse. Note that the user can easily go back and forth to each of these three stages and change the previous choices, before pressing the final submit button.

Take the question "*who voiced meg on family guy?*" for example. The labeler will be presented with two entity choices: `Meg Griffin` and `Family Guy`, where the former links "meg" to the character's entity and the latter links to the TV show. Depending on the entity selected, legitimate Freebase predicates of the selected entity will be shown, along with the objects (either properties or entities). Suppose the labeler chooses `Meg Griffin` as the topic entity. He should then pick `actor` as the main relationship, meaning the answer should be the persons who have played this role. To accurately describe the question, the labeler should add additional filters like the TV series is `Family Guy` and the performance type is `voice` in the final stage[2].

The design of our UI is inspired by recent work on semantic parsing that has been applied to the WEBQUESTIONS dataset (Bast and Haussmann, 2015; Reddy et al., 2014; Berant and Liang, 2014; Yih et al., 2015), as these approaches use a simpler and yet more restricted semantic representation than first-order logic expressions. Following the notion of *query graph* in (Yih et al., 2015), the semantic parse is *anchored* to one of the entities in the question as the *topic entity* and the core component is to represent the relation between the entity and the answer, referred as the *inferential chain*. *Constraints*, such as properties of the answer or additional conditions the relation needs to hold, are captured as well. Figure 1 shows an example of these annotated semantic parse components and the corresponding SPARQL query. While it is clear that our UI does not cover complicated, highly compositional questions, most questions in WEBQUESTIONS can be covered[3].

Labeling process In order to ensure the data quality, we recruit five annotators who are familiar with design of Freebase. Our goal is to provide

[2] Screenshots are included in the supplementary material.

[3] We manually edited the SPARQL queries for about 3.1% of the questions in WEBQUESTIONS that are not expressible by our UI.

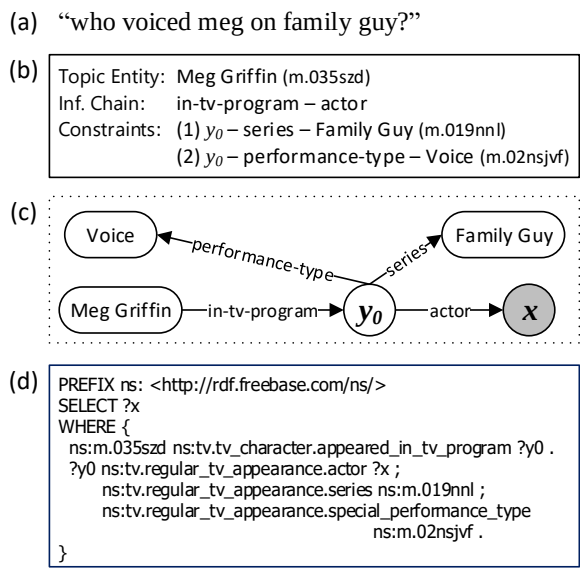

Figure 1: Example semantic parse of the question (a) "who voiced meg on family guy?" The three components in (b) record the labels collected through our dialog-like user interface, and can be mapped deterministically to either the corresponding query graph (c) or the SPARQL query (d).

correct semantic parses for each of the legitimate and unambiguous questions in WEBQUESTIONS. Our labeling instructions (included in the supplementary material) follow several key principles. For instance, the annotators should focus on giving the correct semantic parse of a question, based on the assumption that it will result in correct answers if the KB is *complete* and *correct*.

Among all the 5,810 questions in WEBQUESTIONS, there are 1,073 questions that the annotators cannot provide the complete parses to find the answers, due to issues with the questions or Freebase. For example, some questions are ambiguous and without clear intent (e.g., "*where did romans go?*"). Others are questions that Freebase is not the appropriate information source (e.g., "*where to watch tv online for free in canada?*").

3 Using Semantic Parses

In order to compare two training paradigms, learning from *question-answer pairs* and learning from *semantic parses*, we adopt the Staged Query Graph Generation (STAGG) algorithm (Yih et al., 2015), which achieves the highest published answer prediction accuracy on the WEBQUESTIONS dataset. STAGG formulates the output semantic parse in a query graph representation that mimics the design of a graph knowledge base. It searches over potential query graphs for a question, iteratively growing the query graph by sequentially adding a main *topic entity*, then adding an *inferential chain* and finally adding a set of *constraints*. During the search process, each candidate query graph is judged by a scoring function on how likely the graph is a correct parse, based on features indicating how each individual component matches the original question, as well as some properties of the whole query graph. Example features include the score output by the entity linking system, the match score of the inferential chain to the relation described in the question from a deep neural network model, number of nodes in the candidate query graph, and the number of matching words in constraints. For additional details see (Yih et al., 2015).

When question-answer pairs are available, we create a set of query graphs connecting entities in the question to the answers in the training set, as in (Yih et al., 2015). We score the quality of a query graph by using the F_1 score between the answer derived from the query graph and the answer in the training set. These scores are then used in a learning-to-rank approach to predict high-quality query graphs.

In the case that semantic parses are available, we change the score that we use for evaluating the quality of a query graph. In particular, we assign the query graph score to be zero whenever the query graph is not a subgraph consistent with the semantic parse label and to be the F_1 score described above otherwise. The hope is that by leveraging the semantic parse, we can significantly reduce the number of incorrect query graphs used during training. For instance, the predicate `music.artist.track` was incorrectly predicted as the inferential chain for the question "*what are the songs that justin bieber write?*", where a correct parse should use the relation `music.composer.compositions`.

4 The Value of Semantic Parses

In this section, we explore the costs of collecting semantic parse labels and the benefits of using them.

4.1 Benefits of Semantic Parses

Leveraging the new dataset, we study whether a semantic parser learned using full parses instead

Training Signals	Prec.	Rec.	Avg. F_1	Acc.
Answers	67.3	73.1	66.8	58.8
Sem. Parses	70.9	80.3	71.7	63.9

Table 1: The results of two different model training settings: answers only vs. semantic parses.

of just question-answer pairs can answer questions more accurately, using the knowledge base. Below, we describe our basic experimental setting and report the main results.

Experimental setting We followed the same training/testing splits as in the original WEBQUESTIONS dataset, but only used questions with complete parses and answers for training and evaluation in our experiments. In the end, 3,098 questions are used for model training and 1,639 questions are used for evaluation[4]. Because there can be multiple answers to a question, precision, recall and F_1 are computed for each individual question. The average F_1 score is reported as the main evaluation metric. In addition, we also report the true accuracy – a question is considered answered correctly only when the predicted answers exactly match one of the answer sets.

Results Table 1 shows the results of two different models: learning from question-answer pairs vs. learning from semantic parses. With the labeled parses, the average F_1 score is 4.9-point higher (71.7% vs. 66.8%). The stricter metric, complete answer set accuracy, also reflects the same trend, where the accuracy of training with labeled parses is 5.1% higher than using only the answers (63.9% vs. 58.8%).

While it is expected that training using the annotated parses could result in a better model, it is still interesting to see the performance gap, especially when the evaluation is on the correctness of the answers rather than the parses. We examined the output answers to the questions where the two models differ. Although the setting of using answers only often guesses the correct relations connecting the topic entity and answers, it can be confused by related, but incorrect relations as well. Similar phenomena also occur on constraints, which suggests that subtle differences in the meaning are difficult to catch if the semantic parses are automatically generated using only the answers.

[4]The average F_1 score of the original STAGG's output to these 1,639 questions is 60.3%, evaluated using WEBQUESTIONS. Note that the number is not directly comparable to what we report in Table 1 because many of the labeled answers in WEBQUESTIONS are either incorrect or incomplete.

Labeling Methods	Ans.	Ans.	Sem. Parses
Annotator	MTurkers	Experts	Experts
Avg. time/Question	Unknown	82 sec	21 sec
Labeling Correctness	66%	92%	94%

Table 2: Comparing labeling methods on 50 sampled questions.

4.2 Costs of Semantic Parses

Our labeling process is very different from that of the original WEBQUESTIONS dataset, where the question is paired with answers found on the Freebase Website by Amazon MTurk workers. To compare these two annotation methods, we sampled 50 questions and had one expert label them using two schemes: finding answers using the Freebase Website and labeling the semantic parses using our UI. The time needed, as well as the correctness of the answers are summarized in Table 2.

Interestingly, in this study we found that it actually took less time to label these questions with semantic parses using our UI, than to label with only answers. There could be several possible explanations. First, as many questions in this dataset are actually "simple" and do not need complicated compositional structured semantic parses, our UI can help make the labeling process very efficient. By ranking the possible linked entities and likely relations, the annotators are able to pick the correct component labels fairly easily. In contrast, simple questions may have many legitimate answers. Enumerating all of the correct answers can take significantly longer than authoring a semantic parse that computes them.

When we compare the annotation quality between labeling semantic parses and answers, we find that the correctness[5] of the answers are about the same (92% vs 94%). In the original WEBQUESTIONS dataset, only 66% of the answers are completely correct. This is largely due to the low accuracy (42.9%) of the 14 questions containing multiple answers. This indicates that to ensure data quality, more verification is needed when leveraging crowdsourcing.

5 Discussion

Unlike the work of (Liang et al., 2013; Clarke et al., 2010), we demonstrate that semantic parses

[5]We considered a label to be correct only if the derived/labeled answer set is completely accurate.

can improve over state-of-the-art knowledge base question answering systems. There are a number of potential differences that are likely to contribute to this finding. Unlike previous work, we compare training with answers and training with semantic parses while making only minimal changes in a state-of-the-art training algorithm. This enables a more direct evaluation of the potential benefits of using semantic parses. Second, and perhaps the more significant difference, is that our evaluation is based on Freebase which is significantly larger than the knowledge bases used in the previous work. We suspect that the gains provided by semantic parse labels are due a significant reduction in the number of paths between candidate entities and answers when we limit to semantically valid paths. However, in domains where the number of potential paths between candidate entities and answers is small, the value of collecting semantic parse labels might also be small.

Semantic parsing labels provide additional benefits. For example, collecting semantic parse labels relative to a knowledge base can ensure that the answers are more faithful to the knowledge base and better captures which questions are answerable by the knowledge base. Moreover, by creating semantic parses using a labeling system based on the target knowledge base, the correctness and completeness of answers can be improved. This is especially true for question that have large answer sets. Finally, semantic labels are more robust to changes in knowledge base facts because answers can be computed via execution of the semantic representation for the question. For instance, the answer to "Who does Chris Hemsworth have a baby with?" might change if the knowledge base is updated with new facts about children but the semantic parse would not need to change.

Notice that besides being used for the full semantic parsing task, our WEBQUESTIONSSP dataset is a good test bed for several important semantic tasks as well. For instance, the topic entity annotations are beneficial to training and testing entity linking systems. The core inferential chains alone are quality annotations for relation extraction and matching. Specific types of constraints are useful too. For example, the temporal semantic labels are valuable for identifying temporal expressions and their time spans. Because our dataset specifically focuses on questions, it complements existing datasets in these individual tasks, as they tend to target at normal corpora of regular sentences.

While our labeling interface design was aimed at supporting labeling experts, it would be valuable to enable crowdsourcing workers to provide semantic parse labels. One promising approach is to use a more dialog-driven interface using natural language (similar to (He et al., 2015)). Such UI design is also crucial for extending our work to handling more complicated questions. For instance, allowing users to traverse longer paths in a sequential manner will increase the expressiveness of the output parses, both in the core relation and constraints. Displaying a small knowledge graph centered at the selected entities and relations may help users explore alternative relations more effectively as well.

Acknowledgments

We thank Andrei Aron for the initial design of the labeling interface.

References

Hannah Bast and Elmar Haussmann. 2015. More accurate question answering on Freebase. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1431–1440. ACM.

Jonathan Berant and Percy Liang. 2014. Semantic parsing via paraphrasing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1415–1425, Baltimore, Maryland, June. Association for Computational Linguistics.

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, October. Association for Computational Linguistics.

Qingqing Cai and Alexander Yates. 2013. Large-scale semantic parsing via schema matching and lexicon extension. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 423–433, Sofia, Bulgaria, August. Association for Computational Linguistics.

James Clarke, Dan Goldwasser, Ming-Wei Chang, and Dan Roth. 2010. Driving semantic parsing from

the world's response. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 18–27. Association for Computational Linguistics.

Luheng He, Mike Lewis, and Luke Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653, Lisbon, Portugal, September. Association for Computational Linguistics.

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1556, Seattle, Washington, USA, October. Association for Computational Linguistics.

Percy Liang, Michael I Jordan, and Dan Klein. 2013. Learning dependency-based compositional semantics. *Computational Linguistics*, 39(2):389–446.

Siva Reddy, Mirella Lapata, and Mark Steedman. 2014. Large-scale semantic parsing without question-answer pairs. *Transactions of the Association for Computational Linguistics*, 2:377–392.

Yuk Wah Wong and Raymond J Mooney. 2007. Learning synchronous grammars for semantic parsing with lambda calculus. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yi Yang and Ming-Wei Chang. 2015. S-MART: Novel tree-based structured learning algorithms applied to tweet entity linking. 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 504–513, Beijing, China, July. 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, July. Association for Computational Linguistics.

John Zelle and Raymond Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1050–1055.

Luke S Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Conference on Uncertainty in Artificial Intelligence (UAI)*.