A Corpus for Complex Question Answering over Knowledge Graphs

Priyansh Trivedi[1], Gaurav Maheshwari[1], Mohnish Dubey[1], and Jens Lehmann[1,2]

[1] University of Bonn, Germany
{priyansh.trivedi, gaurav.maheshwari}@uni-bonn.de {dubey, jens.lehmann}@cs.uni-bonn.de
[2] Fraunhofer IAIS, Bonn, Germany
jens.lehmann@iais.fraunhofer.de

Resource Type: Dataset
Website and documentation: `http://sda.tech/qa-dataset/`
Permanent URL: `https://figshare.com/projects/LC-QuAD/21812`

Abstract. Being able to access knowledge bases in an intuitive way has been an active area of research over the past years. In particular, several question answering (QA) approaches which allow to query RDF datasets in natural language have been developed as they allow end users to access knowledge without needing to learn the schema of a knowledge base and learn a formal query language. To foster this research area, several training datasets have been created, e.g. in the QALD (Question Answering over Linked Data) initiative. However, existing datasets are insufficient in terms of size, variety or complexity to apply and evaluate a range of machine learning based QA approaches for learning complex SPARQL queries. With the provision of the Large-Scale Complex Question Answering Dataset (LC-QuAD), we close this gap by providing a dataset with 5000 questions and their corresponding SPARQL queries over the DBpedia dataset. In this article, we describe the dataset creation process and how we ensure a high variety of questions, which should enable to assess the robustness and accuracy of the next generation of QA systems for knowledge graphs.

1 Introduction

With the advent of large scale knowledge bases (KBs), such as DBpedia [7], Freebase [1], and Wikidata [12], Question Answering (QA) over structured data has become a major research topic (see [6] for a survey on QA systems). QA systems over structured data, as defined in [6] is users asking questions in natural language in their own terminology and receiving a concise answer from the system. Using structured data as their background knowledge, these systems frequently model the QA problem as that of conversion of natural language questions (NLQ) to a formal query language expression, such as SPARQL or λ-Calculus expressions.

One of the pivotal requirements to evaluate and solve the QA problem, as we will discuss in detail in Section 2, is the availability of a large dataset comprising of varied questions and their logical forms. In this direction, we introduce the LC-QuAD (Large-Scale Complex Question Answering Dataset) dataset. LC-QuAD consists of

5000 questions along with the intended SPARQL queries required to answer questions over DBpedia. The dataset includes complex questions, i.e. questions in which the intended SPARQL query does not consist of a single triple pattern. We use the term “complex" to distinguish the dataset from the simple questions corpus described in SimpleQuestions [2]. To the best of our knowledge, this is the largest QA dataset including complex questions with the next largest being Free917 [3] with 917 questions and QALD-6 [11] with 450 training questions and 100 test questions, respectively.

We frame our question generation problem as a transduction problem, similar to [10], in which KB *subgraphs* generated by the *seed entity* is fitted into a set of *SPARQL* templates which are then converted into a Normalized Natural Question Template (NNQT). This acts as a canonical structure which is then manually transformed into an NLQ having lexical and syntactic variations. Finally, a review is performed to increase the quality of the dataset.

The main contributions are as follows:

1. A dataset of 5000 questions with their intended SPARQL queries for DBpedia. The questions exhibit large syntactic and structural variations.
2. A framework for generating NLQs and their SPARQL queries which reduces the need for manual intervention.

The article is organized into the following sections: (2) Relevance, where the importance of the resource is discussed; (3) Dataset Creation Workflow, where the approach of creating the dataset is discussed; (4) Dataset Characteristics; in which various statistics about the dataset are discussed; (5) Availability & Sustainability, describing the accessibility and long term preservation of the dataset; and (6) Conclusion & Future Work, summarizing and describing future possibilities.

2 Relevance

Relevance for Question Answering Research: Question answering approaches over structured data typically fall into two categories (as described in [14]): (i) *semantic parsing* based methods where the focus is to construct a semantic parser which can convert NLQs to an intermediate form, and then convert the intermediate form into a logical form, and (ii) *information retrieval* based techniques, which convert NLQs to a formal query language expression or directly to an answer, usually without any explicit intermediary form.

Approaches in the first category (semantic parsing based methods), frequently rely on handmade rules [4,6]. Naturally, a goal of current research is to automate these manual steps. However, the size of the currently available training datasets is limited. The maximum size of the SPARQL-based QA dataset is 450 queries [11] and for λ-Calculus, the maximum size is 917 queries [3]. Due to these size limitations, it is currently unknown to what extent can these manual steps be automated. In particular, the relation between the size of a dataset, and the improvement in accuracy of employed ML techniques is unknown. The provision of LC-QuAD will allow to address these research questions in the future publication of semantic parsing based approaches.

Recent approaches in the second category (information retrieval based) are based on neural networks and have achieved promising results [2,8]. However, these techniques are currently limited to answering simple questions, i.e. those which can be answered using a SPARQL query with a single triple pattern. Many queries are not simple: Comparative questions (e.g. "Was John Oliver born before Jon Stewart?"), boolean questions (e.g. "Is Poland a part of Eurozone?"), questions involving fact aggregation (e.g. "Who has won the most Grammy awards?"), or even logically composite question (e.g. "In which university did both Christopher Manning and Sebastian Thrun teach?") cannot be answered by a system restricted to simple questions. We believe that it would be very interesting to explore neural network based approaches also for answering these complex questions. LC-QuAD provides initial foundations for exploring this research direction. While 5000 questions are likely insufficient in the long term, it should also be noted that the dataset size can be increased substantially by entity replacement (see Section 6). This dataset may enable neural networks based QA system to process a much larger variety of questions, and may lead to a substantial increase in their F-score.

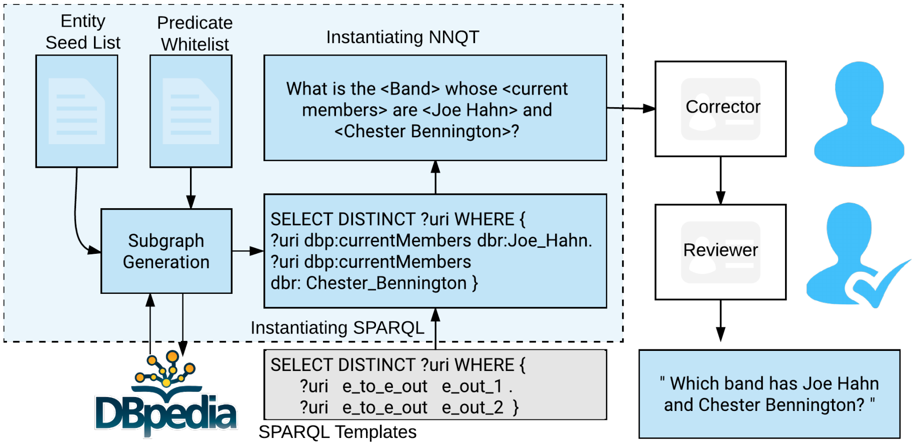
Relevance for Other Research Areas

- **Entity and Predicate Linking**: During the expert intervention part of the workflow (see Section 3), the tokens referring to entities and predicates in the SPARQL query were edited as well. As a result, our dataset can be treated as a set of questions, along with a corresponding list of entities and predicates present in it. There are 5000 total questions, 615 predicates and 5042 entites in the dataset. In future work, we will release a version of the dataset where the questions are annotated with RDF entities.
- **SPARQL Verbalization**: This dataset can also assist the task of SPARQL verbalization, which has attracted research interest in the Semantic Web community [5,9].

Relevance of and for the Semantic Web Community A significant portion of research in question answering over structured data has been done on non-RDF knowledge graphs [8,13]. This could be attributed in part to the absence of large-scale QA datasets which use semantic technologies. By closing this gap via LC-QuAD, we believe that there can be a two fold benefit: On the one hand, researchers in question answering outside of the Semantic Web community can benefit from existing W3C standards, such as SPARQL, as a framework for formalizing and approaching the QA problem. While, on the other hand, the Semantic Web community itself will be more centrally positioned in the area of question answering.

3 Dataset Generation Workflow

The primary objective while designing the framework for question generation was to generate a high quality large dataset with low domain expert intervention. In both QALD-6 [11], and Free917 [3], the logical forms of the questions were generated manually. This process of writing formal expressions needs domain experts with a deep understanding of the underlying KB schema, and syntaxes of the logical form. Secondly, following this approach makes the data more susceptible to human errors, as unlike natural language, formal languages are not fault tolerant.


Entity Seed List
Predicate Whitelist
Instantiating NNQT
What is the <Band> whose <current members> are <Joe Hahn> and <Chester Bennington>?
Corrector
Subgraph Generation
Reviewer
Instantiating SPARQL
DBpedia
SPARQL Templates
" Which band has Joe Hahn and Chester Bennington? "

```
SELECT DISTINCT ?uri WHERE {
?uri dbp:currentMembers dbr:Joe_Hahn.
?uri dbp:currentMembers
dbr: Chester_Bennington }
```

```
SELECT DISTINCT ?uri WHERE {
    ?uri   e_to_e_out   e_out_1 .
    ?uri   e_to_e_out   e_out_2 }
```

Fig. 1. Using a list of seed entities, and filtering by a predicate whitelist, we generate subgraphs of DBpedia to insantiate SPARQL templates, thereby generating valid SPARQL queries. These SPARQL queries are then used to instantiate NNQTs and generate questions (which are often grammatically incorrect). These questions are manually corrected and paraphrased. This is then reviewed and optionally edited by the reviewer.

To avoid these aforementioned shortcomings, instead of starting with NLQs and manually writing their corresponding logical forms, we invert the process. Figure 1 provides a outline of our dataset generation framework. It begins by creating a set of SPARQL templates[3], a list of seed entities[4], and a predicate whitelist[5]. Then, for each entity in the list of seed entities, we extract subgraphs from DBpedia. Here, each subgraph contains triples within a 2-hop distance from the seed entity in the RDF graph. We then interpret the templates to create valid SPARQL queries using the triples in the subgraph. It is to be noted that while we use DBpedia resources to create these lists, this framework can be generalized to any target knowledge base.

The previously described approach generates SPARQL queries with non-empty results over the target knowledge base. However, as human intervention is required to paraphrase each query into a question, we avoid generating similar questions. Herein, we define two questions to be similar if they have same SPARQL template, same predicates, and entities of same RDF class, which, when verbalized would also have a *similar* syntactic structure. For instance, Q1: *What is the capital of Germany?* has the following logical expression: `SELECT ?uri WHERE {dbr:Germany dbo:capital ?uri .}`. This question is similar to Q2: *What is the capital of France?* whose logical form is `SELECT ?uri WHERE {dbr:France dbo:capital ?uri .}`. Thus, in order to achieve more variations in our dataset with the same amount of human work, we prune the subgraphs to avoid generation of similar questions. In the future, we aim to automatically increase the size of our dataset by replacing entities (e.g. Germany in Q1) with entities of the same class (e.g. France in Q2).

[3] https://github.com/SmartDataAnalytics/LC-QuAD/blob/develop/templates.py

[4] https://github.com/SmartDataAnalytics/LC-QuAD/blob/develop/resources/entities.txt

[5] https://github.com/SmartDataAnalytics/LC-QuAD/blob/develop/resources/relations.txt

Template	`SELECT DISTINCT ?uri WHERE { ?x e_in_to_e_in_out e_in_out . ?x e_in_to_e ?uri }`
Query	`SELECT DISTINCT ?uri WHERE { ?x dbp:league dbr:Turkish_Handball_Super_League . ?x dbp:mascot ?uri }`
NNQT Instance	What is the <mascot> of the <handball team> whose <league> is <Turkish Handball Super League >?
Question	What are the mascots of the teams participating in the turkish handball super league?
Template	`SELECT DISTINCT ?uri WHERE { ?x e_out_to_e_out_out e_out_out . ?uri e_to_e_out ?x }`
Query	`SELECT DISTINCT ?uri WHERE { ?x dbo:award dbr:BAFTA_Award_for_Best_Film_Music . ?uri dbo:musicComposer ?x }`
NNQT Instance	List the <movies> whose <music composer>'s <honorary title> is <BAFTA Award for Best Film Music>.?
Question	List down the movies whose music composers have won the BAFTA Award for Best Film Music ?

Table 1. Some Examples from LC-QuAD

Our dataset is characteristic of the target KB, i.e. DBpedia. Thus, inconsistencies or semantically invalid triples in the KB can percolate into the dataset in the form of nonsensical questions. Since DBpedia has a lot of predicates which are used for metadata purposes, and are not of immediate semantic information[6], those should be avoided in the question generation process. To avoid these triples, we create a whitelist of 615 DBpedia predicates, and trim all the triples in the subgraph whose predicate is not in the whitelist.

Thereafter, we create an equivalent natural language template for every SPARQL template, called Normalized Natural Question Templates (NNQT). These are then instantiated to generate NLQs corresponding to every SPARQL query. The generated NLQs are often grammatically incorrect, but can be used by humans as a base for manual paraphrasing. The grammatical errors are due to fact that surface forms of DBpedia predicates correspond to varying parts of speech. For instance, while *president* is a noun, *largest city* is a modified noun, *bought* is a verb, whereas *born in* a prepositional phrase. These variations, along with complex entity surface forms (e.g. *2009 FIFA Club World Cup squads*) create a need for manual intervention to correct the grammar and paraphrase the questions. This task can be done by fluent english speakers, who are not required to understand formal query languages, or the underlying schema of the KB. In this manner, using NNQT, we transduce the task of interpreting and verbalizing SPARQL queries, to a simpler task of grammar correction and paraphrasing, and thereby reduce the domain expertise required for our dataset generation process.

Finally, every question is reviewed by an independent reviewer. This second iteration ensures a higher quality of data, since the reviewer is also allowed to edit the questions in case any errors are found.

[6] For e.g., dbo:abstract, dbo:soundRecording, dbo:thumbnail, dbo:wikiPageExternalLink, dbo:filename etc

4 Dataset Characteristics

Table 2 compares some statistics of QA Datasets over structured data. While QALD has 450 questions and Free917 has 917, LC-QuAD has 5000 questions. As mentioned in Section 2, QALD is the only dataset based on DBpedia, therefore, in this section we describe the characteristics of our dataset in contrast to it. Although LC-QuAD is tenfold in size compared to it, questions in QALD dataset are more complex and colloquial as they have been created directly by domain experts. Since the questions in our dataset are not extracted out of some external source, they are not an accurate representative of actual questions asked, but are characteristic of the knowledge base on which they were made. Nevertheless, due to human paraphrasing of both syntactic structure of the questions as well as the surface forms of entities and predicates, the questions in our dataset resemble questions actually asked by humans.

On an average, every question in our dataset has 12.29 tokens. The manual paraphrasing process was done by the first three authors who are native English speakers. Although the time taken to paraphrase a question varies significantly depending on the SPARQL template it is based on, it took about 48 seconds on average to correct each question. After this, the final reviewer took about 20 seconds to complete verification and, if needed, further editing. On the other hand, when a randomly sampled set of 100 SPARQL queries from our dataset was given to the same people (without instantiated NNQTs), it took them about 94 seconds to verbalize a query. This indicates that our framework reduces the workload of creating QA datasets.[7]

Table 2. A comparison of datasets having questions and their corresponding logical forms

Data Set	Size	Entities	Predicates	Formal Lang.
QALD-6	450	383	378	SPARQL
Free917	917	733	852	λ-Calculus
LC-QuAD	5000	5042	615	SPARQL

Our dataset has 5042 entities and 615 predicates over 38 unique SPARQL templates. The SPARQL queries have been generated based on the most recent (2016-04) DBpedia release[8]. Among the 5000 verbalized SPARQL queries, only 18% are simple questions, and the remaining queries either involve more than one triple, or COUNT/ASK keyword, or both. Moreover, we have 18.06% queries with a COUNT based aggregate, and 9.57% boolean queries. As of now, we do not have queries with OPTIONAL, or UNION keyword in our dataset. Also, we do not have conditional aggregates in the query head.

[7] Naturally, the time required to start completely from scratch and think of a typical query and formalise it in SPARQL would be substantially higher and also lead to a low diversity from previous experience in the QALD challenge.

[8] http://wiki.dbpedia.org/downloads-2016-04

5 Availability and Sustainability

In this section, we describe the interfaces to access the dataset as well as how we plan to support sustainability. We have published our dataset on figshare[9] under CC BY 4.0[10] license. Figshare promises data persistence and public availability, thereby ensuring that the dataset should always be accessible regardless of the running status of our servers. The figshare project of LC-QuAD includes following files

- **LC-QuAD** - A JSON dump of Question Answering Dataset.
- **VoID description** - A machine readable description of the dataset in RDF.
- **Tertiary resources** - These include numerous resources, such as SPARQL templates, NNQTs, predicate whitelists etc. mentioned throughout the article.

Regarding sustainability, the dataset will be integrated into the QALD challenge – specifically in QALD-8 and beyond. QALD is running since 2011 and recently the HOBBIT EU project has taken over its maintenance. From 2019 on, the HOBBIT association will run the challenge.

Our framework is available as an open source repository[11], under a GPL 3.0[12] License. The documentation of the framework, and its user manual have been published on the repository's Wiki as well. We intend to actively use Github issues to track feature requests and bug reports. Lastly, we will also announce all the new updates of the framework and dataset on all public Semantic Web lists.

6 Conclusions and Future Work

In this article, we described a framework for generating QA dataset having questions and their equivalent logical forms. This framework aims to reduce human intervention thereby enabling creation of larger datasets with fewer errors. We used it to create a dataset, LC-QuAD, having 5000 questions and their corresponding SPARQLs. Although we used DBpedia as the target KB for our dataset, the framework is KB agnostic. We compared the characteristics of the dataset with pre-existing datasets and also described its shortcomings.

In the future, we aim to increase the number of SPARQL templates covered, thus increasing its syntactic variety. Moreover, to increase the size of the dataset by a certain factor, we can replace the entities in the questions with similar entities to synthetically add new questions. The software for this is already available and has been applied to create 2.1 million questions from 150 seed questions in QALD[13]. Increasing the dataset size in this way will likely benefit neural network based approaches for question answering as they learn the regularities in human language from scratch. However, this effect will diminish and estimating a factor up to which accuracy gains can be observed is subject for future work. Additionally, we plan to explore machine translation based

[9] https://figshare.com/projects/LC-QuAD/21812

[10] https://creativecommons.org/licenses/by/4.0/

[11] https://github.com/SmartDataAnalytics/LC-QuAD

[12] https://www.gnu.org/licenses/gpl.html

[13] https://github.com/hobbit-project/QuestionAnsweringBenchmark

techniques to reduce the need of manual grammar correction. As mentioned in Section 2, *in the upcoming version of LC-QuAD*, we will annotate the entities in every question, thereby enabling the dataset to be used for the entity linking task as well as exploring advanced techniques such as jointly trained entity linker and semantic parser.

Acknowledgements This work was partly supported by the grant from the European Union's Horizon 2020 research Europe flag and innovation programme for the projects Big Data Europe (GA no. 644564) and HOBBIT (GA no. 688227).

References

1. K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD conference on Management of data*, pages 1247–1250. AcM, 2008.
2. A. Bordes, N. Usunier, S. Chopra, and J. Weston. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*, 2015.
3. Q. Cai and A. Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *ACL (1)*, pages 423–433, 2013.
4. M. Dubey, S. Dasgupta, A. Sharma, K. Höffner, and J. Lehmann. Asknow: A framework for natural language query formalization in sparql. In *International Semantic Web Conference*, pages 300–316. Springer, 2016.
5. B. Ell, D. Vrandečić, and E. Simperl. Spartiqulation: Verbalizing sparql queries. In *Extended Semantic Web Conference*, pages 117–131. Springer, 2012.
6. K. Höffner, S. Walter, E. Marx, R. Usbeck, J. Lehmann, and A.-C. Ngonga Ngomo. Survey on challenges of question answering in the semantic web. *Semantic Web*, (Preprint):1–26, 2016.
7. J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, and C. Bizer. DBpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.
8. D. Lukovnikov, A. Fischer, J. Lehmann, and S. Auer. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th International World Wide Web Conference*, pages 1211–1220. International World Wide Web Conferences Steering Committee, 2017.
9. A.-C. Ngonga Ngomo, L. Bühmann, C. Unger, J. Lehmann, and D. Gerber. Sorry, i don't speak sparql: translating sparql queries into natural language. In *Proceedings of the 22nd international World Wide Web conference*, pages 977–988. ACM, 2013.
10. I. V. Serban, A. García-Durán, Ç. Gülçehre, S. Ahn, S. Chandar, A. Courville, and Y. Bengio. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. In *54th Annual Meeting of the Association for Computational Linguistics*, page 588. Association for Computational Linguistics, 2016.
11. C. Unger, A.-C. N. Ngomo, and E. Cabrio. 6th open challenge on question answering over linked data (qald-6). In *Semantic Web Evaluation Challenge*, pages 171–177. Springer, 2016.
12. D. Vrandečić. Wikidata: A new platform for collaborative data collection. In *Proceedings of the 21st International World Wide Web Conference*, pages 1063–1064. ACM, 2012.
13. S. W.-t. Yih, M.-W. Chang, X. He, and J. Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. 2015.
14. Y. Zhang, K. Liu, S. He, G. Ji, Z. Liu, H. Wu, and J. Zhao. Question answering over knowledge base with neural attention combining global knowledge information. *arXiv preprint arXiv:1606.00979*, 2016.