Question Answering Over Linked Data: What is Difficult to Answer? What Affects the F scores?

Muhammad Saleem[1], Samaneh Nazari Dastjerdi[2], Ricardo Usbeck[3], and Axel-Cyrille Ngonga Ngomo[1,3]

[1] University Leipzig, Germany
lastname@informatik.uni-leipzig.de
[2] TU Ilmenau, Germany
Samaneh.nazari-dastjerdi@tu-ilmenau.de
[3] University Paderborn, Germany
firstname.lastname@upb.de

Abstract. We present a fine-grained analysis of the Question Answering over Linked Data (QALD-6) challenge. We divide the QALD-6 questions into 8 main categories and compare state-of-the-art questions answering (QA) systems over Linked Data against the individual categories. We show the difficulty (in terms of overall F scores of the QA systems) of each category. We show the effect of various natural language and SPARQL features such as the number of triple patterns, number of keywords, the answer size, the type of answers, the effect of aggregate functions, and the SPARQL query forms on the overall F scores of the QA systems.

1 introduction

The SPARQL query language is a W3C standard[4] to retrieve data from the Linked Open Data cloud[5]. However, learning SPARQL for naive user can be tricky. In particular, formulating meaningful queries to retrieve the desired data is not a trivial task for the common users. To this end, the significant growth of the LOD datasets has motivated a considerable amount of works on question answering over Linked Data [1,2,3,4,5,9,10]. Consequently, this has motivated a good number of benchmarks (see, e.g., QALD [8,7,6]) to test QA systems. QALD is a series of question answering challenges over Linked Data. The questions are collected various sources such as real-life log files, users or experts. The questions are based on different versions of the DBpedia dataset and are provided in many languages such as English, Hindi, German, French etc. For each natural language question, QALD provides the corresponding SPARQL query, the exact answers, and the list of keywords extracted from the question. The overall goal is to compare QA systems over Linked Data with respect to different key performance indicators such as precision, accuracy, and F scores. Such benchmarks provided the possibility to analyze the strengths and weaknesses of many QA systems objectively.

[4] https://www.w3.org/TR/rdf-sparql-query/
[5] http://lod-cloud.net/

While QALD benchmarks contain a variety of questions, the QALD challenge does not provide a fine-grained analysis of the questions itself or a detailed evaluation of the results. It is paramount to know what were the easy categories? Which category is in general hard to answer and why? Which features increase or decrease the complexity of the question? With which feature category does a system's performance correlate? Where do systems fail? In this report, we provide a fine-grained analysis of the QALD-6 challenge results. In particular, we are interested in the following questions:

- What are the general categories of QALD-6 questions? Which feature can be derived?
- How do QA systems over Linked Data perform across the different question categories?
- Which types of questions are hard to answer and which are relatively easy? Why?
- What is the effect of the number of triple patterns on the performance?
- What is the effect of the number of keywords on the performance?
- What is the effect of the number of answers on the performance?
- What is the effect of answer types (e.g., String, Date, Resource, Boolean etc.) on the performance?
- Do SPARQL aggregates such as sum, min, max or avg, increase or decrease the performance?

It is important to mention that in this paper we are not presenting the details of the QA systems over Linked Data. Rather, we are interested in the detailed analysis of the QALD-6 questions and its corresponding results. The description of the QALD-6 participated QA systems can be found at [8].

2 Results and Discussion

In this section we provide a fine-grained analysis of QALD-6 results. In particular, we first divide the overall QALD-6 questions in 8 main categories. We then show the performance of the QA systems (participated at QALD-6 challenge) for individual categories. We then discuss the complexity of each category in terms of how difficult it is correctly answer. After that, we measure the effect of various performance indicators on the F scores of the QA systems. The overall goal is to find answers for each of the questions discussed in previous section and pinpoint limitations of QA systems.

2.1 QALD-6 Questions Categories and Result

First, we want to investigate what the general categories of QALD-6 questions are? To this end, we categories the complete 100 QALD-6 questions into 8 categories given in Table 1. We categorized the questions according to their starting word. We can see that the questions of type *"Who?"* (total 21 questions)

Table 1: QALD-6 category-wise distribution of questions.

Who?	What?	Which?	How Many?	Give Me	When?	Where?	In Which?
21	22	12	8	10	6	3	9

and *"What?"* (total 22 question) are the majority. The questions of type *"When?"* and *"Where?"* (6 respectively 3)are not that common. The main conclusion by analysis is that QALD-6 questions are not uniformly distributed across the different categories. Thus, adding more questions of type *"Where?"* could be useful for the overall quality of QALD challenges in the future.

Once we know the general categories of the QALD-6 questions, next we want to investigate how QA systems over Linked Data perform across the different question categories?

Our main hypothesis is that a system can win the QALD challenge, even if it performs worse than other systems in some specific question category.

This fine-grained analysis will enable developers of QA systems to get to know their system shortcomings for a particular category and hence let them focus on these particular categories in future improvements. Figure 1 shows the comparison of the QA systems across the individual categories of the QALD-6 questions. We can see that CANALI is the overall winner for the challenge. However, it is not the winner across all the categories. For example, in the *"Give me?"* category, UTQA outperforms CANALI. This clearly suggest that CANALI developers should focus on correctly answering the questions starting with *"Give me"*. In conclusion, our hypothesis is proved that there is not a single winner across all the categories.

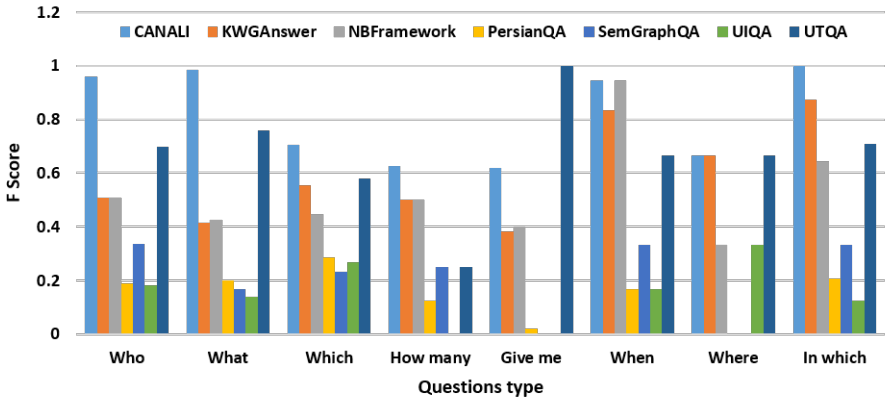


Fig. 1: Category-wise comparison of of the QA systems.

2.2 Complexities of QALD-6 Categories

The second research questions investigates the complexities of each category, i.e., which type of questions are hard to answer and which are relatively easy? Figure 2 shows the average F score for individual questions categories over all 6 QA systems. Note, that the F Score used within here is the macro F score, i.e., the average over all F scores for individual questions.[6] The results show that the questions of type *"When?"* (avg. F score = 0.57) is easier to correctly answer which is followed by category *"In Which?"* (avg. F score = 0.55), *"Who?"* (avg. F score = 0.48), *"What?"* (avg. F score = 0.44), *"Which?"* (avg. F score = 0.43), *"Where?"* (avg. F score = 0.38), *"Give Me?"* (avg. F score = 0.34), and *"How many?"* (avg. F score = 0.32). Interestingly, the questions starting with *"How Many?"* are the most difficult to answer. The result of such type of questions is a single value and the corresponding SPARQL query of these questions mostly use the COUNT function. However, some time it simply uses a property, e.g. How many inhabitants has France uses a property not a count. It seems that aggregates functions, e.g, count, min, max, avg, are hard to answer correctly. We will further investigate the effect of aggregates functions in Section 2.5.

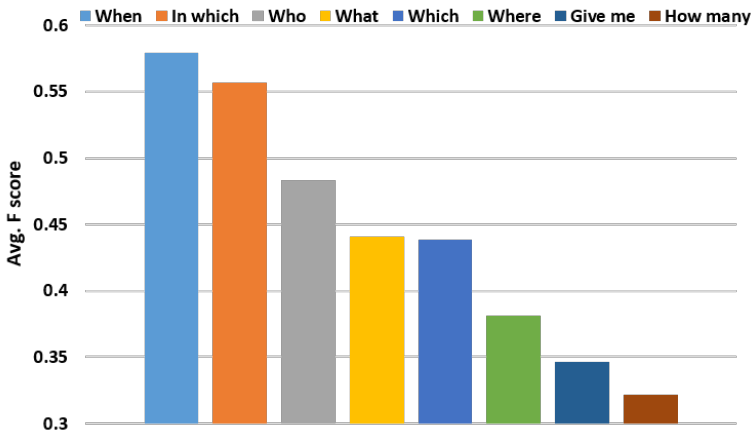
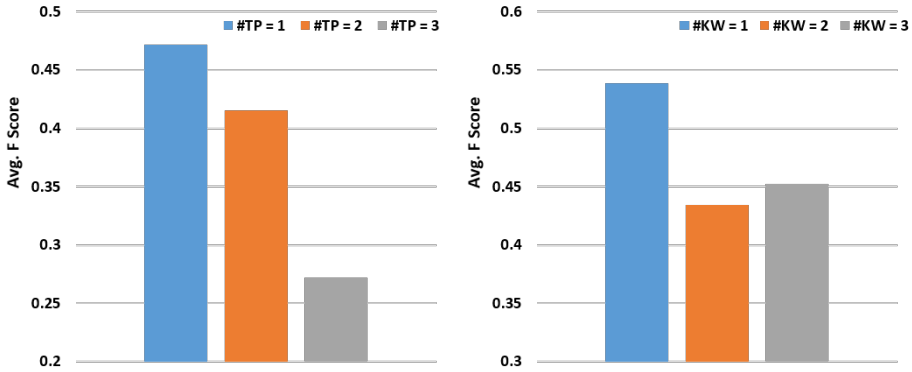


Fig. 2: Difficulties of QALD-6 categories.

2.3 Effect of Number of Triple Patterns and Keywords

Along with questions in natural languages, the QALD challenge also provide corresponding SPARQL queries to answer every questions. In addition, they also provide the exact keywords or named entities in each question. We want

[6] `https://qald.sebastianwalter.org/6/documents/qald-6_results.pdf`



(a) Effect of #triple patterns.

(b) Effect of #key words

Fig. 3: Effect of the number of triple patterns and number of keywords on the F scores.

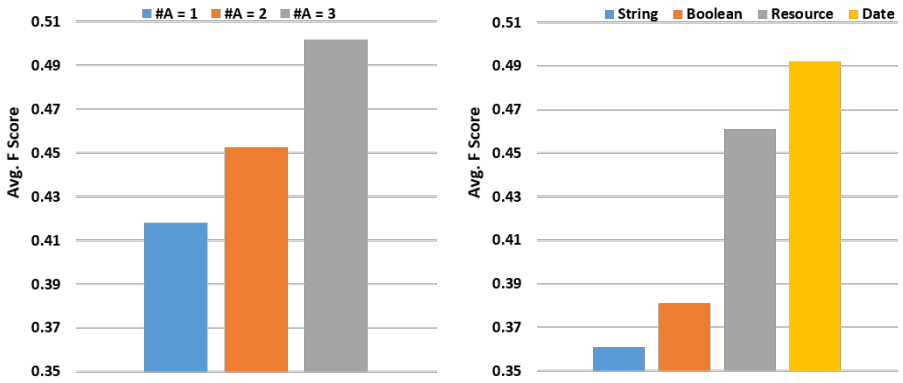
to investigate the effect of both of these features on the average F scores of the QA systems. The result in Figure 3a shows that the number of triple patterns in the SPARQL query has an inverse relationship with the average F score. If the number of triple patterns in the query is only 1 then the average F score is 0.47 which is dropped to 0.41 with number of triple patterns equal to 2, which is further dropped to only 0.27 with number of triple patterns equal to 3. The reason for this behaviour is that the complexity of the SPARQL increases with the triple patterns. Consequently, the more triple patterns the harder are the questions.

Figure 3b shows the effect of the number of keywords on the average F score of the systems. When the number of keywords is 1 the average F score is 0.53 which is reduced to 0.43 with number of keywords = 2. However, it is again increased to 0.45 when number of keywords = 3. This shows that the number of keywords does not have a significant impact on the average F scores. However, this point needs further investigation. This result could be different when applied to another QALD versions, i.e., the upcoming QALD-7 and QALD-8.

2.4 Effect of Answer Size and Answer Types

QALD questions can have exactly one answer or a set of answers. Moreover, the answers can be of four types: 1) String, 2) Boolean, 3) RDF Resource(s), i.e., an Http URI, and 4) Date. In this section, we want to investigate the effect of these two features, i.e. the size of the answer set as well as the type of the answer set, on the average F scores of the QA systems. The result in Figure 4a shows that the number of answers has a direct relationship to the average F score. When the number of answers = 1 the average F score = 0.41, when the number of answers = 2 the average F score = 0.45, and when the number of answers = 3 the average

F score = 0.50. The reason for this behaviour is that if the number of answers are more than one then it is possible for a given QA system to correctly identify some of the answers if not all exactly. Thus, the given system still can achieve high F scores with only a partial answer set.



(a) Effect of #answers

(b) Effect of answer type

Fig. 4: Effect of the number of answers and the answer types.

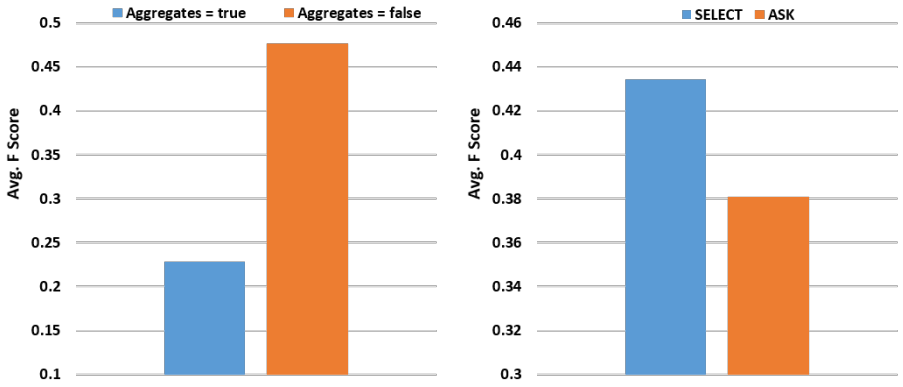
Figure 4b shows that the Date type answers are relatively easy to be correctly answered. The average F score for Date type answers is 0.49 which is followed by Resource type answer with average F score = 0.46, which in turn is followed by Boolean type answers with average F score = 0.38 and finally the String type answers are the most difficult to answer with an average F score = 0.36.

2.5 Effect of Aggregates and SPARQL forms

Finally, we want to investigate the effect of aggregate functions and the SPARQL query forms on the overall F scores of the state-of-the-art QA systems. For each question, QALD-6 provides information whether any aggregate function is required in the corresponding SPARQL query to correctly answer the given question. In addition, SPARQL has four query forms[7] namely SELECT, ASK, DESCRIBE, and CONSTRUCT. QALD challenges make use of the SELECT and ASK query forms.

Figure 5a shows that aggregates are much harder to answer. We have an average F score = 0.22 when aggregates is true and average F score = 0.47 when no aggregates are used. Surprisingly, the ASK (avg. F score = 0.38) queries are much harder to answer correctly compared to SELECT (avg. F score = 0.38) queries. Note that the answer of ASK queries is a Boolean value. Thus, this result is related to results presented in Figure 4b.

[7] SPARQL query forms: `https://www.w3.org/TR/rdf-sparql-query/#QueryForms`



(a) Effect of Aggregates (b) Effect of SPARQL forms

Fig. 5: Effect of the aggregates and SPARQL query forms on the F scores.

3 Conclusion

In this paper, we presented a fine-grained analysis of the QALD-6 challenge. We divided the QALD-6 questions into 8 different categories. We then compared the existing QA systems over Linked Data across each of these 8 categories. We proved that there is no sole winner across all of the categories. It turned out that questions of category *"When"* and *"In Which"* are relatively easy as compared to the categories *"Give Me"* and *"How Many"*. We showed that the number of triple patterns has an inverse relationship with average F scores of the QA systems. In addition, it was shown that the number of keywords does not significantly affect the overall F scores of QA systems. Yet, the number of answers has a direct relation with the average F scores of the systems. Date type questions are easier than questions whose answer is of type String. Aggregates are much harder to be correctly handled by most of the QA systems. Finally, ASK query forms are harder than SELECT query forms w.r.t. to the average F scores.

In the future, we want to do the same analysis for all of the QALD challenges. We will then present the combined results of all the challenges. We believe this will lead us to more concrete and stable results.

Acknowledgments

This work has been supported by the H2020 project HOBBIT (GA no. 688227) as well as the EuroStars projects DIESEL (no. 01QE1512C) and QAMEL (no. 01QE1549C). This work has also been supported by the German Federal Ministry of Transport and Digital Infrastructure (BMVI) in the projects LIMBO (no. 19F2029I) and OPAL (no. 19F2028A) as well as by the German Federal Ministry of Education and Research (BMBF) within 'KMU-innovativ: Forschung für die

zivile Sicherheit' in particular 'Forschung für die zivile Sicherheit' and the project SOLIDE (no. 13N14456).

References

1. P. Baudis and J. Sedivý. Modeling of the question answering task in the yodaqa system. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 6th International Conference of the CLEF Association, CLEF 2015, Toulouse, France, September 8-11, 2015, Proceedings*, pages 222–228, 2015.
2. A. Freitas, J. G. Oliveira, E. Curry, S. O'Riain, and J. C. P. da Silva. Treo: combining entity-search, spreading activation and semantic relatedness for querying linked data. In *1st Workshop on Question Answering over Linked Data (QALD-1)*, 2011.
3. V. Lopez, M. Fernández, E. Motta, and N. Stieler. PowerAqua: Supporting users in querying and exploring the Semantic Web. *Semantic Web Journal*, 3:249–265, 2012.
4. S. Shekarpour, E. Marx, A.-C. N. Ngomo, and S. Auer. Sina: Semantic interpretation of user queries for question answering on interlinked data. *Journal of Web Semantics*, 2014.
5. C. Unger, L. Bühmann, J. Lehmann, A. N. Ngomo, D. Gerber, and P. Cimiano. Template-based question answering over RDF data. In *21st WWW conference*, pages 639–648, 2012.
6. C. Unger, C. Forascu, V. Lopez, A. N. Ngomo, E. Cabrio, P. Cimiano, and S. Walter. Question answering over linked data (QALD-4). In *CLEF*, pages 1172–1180, 2014.
7. C. Unger, C. Forascu, V. Lopez, A. N. Ngomo, E. Cabrio, P. Cimiano, and S. Walter. Question answering over linked data (QALD-5). In *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.*, 2015.
8. C. Unger, A. Ngonga, and E. Cabrio. 6th open challenge on question answering over linked data (qald-6). In *The Semantic Web: ESWC 2016 Challenges.*, 2016.
9. R. Usbeck, A.-C. Ngomo, L. Bühmann, and C. Unger. Hawk – hybrid question answering using linked data. In *The Semantic Web. Latest Advances and New Domains*, volume 9088 of *Lecture Notes in Computer Science*, pages 353–368. Springer International Publishing, 2015.
10. K. Xu, Y. Feng, S. Huang, and D. Zhao. Question answering via phrasal semantic parsing. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 6th International Conference of the CLEF Association, CLEF 2015, Toulouse, France, September 8-11, 2015, Proceedings*, pages 414–426, 2015.