# Crowdsourced Linked Data Question Answering with AQUACOLD

Nicholas Collis
University of Bedfordshire, Luton, England
nwcphd@gmail.com

Ingo Frommholz
University of Wolverhampton, UK
ifrommholz@acm.org

***Abstract*—There is a need for Question Answering (QA) to return accurate answers to complex natural language questions over Linked Data, improving the accessibility of Linked Data (LD) search by abstracting the complexity of SPARQL whilst retaining its expressiveness. This work presents AQUACOLD, a LD QA system which harnesses the power of crowdsourcing to meet this need.**

***Index Terms*—Linked Data, Natural Language, Question Answering, Crowdsourcing, SPARQL**

## I. Introduction

The field of Question Answering (QA) focuses on providing specific answers to natural language questions posed by users [1]. Supplying direct answers is seen as preferable for end users compared to returning links to web pages that may contain the correct answer(s) [2]. Contemporary QA systems including Google, Siri and Alexa can provide answers to basic questions such as *'Who is in the current Manchester United squad?'*, however, simple queries such as this form the minority of web searches, with over 97% of questions answered 10 times or less [3]. QA systems are often unable to answer more complex questions such as *'Which Manchester United goalkeepers were born in Italy'* and instead display weblinks to information sources which may or may not include the correct answer. This highlights the need for a tool that can return accurate answers for more complex queries, which would benefit Digital Libraries by providing direct answers from a range of interconnected Linked Data sources. To surmount the difficulties inherent in parsing unstructured text, several question answering tools reason over Linked Data instead of (or together with) unstructured text to locate the correct answer(s). Linked Data refers to a technology stack (RDF, OWL, SPARQL) and set of defined guidelines that provides a mechanism for data to be published, queried and inferred on the web, forming a framework of interlinked nodes that can produce higher quality results for *factoid* questions (*who, where, which, when, what, is*) than unstructured text [4].

Question Answering systems powered by Linked Data often transform Natural Language questions into SPARQL, a structured query language designed for querying Linked Data. Recent examples include QAnswer[1], PowerAqua[2] and gAnswer[3]. Such techniques are successful in providing answers to questions of high complexity (depth) over a narrow domain (breadth) or shallow depth over a broad domain, but success typically decreases as question depth and breadth increases. This is largely due the difficulties inherent in producing a structured data representation for every type of question [5]. Employing crowdsourcing to translate natural language into a structured query language such as SPARQL has been shown to produce a Question Answering space with more breadth and depth than is achievable through purely programmatic methods [6]. However, this has often required paid microtask workers to translate natural language questions to their Linked Data representation, which has been shown to produce biased results and unaligned incentive structures with minimal scalability [7]. The proposed system, AQUACOLD (Aggregated Query-Understanding And Construction Over Linked Data), is aimed at empowering domain experts or users to provide support to their respective community by labeling and curating their queries and make them reusable for other users with similar information needs. To this end, AQUACOLD is a novel QA system that enables users to create Linked Data templates that can be used to produce answers for and get answer to multiple related natural language questions, through a familiar spreadsheet inspired user interface for filtering and creating sets of Linked Data, which can then be labelled with natural language, turned into generalised templates for answering related questions, which can be provided in response to a natural language question. This allows a wide range of complex questions to be asked and answered by the expert crowd with organically aligned incentives for producing generalised templates that arise from genuine information needs.

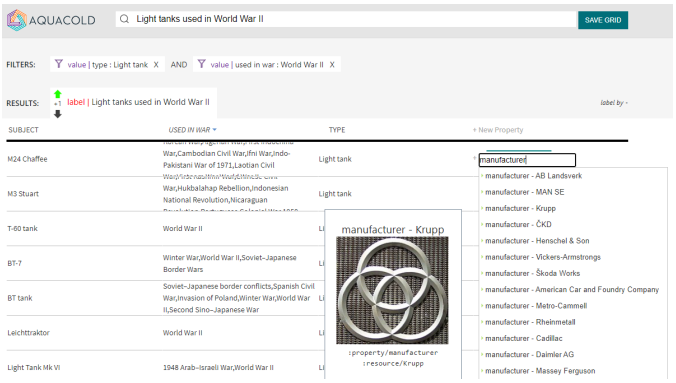


Fig. 1. The AQUACOLD interface

[1]https://www.qanswer.eu
[2]http://technologies.kmi.open.ac.uk/poweraqua
[3]http://ganswer.gstore.cn

## II. AQUACOLD System Overview

AQUACOLD combines elements of Natural Language search, Crowdsourcing and structured Query Builders into a QA tool for the Linked Data web which requires no prior experience of SPARQL or knowledge of the underlying schema.

### A. AQUACOLD Interface

The user interface of the AQUACOLD prototype is shown in Fig. 1. Key elements include: a combined search and labelling input box (at the top of the diagram); a Linked Data Query Builder interface which can explore a given data source, link to other nodes connected by a shared property and progressively build a set of results in response to the filters selected by users; autocompletion (seen in the drop down in the right of the diagram) to show what nodes in the Linked Data source contain the text entered so far, together with additional information including a description and picture. Voting (the green arrows on the diagram) is used to rank results, so those with the highest user vote score appear at the top of the search. Generalised query templates are created based on named entities shared by the question text and filter labels. These templates are employed to answer related queries from subsequent users.

### B. The Linked Data Feedback Loop

The following example illustrates how users can engage with AQUACOLD to find answers to their questions (Fig. 2) and feed back their solutions to subsequent users.

1) **User A** opens AQUACOLD looking for information on first world war tanks. They enter *'Tanks used in World War 1'* into the query input box. No matches are found.
2) Without any results to view, **User A** adds the filters `dbp:type=:Tank` and `dbp:usedInWar=:worldWar1`. This updates the grid with the relevant information.
3) With the grid complete, **User A** labels the results grid as *'Tanks used in World War 1'*.
4) AQUACOLD stores **User A's** label and the associated filters, including details of which filter labels match entities in the text, to be used as templates e.g. *Tanks used in [*]*.
5) The next user, **User B** queries AQUACOLD for *'Tanks used in World War 2'*. Although a result grid and filters for this query have not been explicitly created by a previous user, The template created from the previous step is used to generate the results, substituting the filter *'World War 1'* with *'World War 2'*.

The demonstration of the AQUACOLD prototype will include an overview of the system and provide an example of the Linked Data Feedback Loop.[4]

[4]https://tinyurl.com/aquacolddemo

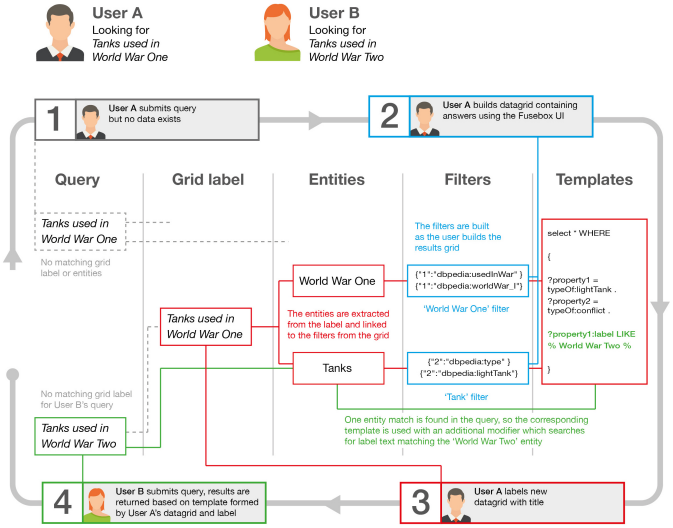


Fig. 2. The AQUACOLD *Linked Data Feedback Loop*

## References


[1] H.-J. Oh, K.-Y. Sung, M.-G. Jang, and S.-H. Myaeng, "Compositional question answering: A divide and conquer approach," *Information Processing & Management*, vol. 47, pp. 808–824, 2011.

[2] M. S. Bernstein, J. Teevan, S. Dumais, D. Liebling, and E. Horvitz, "Direct answers for search queries in the long tail," *Proceedings of the 2012 ACM annual conference*, pp. 237–246, 2012.

[3] R. W. White, M. Bilenko, and S. Cucerzan, "Studying the Use of Popular Destinations to Enhance Web Search Interaction," *Proceedings SIGIR 2007*, pp. 159–166, 2007.

[4] A. Bozzon, M. Brambilla, and S. Ceri, "Liquid Query : Multi-Domain Exploratory Search on the Web," *Proceedings of the 19th International Conference on World Wide Web*, pp. 161–170, 2010.

[5] K. Höffner and J. Lehmann, "Survey on Challenges of Question Answering in the Semantic Web," *Semantic Web (2017)*, vol. 8, no. November, pp. 895–920, 2016.

[6] H.-j. D. C.-y. Wu and R. T.-h. Tsai, "From Entity Recognition to Entity Linking : A Survey of Advanced Entity Linking Techniques," *The 26th Annual Conference of the Japanese Society for Artitifical Intelligence*, pp. 1–10, 2012.

[7] D. Damljanovic, J. Petrak, M. Lupu, H. Cunningham, M. Carlsson, G. Engstrom, and B. Andersson, "Random Indexing for Finding Similar Nodes within Large RDF graphs," *CEUR Workshop Proceedings*, no. 737, pp. 36–50, 2011.