The 6th Open Challenge on
Question Answering Over Linked Data

QALD-6

Document Version: January 15, 2016

QALD-6 is the sixth in a series of evaluation campaigns on question answering over linked data, with a strong emphasis on multilinguality, hybrid approaches using information from both structured and unstructured data, and question answering over RDF data cubes. QALD-6 is organized as a challenge at ESWC 2016 (http://2016.eswc-conferences.org). It is aimed at all kinds of systems that mediate between a user, expressing his or her information need in natural language, and semantic data. All relevant information for participating in the challenge are given in this document.

1 Introduction

Motivation and Goal

The past years have seen a growing amount of research on question answering over Semantic Web data, shaping an interaction paradigm that allows end users to profit from the expressive power of Semantic Web standards while at the same time hiding their complexity behind an intuitive and easy-to-use interface. The Question Answering over Linked Data challenge provides an up-to-date benchmark for assessing and comparing systems that mediate between a user, expressing his or her information need in natural language, and RDF data. It thus targets all researchers and practitioners working on querying linked data, natural language processing for question answering, multilingual information retrieval and related topics.

The key challenge for question answering over linked data is to translate a user's information need into a form such that it can be evaluated using standard Semantic Web query processing and inferencing techniques. In order to focus on specific aspects and involved challenges, QALD comprises three tasks: multilingual question answering over DBpedia, hybrid question answering over both RDF and free text data, and question answering over RDF data cubes. The main goal is to gain insights into the strengths and shortcomings of different approaches and into possible solutions for coping with the heterogeneous and distributed nature of Semantic Web data.

Organization

- *Elena Cabrio*, University of Nice Sophia Antipolis, France
- *Axel-Cyrille Ngonga Ngomo*, University of Leipzig, Germany
- *Christina Unger*, CITEC, Bielefeld University, Germany

In addition, the following organization committee members support the construction of the benchmark data and question sets, as well as the dissemination of the challenge.

- *Philipp Cimiano*, CITEC, Bielefeld University, Germany
- *Hady Elsahar*, Laboratoire Hubert Curien, Saint-Etienne, France
- *Corina Forascu*, Alexandru Ioan Cuza University, Iasi, Romania
- *Konrad Höffner*, AKSW, University of Leipzig, Germany
- *Vanessa Lopez*, IBM Research, Dublin, Ireland
- *Ricardo Usbeck*, AKSW, University of Leipzig, Germany
- *Amir Veyseh*, University of Tehran, Iran
- *Sebastian Walter*, CITEC, Bielefeld University, Germany

2 Relevant information in a nutshell

Challenge website: http://www.sc.cit-ec.uni-bielefeld.de/qald/ > QALD-6

ESWC 2016: http://2016.eswc-conferences.org

Task 1: Multilingual question answering over RDF data

- *Dataset:* DBpedia 2015
  http://downloads.dbpedia.org/2015-10/
- *Endpoint:* http://dbpedia.org/sparql
- *Training data:* 350 annotated questions in eight languages (English, Spanish, Italian, German, French, Dutch, Romanian, Farsi)
  http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-multilingual.json
- *Test data:* to be published on April 8, 2016

Task 2: Hybrid question answering over RDF and free text data

- *Dataset:* DBpedia 2015 with free text abstracts
  http://downloads.dbpedia.org/2015-10/

  Optional: English Wikipedia
  https://dumps.wikimedia.org
- *Training data:* 50 annotated questions in English
  http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-hybrid.json
- *Test data:* to be published on April 8, 2016

Task 3: Statistical question answering over RDF data cubes

- *Dataset:* LinkedSpending
  http://linkedspending.aksw.org
- *Endpoint:* http://linkedspending.aksw.org/sparql
- *Training data:* 100 annotated questions in English
  http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-datacube.json
- *Test data:* to be published on April 8, 2016

Schedule

Release of training data and instructions:	January 15, 2016
Paper submission deadline:	March 11, 2016
Release of test data:	April 8, 2016
Deadline for submission of system answers:	April 15, 2016
Release of evaluation results:	April 18, 2016
Submission of camera-ready papers:	April 24, 2016

Submission and evaluation

Submission of results and evaluation is done by means of an online form:
http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/index.php?x=evaltool&q=6

Results for the training phase can be uploaded at any time; results for the test phase can be uploaded from April 8 to May 6.

Resources

You are free to use any external resources. A list of potentially useful resources and tools is provided in Section 6.

Contact

Updates on the open challenge will be published on the *Interacting with Linked Data* mailing list:

https://lists.techfak.uni-bielefeld.de/cit-ec/mailman/listinfo/ild

In case of question, problems and comments, please contact Christina Unger:
cunger@cit-ec.uni-bielefeld.de

3 Description of the tasks

The main task of QALD is to retrieve answers to natural language questions or keywords from a given RDF dataset. In order to focus on specific aspects and challenges involved, QALD-6 comprises three tasks.

3.1 Task 1: Multilingual question answering over RDF data

Given the diversity of languages used on the web, there is an impeding need to facilitate multilingual access to semantic data. The core task of QALD is thus to retrieve answers from an RDF data repository given an information need expressed in a variety of natural languages.

The underlying RDF dataset is DBpedia, a community effort to extract structured information from Wikipedia and to make this information available as RDF data. The RDF dataset relevant for the challenge is the official DBpedia 2015 dataset for English, including links, most importantly to YAGO[1] categories. This dataset comprises the following files:

- http://downloads.dbpedia.org/2015-10/core-i18n/en/
  - `infobox_property_definitions_en.ttl`
  - `infobox_properties_unredirected_en.ttl`
  - `instance_types_en.ttl`
  - `instance_types_transitive_en.ttl`
  - `labels_en.ttl`
  - `mappingbased_literals_en.ttl`
  - `mappingbased_properties_en.ttl`
  - `specific_mappingbased_properties_en.ttl`
  - `persondata_en.ttl`
  - `interlanguage_links_en.ttl` (if you work on a language other than English)
- Ontology: http://wiki.dbpedia.org/services-resources/ontology

In order to work with the dataset, you can either load this data into your favorite triple store, or access it via a SPARQL endpoint. The official DBpedia SPARQL endpoint can be accessed at http://dbpedia.org/sparql/.

The training data consists of the 350 questions compiled from previous instantiations of the challenge. The questions are available in eight different languages: English, Spanish, German, Italian, French, Dutch, Romanian, and Farsi. Those questions are general, open-domain factual questions, such as *Which book has the most pages?*

The questions vary with respect to their complexity, including questions with counts (e.g., *How many children does Eddie Murphy have?*), superlatives (e.g., *Which museum in New York has the most visitors?*), comparatives (e.g., *Is Lake Baikal bigger than the Great Bear Lake?*), and temporal aggregators (e.g., *How many companies were founded in the same year as Google?*). Each question is annotated with a manually specified SPARQL query and answers.

As an additional challenge, a few of the training and test questions are out of scope, i.e. they cannot be answered with respect to DBpedia. The query is specified as `OUT OF SCOPE` and the answer set is empty.

The test dataset will consist of 100 similar questions compiled from existing question and query logs, in order to provide unbiased questions expressing real-world information needs.

[1] For detailed information on the YAGO class hierarchy, please see http://www.mpi-inf.mpg.de/yago-naga/yago/.

Training data

http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-multilingual.json

Test data

to be published on April 8, 2016

3.2 Task 2: Hybrid question answering over RDF and free text data

A lot of information is still available only in textual form, both on the web and in the form of labels and abstracts in linked data sources. Therefore, approaches are needed that can not only deal with the specific character of structured data but also with finding information in several sources, processing both structured and unstructured information, and combining such gathered information into one answer.

QALD therefore includes a task on hybrid question answering, asking systems to retrieve answers for questions that required the integration of data both from RDF and from textual sources. The task builds on DBpedia 2015 as RDF knowledge base, together with free text abstracts contained in DBpedia as well as, optionally, the English Wikipedia as textual data source. The relevant DBpedia files are the following:

- http://downloads.dbpedia.org/2015-10/core-i18n/en/
  - infobox_property_definitions_en.ttl
  - infobox_properties_unredirected_en.ttl
  - instance_types_en.ttl
  - instance_types_transitive_en.ttl
  - labels_en.ttl
  - mappingbased_literals_en.ttl
  - mappingbased_properties_en.ttl
  - specific_mappingbased_properties_en.ttl
  - persondata_en.ttl
  - long_abstracts_en.ttl.bz2
- Ontology: http://wiki.dbpedia.org/services-resources/ontology

A dump of the English Wikipedia can be found at https://dumps.wikimedia.org.

As training data, we build on the 50 English questions compiled for last year's challenge (partly based on questions used in the INEX Linked Data track[2]). The questions are annotated with answers as well as a pseudo query that indicates which information can be obtained from RDF data and which from free text. The pseudo query is like an RDF query but can contain free text as subject, property, or object of a triple. As test questions, we will provide 50 similar questions.

An example is the question *Who is the front man of the band that wrote Coffee & TV?*, with the following corresponding pseudo query containing three triples, two RDF triples and a triple containing free text as property and object:

```
SELECT DISTINCT ?uri
WHERE {
    <http://dbpedia.org/resource/Coffee_&_TV>
```

[2] http://inex.mmci.uni-saarland.de/tracks/dc/index.html

```
    <http://dbpedia.org/ontology/musicalArtist> ?x .
    ?x <http://dbpedia.org/ontology/bandMember> ?uri .
    ?uri text:"is" text:"frontman" .
}
```

And the manually specified answer is <http://dbpedia.org/resource/Damon_Albarn>.

One way to answer the question is to first retrieve the band members of the musical artist associated with the song Coffee & TV from the RDF data using the first two triples, and then check the abstract of the returned URIs for the information whether they are the frontman of the band. In this case, the abstract of Damon Albarn contains the following sentence:

> He is best known for being the frontman of the Britpop/alternative rock band Blur [...]

All queries are designed in a way that they require both RDF data and free text to be answered. The main goal is not to take into account the vast amount of data available and problems arising from noisy, duplicate and conflicting information, but rather to enable a controlled and fair evaluation, given that hybrid question answering is a still very young line of research.

Note that the pseudo queries only provide one possible way of answering the question, and they might still require some form of textual similarity and entailment. Also note that the pseudo queries cannot be evaluated against the SPARQL endpoint.

Training data

http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-hybrid.json

Test data

to be published on April 8, 2016

3.3 Task 3: Statistical question answering over RDF data cubes

With this new task, QALD aims to stimulate the development of approaches that can handle multi-dimensional, statistical data modelled using the RDF data cube vocabulary. The provided data consists of a selection of 50 data cubes from the LinkedSpending[3] government spending knowledge base. It can be downloaded from http://linkedspending.aksw.org/extensions/page/page/export/qbench2datasets.zip or accessed through the endpoint at http://linkedspending.aksw.org/sparql.

Question answering over RDF data cubes poses challenges that are different from general, open-domain question answering as represented by the above two tasks, such as the different data structure, explicit or implied aggregations and intervals. As an example consider the question *How much did the Philippines receive in 2007?* with the following SPARQL query:

```
select SUM(xsd:decimal(?v1))
{
 ?o a qb:Observation.
 ?o :recipient-country :ph.
 ?o qb:dataSet :finland-aid.
 ?o :refYear ?v0.
 filter(year(?v0)=2007).
 ?o :finland-aid-amount ?v1.
}
```

[3] http://linkedspending.aksw.org

The example illustrates the following challenges:

- *Implied aggregation:* The expected sum total is not directly mentioned.
- *Lexical gap:* The knowledge that an *amount* can be *received* is not given, so the measure *amount* can only be indirectly ascertained as the only measure with the correct answer type through the question word (*How much*).
- *Ambiguity:* Data cubes can contain large amounts of numerical values, impeding the conclusion that *2007* references a time period.
- *Data cube identification:* There is no reference to the description of the correct data cube (*Finland Aid Data*) but it can be matched to both the Phillipines and the year of 2007.

The training question set consists of the 100 question benchmark compiled in the CubeQA project,[4] annotated with SPARQL queries, answers and the correct data cube for each question. As test data, we will provide 50 additional questions, each answerable using a single data cube.

Training data

http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/6/data/qald-6-train-datacube.json

Test data

to be published on April 8, 2016

[4] http://aksw.org/Projects/CubeQA.html

4 Format of training and test data

Annotations are provided in JSON format. This format is currently developed as the standard format for all question answering benchmarks. Its general structure is as follows (red marking the obligatory fields):

```
{
  "dataset":
  {
    "id":
  },
  "questions": [
    {
      "id":
      "answertype":
      "aggregation":
      "onlydbo":
      "hybrid":
      "question": [
        {
          "language":
          "string":
          "keywords":
          "annotations": [

          ],
        }
      ],
      "query":
        {
          "SPARQL":
          "pseudo":
          "schemaless":
        },
      "answers": [

      ]
    }
  ]
}
```

The ID of the dataset in our case indicates the task and whether it is train or test (i.e. `qald-6-train-multilingual`, `qald-6-train-hybrid`, and `qald-6-train-datacube`).

Each of the questions specifies an ID together with the following attributes:

- `answertype` gives the expected answer type, which can be one the following:
  - `resource`: One or many resources, for which the URI is provided.
  - `string`: A string value such as `Valentina Tereshkova`.
  - `number`: A numerical value such as `47` or `1.8`.
  - `date`: A date provided in the format `YYYY-MM-DD`, e.g. `1983-11-02`.
  - `boolean`: Either `true` or `false`.
- `hybrid` specifies whether the question is a hybrid question, i.e. requires the use of both RDF and free text data
- `aggregation` indicates whether any operations beyond triple pattern matching are required to answer the question (e.g., counting, filters, ordering, etc.).

- `onlydbo` reports whether the query relies solely on concepts from the DBpedia ontology. If the value is `false`, the query might rely on the DBpedia property namespace (`http://dbpedia.org/property/`), FOAF or some YAGO category.

Note that for hybrid questions, the attributes `aggregation` and `onlydbo` refer to the RDF part of the query only.

Most importantly, for each question the dataset specifies the language, a question string and keywords, together with a corresponding query as well as the correct answer(s). The language is provided as an ISO 639-1 code, in our case: `en` (English), `de` (German), `es` (Spanish), `it` (Italian), `fr` (French), `nl` (Dutch), `ro` (Romanian), `fa` (Farsi). The query is in the case of tasks 1 and 3 a SPARQL query, in case of task 2 a pseudo query. Schemaless queries might be provided in the course of the training phase. The answers follow the JSON format of SPARQL results, cf. the W3C specification at http://www.w3.org/TR/sparql11-results-json/ and the next section.

In addition, systems can provide a double value $0 \leq d \leq 1$ as confidence measure, for example:

```
{
  "id": "42",
  "answers": [ ],
  "confidence": 0.9
}
```

The score for each question will be multiplied by the specified confidence value (or with `1.0` if none is specified).

5 Submission and evaluation

5.1 Submission

Results can be submitted from April 8 to April 15 via an online evaluation form:

http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/index.php?x=evaltool&q=6

This form is available both during training and test phase (note the drop down box that allows you to either specify `training` or `test`) and you can upload results as often as you like, for example trying different configurations of your system. During test phase, the upload with the best results will count. During training phase the evaluation tool will display the achieved results.

All submissions are required to comply with the JSON format specified in the previous section. For all questions, the dataset ID and question IDs are obligatory. Also you have to specify the answer(s) your system returned for that question. The structure of the answers has to comply with the SPARQL result standard: http://www.w3.org/TR/sparql11-results-json/. The bindings are obligatory, all other information can be provided or left out. That is, the minimal form an answer of a `SELECT` query should have is the following:

```
{ "results": {
    "bindings": [
        { "var": { "value": "42.0" }}
     ]
  }
}
```

The full answer could look as follows:

```
{ "head":
   { "link": [],
     "vars": ["var"]
   },
  "results": {
     "distinct": false,
     "ordered":  true,
     "bindings": [
        { "var":
            { "type": "typed-literal",
              "datatype": "http://www.w3.org/2001/XMLSchema#decimal",
              "value": "42.0"
            }
        }
     ]
  }
}
```

For `ASK` queries, the answer structure is the following (with `head` being optional):

```
  {
    "head": { },
    "boolean": true | false
  }
```

For questions that are out of the dataset's scope, the file should contain an empty answer set.

You are allowed to change the natural language question or keywords, for example by inserting quotes around named entities, by reformulating expressions your system struggles with, or even by using some controlled language format. If you do so, please document these changes, i.e. replace the provided question string or keywords by the input you used.

Also, it is preferred if your submission leaves out all question strings and keywords except for the ones your system worked on. Otherwise please mark the language in either the system name or configuration slot when uploading a submission. This way we can properly honour your multilingual efforts.

5.2 Evaluation measures

Participating systems will be evaluated with respect to precision and recall. Moreover, participants are encouraged to report performance, i.e. the average time their system takes to answer a query.

For each of the questions, your specified answers will be compared to the answers provided by the gold standard. The evaluation tool computes precision, recall and F-measure for every question q:[5]

$$\begin{aligned} \textit{Recall}(q) &= \frac{\text{number of correct system answers}}{\text{number of gold standard answers}} \\ \textit{Precision}(q) &= \frac{\text{number of correct system answers}}{\text{number of system answers}} \\ \textit{F-measure}(q) &= \frac{2 \times \textit{Precision} \times \textit{Recall}}{\textit{Precision} + \textit{Recall}} \end{aligned}$$

The tool then computes both macro and micro precision, recall and F-measure, both globally and locally, i.e. for

- all questions, and
- only those questions for which your system provides answers.

Therefore we recommend to include in your submission only questions that your system attempted to answer; otherwise the global and local scores will be the same.

[5] In the case of out-of-scope questions, an empty answer set counts as precision and recall 1, while a non-empty answer set counts as precision and recall 0.

6 Useful resources and tools

Prominent tools for indexing and searching datasets and text collections are:

- Lucene and Solr
  http://lucene.apache.org/
  http://lucene.apache.org/solr/
- Terrier
  http://terrier.org/

Building question answering systems is a complex task; it thus helps to exploit high-level tools for component integration as well as existing architectures for question answering systems:

- Apache UIMA
  http://uima.apache.org
- Open Advancement of Question Answering Systems (OAQA)
  http://oaqa.github.io
- Open Knowledgebase and Question Answering (OKBQA)
  http://www.okbqa.org
  https://github.com/okbqa
- openQA
  http://aksw.org/Projects/openQA.html

In the remainder of the section we provide a list of resources and tools that can be exploited especially for the linguistic analysis of a question and the matching of natural language expressions with vocabulary elements from a dataset.

Lexical resources

- WordNet
  http://wordnet.princeton.edu/
- Wiktionary
  http://www.wiktionary.org/
  API: https://www.mediawiki.org/wiki/API:Main_page
- FrameNet
  https://framenet.icsi.berkeley.edu/fndrupal/
- English lexicon for DBpedia 3.8 (in *lemon*[6] format)
  http://lemon-model.net/lexica/dbpedia_en/
- PATTY (collection of semantically-typed relational patterns)
  http://www.mpi-inf.mpg.de/yago-naga/patty/

Text processing

- GATE (General Architecture for Text Engineering)
  http://gate.ac.uk/
- NLTK (Natural Language Toolkit)
  http://nltk.org/

[6] http://lemon-model.net

- Stanford NLP
  http://www-nlp.stanford.edu/software/index.shtml
- LingPipe
  http://alias-i.com/lingpipe/index.html

Romanian:

- http://tutankhamon.racai.ro/webservices/TextProcessing.aspx

Dependency parser:

- MALT
  http://www.maltparser.org/
  Languages (pre-trained): English, French, Swedish
- Stanford parser
  http://nlp.stanford.edu/software/lex-parser.shtml
  Languages: English, German, Chinese, and others
- CHAOS
  http://art.uniroma2.it/external/chaosproject/
  Languages: English, Italian

Named Entity Recognition

- DBpedia Spotlight
  http://spotlight.dbpedia.org
- FOX (Federated Knowledge Extraction Framework)
  http://fox.aksw.org
- NERD (Named Entity Recognition and Disambiguation)
  http://nerd.eurecom.fr/
- Stanford Named Entity Recognizer
  http://nlp.stanford.edu/software/CRF-NER.shtml

String similarity and semantic relatedness

- Wikipedia Miner
  http://wikipedia-miner.cms.waikato.ac.nz/
- WS4J (Java API for several semantic relatedness algorithms)
  https://code.google.com/p/ws4j/
- SecondString (string matching)
  http://secondstring.sourceforge.net
- SimMetrics
  https://github.com/Simmetrics/simmetrics

Textual Entailment

- DIRT
  Paraphrase Collection: http://aclweb.org/aclwiki/index.php?title=DIRT_Paraphrase_Collection
  Demo: http://demo.patrickpantel.com/demos/lexsem/paraphrase.htm
- PPDB (The Paraphrase Database)
  http://www.cis.upenn.edu/~ccb/ppdb/

Translation systems

- English ↔ {Romanian,German,Spanish}
  http://www.racai.ro/tools/translation/racai-translation-system/

Language-specific resources and tools

Romanian:

- http://nlptools.info.uaic.ro/Resources.jsp

Anything missing?

If you know of a cool resource or tool that we forgot to include (especially for the challenge languages other than English), please drop us a note!