QUESTION ANSWERING ON RDF DATA CUBES

Der Fakultät für Mathematik und Informatik
der Universität Leipzig eingereichte

DISSERTATION

zur Erlangung des akademischen Grades

Doctor rerum naturalium
(Dr. rer. nat.)

im Fachgebiet Informatik vorgelegt von

Konrad Höffner, Diplom-Informatiker

geboren am 24.03.1983 in Leipzig, Deutschland

Leipzig, den 23. März 2020

AUTHOR:
Konrad Höffner

TITLE:
Question Answering on RDF Data Cubes

INSTITUTION:
Institut für Informatik
Fakultät für Mathematik und Informatik
Universität Leipzig

SUPERVISORS:
Prof. Dr. Klaus-Peter Fähnrich
Prof. Dr. Jens Lehmann

ABSTRACT

The Semantic Web, a Web of Data, is an extension of the World Wide Web (WWW), a Web of Documents. A large amount of such data is freely available as Linked Open Data (LOD) for many areas of knowledge, forming the LOD Cloud. While this data conforms to the Resource Description Framework (RDF) and can thus be processed by machines, users need to master a formal query language and learn a specific vocabulary. Semantic Question Answering (SQA) systems remove those access barriers by letting the user ask natural language questions that the systems translate into formal queries. Thus, the research area of SQA plays an important role for the acceptance and benefit of the Semantic Web.

The original contributions of this thesis to SQA are: First, we survey the current state of the art of SQA. We complement existing surveys by systematically identifying SQA publications in the chosen timeframe. 72 publications describing 62 different systems are systematically and manually selected using predefined inclusion and exclusion criteria out of 1960 candidates from the end of 2010 to July 2015. The survey identifies common challenges, structured solutions, and recommendations on research opportunities for future systems.

From that point on, we focus on multidimensional numerical data, which is immensely valuable as it influences decisions in health care, policy and finance, among others. With the growth of the open data movement, more and more of it is becoming freely available. A large amount of such data is included in the LOD cloud using the RDF Data Cube (RDC) vocabulary. However, consuming multidimensional numerical data requires experts and specialized tools.

Traditional SQA systems cannot process RDCs because their meta-structure is opaque to applications that expect facts to be encoded in single triples, This motivates our second contribution, the design and implementation of the first SQA algorithm on RDF Data Cubes. We kick-start this new research subfield by creating a user question corpus and a benchmark over multiple data sets. The evaluation of our system on the benchmark, which is included in the public Question Answering over Linked Data (QALD) challenge of 2016, shows the feasibility of the approach, but also highlights challenges, which we discuss in detail as a starting point for future work in the field.

The benchmark is based on our final contribution, the addition of 955 financial government spending data sets to the LOD cloud by transforming data sets of the OpenSpending project to RDF Data Cubes. Open spending data has the power to reduce corruption by increasing accountability and strengthens democracy because voters can make better informed decisions. An informed and trusting public also strengthens the government itself because it is more likely to commit to large projects. OpenSpending.org is an open platform that provides public finance data from governments around the world. The transformation result, called LinkedSpending, consists of more than five million planned and carried out financial transactions in 955 data sets from all over the world as Linked Open Data and is freely available and openly licensed.

PUBLICATIONS

This thesis is based on the following publications and proceedings.

INTERNATIONAL JOURNALS, PEER-REVIEWED

- **LinkedSpending: OpenSpending becomes Linked Open Data** [105]
  K Höffner, M Martin, and J Lehmann
  Semantic Web Journal, 7(1):95–104, 2016

- **Survey on Challenges of Question Answering in the Semantic Web** [108]
  K Höffner, S Walter, E Marx, R Usbeck, J Lehmann, and A.-C Ngonga Ngomo
  Semantic Web Journal, 8(6):895–920, 2017

CONFERENCES, PEER-REVIEWED

- **Towards Question Answering on Statistical Linked Data** [103]
  K Höffner and J Lehmann
  Proceedings of the 10th International Conference on Semantic Systems, 61–64, 2014

- **CubeQA—Question Answering on RDF Data Cubes** [104]
  K Höffner, J Lehmann, and R Usbeck
  The Semantic Web – ISWC 2016, 325–340, 2016

MISCELLANEOUS PUBLICATIONS

The following publications also originated before or during the writing of the thesis but are not part of it:

Question Answering

- **AskNow:**
  A Framework for Natural Language Query Formalization in SPARQL [65]
  M Dubey, S Dasgupta, A Sharma, K Höffner, and J Lehmann
  The Semantic Web. Latest Advances and New Domains, 300-316, 2016

- **DEQA: Deep Web Extraction for Question Answering** [123]
  J Lehmann, T Furche, G Grasso, A.-C Ngonga Ngomo, C Schallhart, A Sellers, C Unger, L Bühmann, D Gerber, K Höffner, D Liu, and S Auer
  The Semantic Web – ISWC 2012, 131–147, 2012

- **Keyword Query Expansion on Linked Data Using Linguistic and Semantic Features** [168]
  S Shekarpour, K Höffner, J Lehmann, and S Auer
  2013 IEEE Seventh International Conference on Semantic Computing, 191–197

- **Towards an Open Question Answering Architecture** [132]
  E Marx, R Usbeck, A.-C Ngonga Ngomo, K Höffner, J Lehmann, and S Auer
  Proceedings of the 10th International Conference on Semantic Systems, 2014

- **User Interface for a Template Based Question Answering System** [106]
  K Höffner, C Unger, L Bühmann, J Lehmann, A.-C. Ngonga Ngomo, D Gerber, and P Cimiano
  Knowledge Engineering and Semantic Web, 4th International Conference, 258–264, 2013

Statistical and Geographical Data

- **GeoKnow: Geo-Anwendungen im DatenWeb** [121]
  J Lehmann, K Höffner, S Prätor, S Lehmann, A.-C Ngonga Ngomo, A Garcia-Rojas, and S Athanasiou
  gis.Business, 5:48–51, 2013

- **LinkedGeoData: A Core for a Web of Spatial Open Data** [177]
  C Stadler, J Lehmann, K Höffner, and S Auer
  Semantic Web Journal, 3(4):333–354, 2012

- **Managing Geospatial Linked Data in the GeoKnow Project** [125]
  J Lehmann, S Athanasiou, A Both, A Garcia-Rojas, G Giannopoulos, D Hladky, K Höffner, J. J. L Grange, A.-C. Ngonga Ngomo, M. A Sherif, C Stadler, M Wauer, P Westphal, and V Zaslawski
  Studies on the Semantic Web, 51–78, 2015

- **The GeoKnow Handbook** [126]
  J Lehmann, S Athanasiou, A Both, L Buehmann, A Garcia-Rojas, G Giannopoulos, D Hladky, K Höffner, J. J. L Grange, A.-C. Ngonga Ngomo, R Pietzsch, R Isele, M. A Sherif, C Stadler, M Wauer, and P Westphal
  Technical Report, 2015

Interlinking

- **kOre: Using Linked Data for OpenScience Information Integration** [68]
  I Ermilov, K Höffner, J Lehmann, and D Mouromtsev
  SEMANTiCS 2015

- **RAVEN: Active Learning of Link Specifications** [149]
  A.-C Ngonga Ngomo, J Lehmann, S Auer, and K Höffner
  Proceedings of the 6th International Conference on Ontology Matching, 814:25–36, 2011

- **RAVEN—Towards Zero-Configuration Link Discovery** [150]
  A.-C Ngonga Ngomo, J Lehmann, S Auer, and K Höffner
  Technical Report, 2012

- **SAIM—One Step Closer to Zero-Configuration Link Discovery** [129]
  K Lyko, K Höffner, R Speck, A.-C Ngonga Ngomo, and J Lehmann
  The Semantic Web: ESWC 2013 Satellite Events, 167–172, 2013

Other

- **Technical Environment for Developing the SNIK Ontology of Information Management in Hospitals** [102]
  K Höffner, F Jahn, C Kücherer, B Paech, B Schneider, M Schöbel, S Stäubert and A Winter
  Studies in Health Technology and Informatics, 243:122–126, 2017

- **The SNIK Graph: Visualization of a Medical Informatics Ontology** [113]
  F Jahn, K Höffner, B Schneider, A Lörke, T Pause, E Ammenwerth and A Winter
  Studies in Health Technology and Informatics, 264:1941–1942, 2019

- **Open and Linkable Knowledge About Management of Health Information Systems** [112]
  K Höffner, F Jahn, A Lörke, T Pause, B Schneider, E Ammenwerth and A Winter
  Studies in Health Technology and Informatics, 264:1678–1679, 2019

ACKNOWLEDGMENTS

First of all, I want to thank my supervisors Prof. Klaus-Peter Fähnrich and Prof. Jens Lehmann as well as Prof. Sören Auer for inviting me to this PhD programme. I am saddened by the loss of Prof. Fähnrich so soon after his retirement. My condolences and best wishes go to his family. I give special thanks to my direct supervisor Prof. Jens Lehmann who has continuously supported me on this long journey with guidance on my research direction, feedback on how to be an effective researcher, cooperation on many papers and proofreading. I also thank my colleagues from the AKSW research group, especially: Edgard Marx for the many cooperations, trips, events and his cheerful and optimistic attitude. Amrapali Zaveri for showing me how to create a great systematic survey and for being always fun to be around. Claus Stadler for helping me on many occasions. Ivan Ermilov for always helping and for rescuing me from that airport with his Russian connections. Ricardo Usbeck for his hard work on our survey, for answering every little question and for being an example of efficient and professional research and communication. Mohamed Sherif for the advice on how to structure and format a thesis. Now go and see the pyramids, they are just down the street! Saeedeh Shekarpour for letting me present her project in California. Mohnish Dubey for showing me how to share my knowledge and for the boardgames. Lorenz Bühmann for letting me profit from his extensive technical experience. Muhammad Saleem for helping me in Japan. Nadine Jochimsen for lending me baby furniture.

I also thank my colleagues from the Institute for Medical Informatics, Statistics and Epidemiology (IMISE), especially: Prof. Alfred Winter for giving me the freedom to implement my many ideas while providing me with many ideas of his own and for always taking the time to review the results and to guide me back on track if I get carried away. Also for granting me a year-long parental leave to spend with my son Adrian. I thank Birgit Schneider and Franziska Jahn for all the fun discussions and the teamwork. Sebastian Stäubert for fulfilling my system administrative wishes at superhuman speed and preserving my sanity. Kathleen Becker and Anja Doil for helping with all the formalities.

I thank Simon Trümpler for being a longtime friend, best man at my wedding, and for being an example of how continued dedication to a single area leads to success. Robert Engsterhold for always being reliable and supportive and for proofreading the thesis. Thomas Pause for taking the time out of his crazy schedule to proofread and for his energy and motivation. The Free School Leipzig for really caring about the pupils and for harnessing my curiosity. I remember Michael I O VI IX (“Time, Essence, Universe, Nature”) Holz, legendary philosopher, English teacher and Abalone partner.

Last but not least I thank my grand family for rooting and supporting me. My wife Anja for prying my lost foreign study diploma grades from the hands of the administration in Montpellier, France, so that I could start this thesis in the first place and for taking much more than her share in caring for our children in the final stages of the thesis. My daughter Pauline and my son Adrian, who were both born during the writing of this thesis, for always bringing a smile on my face.

This work has been supported by the European Social Fund and the Free State of Saxony as well as the FP7 project GeoKnow (GA No. 318159). The thesis would not be possible without the many co-authors of the papers integrated in this thesis. Thus, as is common in the field, the we-form is used throughout this work, except for personal opinions.

CONTENTS

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF TABLES

ACRONYMS

AQE	Automatic Query Expansion
CSV	Comma-Separated Values
DCMI	Dublin Core Metadata Initiative
HMM	Hidden Markov Model
JSON	JavaScript Object Notation
LOD	Linked Open Data
NER	Named Entity Recognition
NL	Natural Language
NLP	Natural Language Processing
OWL	Web Ontology Language
POS	Part of Speech
QA	Question Answering
QALD	Question Answering over Linked Data
RDC	RDF Data Cube
RDCQA	RDF Data Cube Question Answering
RDF	Resource Description Framework
RDFS	RDF Schema
SDMX	Statistical Data and Metadata eXchange
SPARQL	SPARQL Protocol and RDF Query Language
SQA	Semantic Question Answering
tf-idf	term frequency—inverse document frequency
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WWW	World Wide Web
XSD	XML Schema

1 INTRODUCTION

The Semantic Web, a *Web of Data*, is an extension of the World Wide Web, a *Web of Documents*. A large amount of such data is freely available as *Linked Open Data* for many areas of knowledge, forming the *Linked Open Data (LOD) Cloud*. While this data, in the form of Resource Description Framework (RDF), can be processed by machines, users need mastery of a query language and knowledge of a specific vocabulary. Semantic Question Answering (SQA) systems remove those access barriers by allowing the user to ask natural language questions that the systems translate into queries. Similar to Web of Document keyword search, where a user gets a list of possibly relevant web documents, in Semantic Search [194], a user enters a list of keywords and gets a list of possible RDF resources. Keyword queries are not expressive enough for complex information needs, however. Instead, this work investigates Question Answering, where complete questions are posed. In the context of the Semantic Web, this technique is called Semantic Question Answering (SQA, see Section 2.2), which is an active research area with many different approaches, see Chapter 4. Domain independent SQA approaches are flexible enough to accept heterogenous data of many domains. However, they cannot process multidimensional, numerical data, which forms a large part of the Semantic Web in the form of RDF Data Cubes (RDCs), see Section 2.3.

In this thesis, we enable SQA to process RDF Data Cubes. To achieve this goal, we first survey the current state of the art on SQA. Next, we provide the first SQA algorithm that can process RDF Data Cubes. Finally, we describe our conversion of existing financial data cubes to the RDC vocabulary and use it to benchmark our algorithm as well as stimulate further research in this area.

1.1 MOTIVATION

M1: Lack of current SQA surveys

Natural language is complex and ambiguous. SQA systems rely on many different Natural Language Processing (NLP) techniques to capture the intended meaning of a question. A few of those NLP techniques, like parsing and Part of Speech (POS) tagging, can be solved using existing mature high-performance implementations. The others, however, still present difficult challenges. While the massive research effort has led to major advances, as shown by the yearly Question Answering over Linked Data (QALD) evaluation campaign, it suffers from several problems. Instead of a shared effort, many essential components are redeveloped, which is an inefficient use of researcher's time and resources. While shared practices emerge over time, they are not systematically collected. Furthermore, most systems focus on a specific challenge, while other challenges receive less effort, which leads to low overall benchmark scores and thus undervalues the contribution. Previous work [10, 41, 128, 139] has compared a large number of SQA algorithms but there is a partial lack of coverage from the end of 2010 onwards 2013 and an even greater lack of coverage from 2013 onwards, see Table 3.1. Due to the large number of SQA-related

publications[1], current surveys are required to get an accurate impression of the current state of the field, to prevent duplicated efforts and to identify new promising starting points for future research.

M2: Existing Semantic Question Answering approaches cannot process RDF Data Cubes

Generic SQA approaches cannot process RDCs for two reasons:

1. Syntactially, they do not recognize the structure of the RDC meta model.
2. Semantically, the answers need to be derived from observations, which can have many dimensions and whose values are meaningless without the proper context and further processing, such as aggregate functions.

M3: Large collections of data cubes that do not have an RDF Data Cube equivalent

There is a large number of published RDCs.[2] Still, there are large collections of freely available data cubes that are not available as RDF Data Cubes, such as government spending data. Transparency into government spending data has a high public demand. Further benefits include the power to reduce corruption by increasing accountability and strengthen democracy because voters can make better informed decisions. We identify the data sets of the OpenSpending project as suitable candidates for a conversion to RDF data cubes.

1.2 RESEARCH QUESTIONS AND CONTRIBUTIONS

For each motivation, we formulate a research question (RQ) and state our contribution towards it.

RQ1: What are the current approaches for Semantic Question Answering?

To address the lack of current SQA surveys, we conduct a systematic survey of SQA approaches in Chapter 4. SQA systems are manually and systematically selected using predefined inclusion and exclusion criteria, leading to 62 systems described in 72 publications analyzed in detail out of 1960 candidates. To help establish shared practices and guide future research, we add the following sub-questions, which lead us to identify common challenges, categorize solutions, and provide directions for future research:

RQ1.1: What are the challenges that SQA systems have to overcome?

RQ1.2: How are existing SQA systems built and how do they adress the challenges?

RQ1.3: Which aspects of the challenges are still unsolved and which plans exist to handle them in the future?

1 At least between 13 and 20 publications per year between 2011 and 2014, inclusive. See Table 4.3.
2 From the 3480 classes processed by LODStats [69] on 2017-02-10, `qb:Observation` from the RDF Data Cube vocabulary is the 12th most used class with 463 314 instances.

RQ2: How can SQA be applied to RDCs?

The corpus of user questions (Section 5.1, appendix A) is used to analyze typical numerical information needs. Rewriting the corpus questions to reference LinkedSpending (see Chapter 6) data sets yields the QALD6T3 benchmark (see appendix B) that allows the evaluation of the precision and recall of a system. We design (Chapter 5) and evaluate (Section 6.7) the domain independent CubeQA algorithm, which is the first RDF Data Cube Question Answering (RDCQA) system. Section 6.7.1 describes the experimental setup and shows that CubeQA achieves a global F_1 score of 0.43 on the QALD6T3-test benchmark, showing that RDCQA is feasible. Section 6.7.2 discusses limitations and frequent types of errors and quantitatively compares CubeQA to other RDCQA systems that were developed to take part in the public QALD6T3 challenge. Section 7.1 summarizes and answers the following sub-questions:

RQ2.1: What are typical multidimensional numerical user questions?

RQ2.2: Which information needs do those questions contain?

RQ2.3: How can an algorithm use data cube operations to satisfy the information needs of the questions?

RQ2.4: How can we evaluate the performance of a SQA system based on the user questions?

RQ2.5: Is CubeQA powerful enough to be practically useful on challenging questions?

RQ2.6: Is there a tendency towards either high precision or recall?

RQ2.7: What types of errors occur? How frequently are they? What are the reasons?

RQ2.8: How do other RDCQA systems perform?

RQ3: How can we transform a large amount of relevant data cubes to the RDF Data Cube vocabulary?

Our contribution is LinkedSpending, an RDC transformation of the OpenSpending[3] data sets, which provides government spending financial transactions from all over the world and is thus suitable as a core knowledge base that can be enriched and integrated with other, more specialized data sets. Transforming OpenSpending to Linked Data and publishing it adds to and profits from the Semantic Web, which offers benefits including a standardized interface, easier data integration and complex queries over multiple knowledge bases. Chapter 6 addresses the following sub-questions:

RQ3.1: Which collection of data cubes has a large size, a high significance to the general public and is not yet available as RDC?

RQ3.2: What is the typical way to transform data cubes to RDF?

RQ3.3: How can this typical way be adapted to the chosen collection?

RQ3.4: What are the results of the transformation and how can they be used?

3 http://openspending.org

1.3 THESIS STRUCTURE

This thesis consists of six chapters and an appendix:

- Chapter 1 introduces and motivates the research questions and summarizes the contributions.
- Chapter 2 defines the terms used throughout this work.
- Chapter 3 summarizes existing work related to the research questions.
- Chapter 4 is based on Höffner et al. [108] and analyzes the state of the art of Semantic Question Answering in the form of a systematic survey. We define a strict discovery methodology that consists of a multistep process to apply inclusion and exclusion criteria to find and filter surveyed publications. The survey is compared to older, similar surveys as well as evaluation campains. Each surveyed system is introduced and the challenges it faces are identified along with approaches to tackle them. The challenges are categorized by maturity and future developments are discussed.
- Chapter 5 is based on Höffner et al. [103] and Höffner et al. [104], and introduces CubeQA, the first algorithm for SQA on RDF Data Cubes.
- Chapter 6 is based on Höffner et al. [105] and presents LinkedSpending, a conversion of public governmental financial transactions from *OpenSpending* to Linked Open Data. We also present the RDCQA benchmark based on LinkedSpending that was published as task 3 of the 6th QALD evaluation campaign and use it to evaluate CubeQA and to compare it to other RDCQA approaches.
- Chapter 7 concludes the thesis, summarizes the contributions of the presented approaches and proposes future work.
- Appendix A contains the corpus of user questions for analyzing typical numerical information needs.
- Appendix B contains the QALD 6 Task 3 training and test benchmark data sets for evaluating the performance of RDCQA systems.

2 PRELIMINARIES

This work lies at the intersection of several research areas, whose core concepts are introduced in the following.

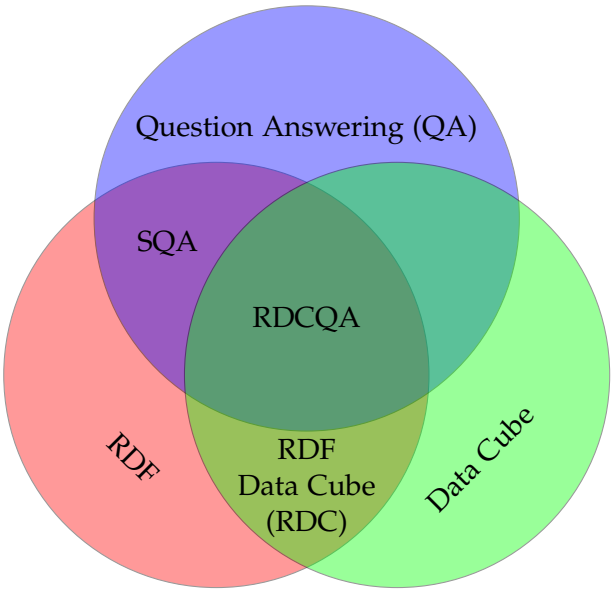

Figure 2.1: Research areas related to RDCQA and their overlap.

2.1 SEMANTIC WEB

The Semantic Web, or Web of Data, was proposed as an extension of the World Wide Web by Berners-Lee et al. [24] and consists of a huge amount of interlinked, machine-interpretable data. This provides the basis for complex information seeking tasks such as "Which hotels are near lakes with a water temperature of more than 20 °C in July?" The Web of Data is well suited to these tasks because it provides information that is interpretable by machines and semantically linked. The Semantic Web includes and extends, among others, the following standards and technologies:

2.1.1 *URIs and URLs*

A Uniform Resource Identifier (URI) is a sequence of characters that uniquely identifies an abstract or physical resource [22]. To save space, URIs can be abbreviated using prefixes, see Table 2.1. For example, `http://dbpedia.org/resource/Berlin` can be abbreviated to `dbr:Berlin`.

A Uniform Resource Locator (URL) is a URI that contains a resource locator, which describes a way to access that resource [25].

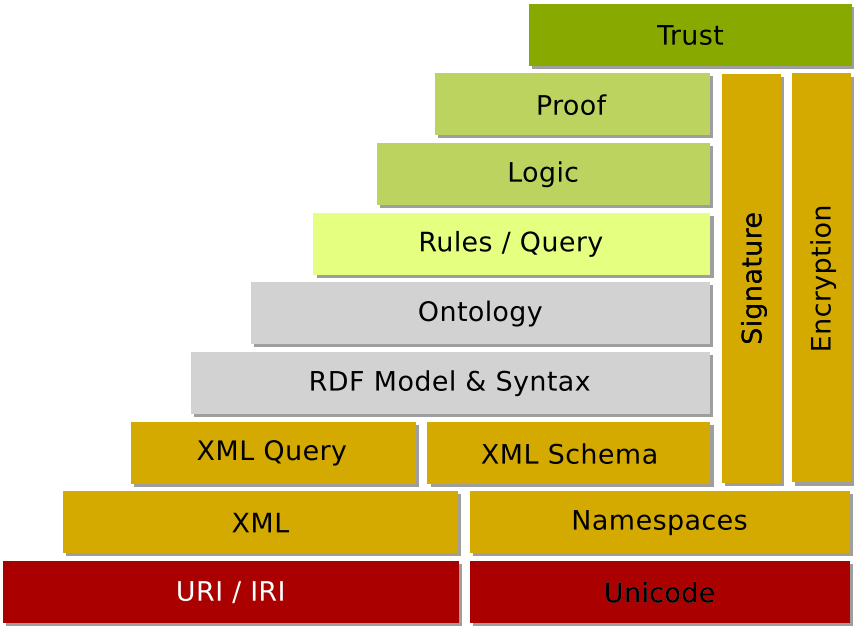

Figure 2.2: The Semantic Web Stack. Source: https://commons.wikimedia.org/wiki/File:W3c-semantic-web-layers.svg

2.1.2 *Linked Data*

While hypertext on the Web of Documents can contain hyperlinks to data, this data does not have hypertext capability itself. For example, you cannot link a row of a Comma-Separated Values (CSV) table to a row in another table. Another weakness of the World Wide Web (WWW) is the missing *semantics* of hyperlinks: they are untyped and thus do not provide machine-processable information about the kind of association a hyperlink represents. *Linked Data* [20] remedies those deficiencies using four rules:

1. Items of discourse are identified by URIs
2. Those URIs are also HTTP URLs, so that more information about a resource can be gained by *dereferencing* it using HTTP lookup.
3. Lookup results in information expressed using the standards RDF and SPARQL.
4. URIs are joined by typed links, so that related information can be discovered.

If Linked Data is published under an open license, it is called *Linked Open Data*.

LINKED OPEN DATA CLOUD All data sets that are publicly available as Linked Data under an open licence and that are connected with other data sets, are collectively called the Linked Open Data (LOD) cloud.

Prefix	URL
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
owl	http://www.w3.org/2002/07/owl#
xsd	http://www.w3.org/2001/XMLSchema#
dc	http://purl.org/dc/terms/
dbo	http://dbpedia.org/ontology/
dbr	http://dbpedia.org/resource/
dbp	http://dbpedia.org/property/
os	http://openspending.org/
ls	http://linkedspending.aksw.org/instance/
lso	http://linkedspending.aksw.org/ontology/
qb	http://purl.org/linked-data/cube#
sdmxd	http://purl.org/linked-data/sdmx/2009/dimension#

Table 2.1: URL prefixes used throughout this work.

2.1.3 *Resource Description Framework*

RDF is a set of specifications[1] to describe arbitrary facts about *resources* (URIs[2]) as *triples*, see Definition 1. A set of triples is called an *(RDF) graph* in the context of querying RDF. A set of triples that describes a certain domain is also be called an *(RDF) knowledge base* and may be the union of multiple RDF graphs. RDF can be serialized in several formats, including *N-Triples* and *Turtle*. A graph database for RDF is called a *triple store*.

Definition 1 (RDF triple) *An* (RDF) triple *represents a single fact and consists of a subject, property (also called a* predicate*) and object. All elements of a triple can be a resource, but the object can also be a literal. Subject and object can also be blank nodes, anonymous resources with no URI, but they are not used in this work. Formally: Let U be a set of URIs, $U = I \cup P \cup C$, where I are the instances, P the properties and C the classes. Let $L \subset \Sigma^*$ be a set of literals, where Σ is the unicode alphabet. We define an RDF triple t as $t \in U \times P \times (U \cup L)$.*

N-TRIPLES Each line in an N-Triples file contains a single triple. Depending on the type of object, a triple is serialized in one of four ways:

```
<subject> <predicate> <object-resource>.
<subject> <predicate> "object-untyped-literal".
```

1 see https://www.w3.org/standards/techs/rdf#w3c_all

2 Resources can also be IRIs, a superset of URIs with a wider range of allowed Unicode characters, but they are not used in this work.

```
<subject> <predicate> "object-untyped-literal"@languagetag.
<subject> <predicate> "object-typed-literal"^^datatype.
```

The city of Berlin is represented in DBpedia [124] as http://dbpedia.org/resource/Berlin. Dereferencing this URL and choosing the N-Triples format yields the following (excerpt, DBpedia version 2016-10):

```
<http://dbpedia.org/resource/Berlin>
    ↪<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
    ↪<http://dbpedia.org/ontology/City>.
<http://dbpedia.org/resource/Berlin>
    ↪<http://dbpedia.org/property/locatedIn>
    ↪<http://dbpedia.org/resource/Germany>.
<http://dbpedia.org/resource/Berlin>
    ↪<http://dbpedia.org/property/areaCode> "030".
<http://dbpedia.org/resource/Berlin>
    ↪<http://www.w3.org/2000/01/rdf-schema#label> "Berlin"@de.
<http://dbpedia.org/resource/Berlin>
    ↪<http://www.w3.org/2000/01/rdf-schema#label> "Berlijn"@nl.
<http://dbpedia.org/resource/Berlin>
    ↪<http://dbpedia.org/ontology/populationTotal>
    ↪"3610156"^^<http://www.w3.org/2001/XMLSchema#integer>.
```

The N-Triples format is easy to process in scripts because it is line-based but it is hard to read due to the verbosity.

TURTLE Turtle is a superset of N-Triples with the added possibility of abbreviating URIs using prefixes as well as grouping together repeated subjects and subject-predicate pairs. The city of Berlin from DBpedia [124] is represented in Turtle as follows (excerpt, version 2016-10):

```
@prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>
@prefix dbp:<http://dbpedia.org/property/>
@prefix dbo:<http://dbpedia.org/ontology/>
@prefix dbr:<http://dbpedia.org/resource/>
@prefix xsd:<http://www.w3.org/2001/XMLSchema#>

dbr:Berlin rdf:type           dbo:City;
           dbp:locatedIn      dbr:Germany;
           dbp:areaCode       "030";
           rdfs:label         "Berlin"@de, "Berlijn"@nl;
           dbp:populationTotal  "3610156"@xsd:integer.
```

From this point on, we use Turtle for all RDF listings because it is concise and easy to read. We also omit prefix declarations in Turtle code and refer to Table 2.1. A prefix can be the empty string; We use this for generic examples and in the context of statements locally to a single knowledge base.

2.1.4 *Ontologies*

In computer science[3], an "ontology" is an "explicit specification of a conceptualization" [90]. Ontologies are an active research area. A domain can be described in great detail using OWL [101], but in the context of RDCQA, ontologies are mostly used as a lightweight schema that provides context to a large amount of instance data. For example, DBpedia [124] version 2016-04 contains 760 classes and 5 044 222 instances. Table 2.2 defines basic terms [183] of ontologies and gives examplary RDF Turtle modelling statements.

A *class* is a set of resources and can be defined either intrinsically or extrinsically. Intrinsically, a class is defined by the properties of its members. For example, a motor boat is a boat that has at least one motor. A member of a class is called its *instance*. Depending on the type of ontology, such an instance my be a class or property itself. An instance that is neither a class nor a property is also called an *individual*. In the context of RDCQA, however, classes are extrinsically defined, that is, the class membership is explicitly mentioned.

$$\text{City} = \{\text{Berlin}, \text{Leipzig}, \ldots, \text{London}\}$$

In RDF, class membership is stated using the property `rdf:type`, with the instance as the subject and the class as the object.

```
dbr:Berlin rdf:type dbo:City.
```

Because `rdf:type` is such a commonly used property, Turtle defines the shorthand "a" for it:

```
dbr:Berlin a dbo:City.
```

As classes are sets, they can be subclasses (subsets) or superclasses (supersets) of other classes.

$$\text{MotorBoat} \subseteq \text{Boat} \subseteq \text{Vehicle}$$

Multiple classes can share the same instance and a class can be a subclass of multiple superclasses:

```
:myBoat a :MotorBoat, SailBoat.
:MotorBoat rdfs:subClassOf :Boat.
:Boat      rdfs:subClassOf :Vehicle.
```

Using the definition of a subset[4] and the transitivity[5]of the subset relation, superclass membership can be *inferred*[6], that is, it can be logically deduced. When answering questions like "Which vehicles are powered by a motor?", a motor boat can be found even if it is defined as an instance of *MotorBoat* and not *Vehicle*. By

3 The term "ontology" has a more abstract meaning in philosophy.
4 $A \subseteq B \Leftrightarrow \forall a \in A : a \in B$
5 $A \subseteq B \subseteq C \Rightarrow A \subseteq C$

Term	Definition	RDF Example
Class	Set of *instances*	`dbr:Berlin rdf:type dbo:City`.
Subclass	Subset of another class	`dbo:City rdfs:subClassOf dbo:Settlement`.
Property	Binary relation between resources	All resources in the middle position in the examples.
Literal	A value such as a string or number	`dbr:Berlin rdfs:label` "Berlin"@en.
		`dbr:Berlin dbo:areaCode` "030".

Table 2.2: Ontology terms and examples.

providing additional correct answers, inferrence increases the recall, see Definition 3.

The subclass relation may contain cycles, that is $C_1 \subseteq C_2 \subseteq \ldots \subseteq C_n \subseteq C_1$, but that is usually avoided as it implies the equivalence of all involved classes.

2.2 QUESTION ANSWERING

2.2.1 *History*

The problem of interfacing between humans and machines has a long history. Starting with binary code, those interfaces evolved over punch cards and assembly languages to high-level programming and query languages, which are more intuitive to human users [206]. Still, they are inaccessible to non-experts, who do not have knowledge of the formal language. Human languages such as English are usable by a larger audience and require less mental effort for human users. For computers however, natural language presents a complex problem, which has been researched by the field of Natural Language Processing (NLP) since the 1950s. Question Answering systems were developed as early as in the early 1960s, such as BASEBALL [89]. Another early Question Answering (QA) system is LUNAR [207], which answers questions about Apollo 11 moon rocks. Those early systems are domain specific, that is, they can only process the vocabulary and data of a certain fixed domain, such as moon rocks. In order to achieve a higher implementor productivity and to allow the usage of different algorithms and databases, NLP applications became increasingly split into modular, reusable components, although the approach of engineering such a system has to be chosen carefully in order to not lose efficiency over a custom-made system [99]. The use of multiple data sources enables open domain (also called "domain independent") QA, which is a much harder problem. A famous example of an open domain QA system is IBM Watson [142].

When Sir Tim Berners-Lee first published his vision for the Semantic Web [23] in 1999, "intelligent agents" played a central role. Ideally, those agents would be able

6 Inference requires support by the query engine, addition of the extra triples to the knowledge base or support by the SQA system.

to perform complex tasks that require interpreting human commands, integrating multiple services, optimizing and weighing implicitly and explicitly given criteria and, if necessary, performing actions in place of their client, such as reserving a table at a restaurant or booking a flight. This vision motivates open domain Semantic Question Answering systems, as they interpret the whole question and can thus be used for varied and complex tasks that other approaches, such as predefined templates, cannot handle.

2.2.2 *Definitions*

QUESTION ANSWERING (QA) We define QA as users asking questions in natural language (NL) using their own terminology to which they receive a concise answer.[7] For example, the question "What is the largest German city?" is answered with "Berlin".[8]

SEMANTIC QUESTION ANSWERING (SQA) We define SQA as QA on RDF data. This commonly requires querying an RDF knowledge base using SPARQL. We concentrate on factual questions, where the answer consists of a set of resources or literal values.

DOCUMENT RETRIEVAL In contrast to Question Answering, which returns answers to questions directly, document retrieval systems, such as internet search engines, return documents. Document retrieval systems usually split the retrieval process in three sequential steps:

1. In the *query processing* step, query analyzers identify documents in the data store.
2. Thereafter, the query is used to *retrieve documents* that match the query terms resulting from the query processing.
3. Finally, the retrieved documents are *ranked* according to some ranking function, commonly term frequency—inverse document frequency (tf-idf) [176].

2.2.3 *Evaluation*

Given C the correct set of resources and A the answers of the algorithm, we define the following performance measures:

Definition 2 (Precision) *The* precision *measures the proportion of answers that are correct. When there are no answers, we define the precision as* 0*.*

$$p = \frac{|C| \cap |A|}{|A|}$$

Definition 3 (Recall) *The* recall *measures the proportion of answers in the set of correct resources.*

$$r = \frac{|C| \cap |A|}{|C|}$$

7 Definition based on Hirschman et al. [100].
8 Interpreting "largest" as "with the largest area".

Definition 4 (F_1 score) *The F_1 score is the harmonic mean of precision and recall:*

$$F_1 = 2\frac{pr}{p+r}$$

When no correct answer is given ($p = r = 0$), we define $F_1 = 0$.

EXAMPLE "What are the original members of the Beatles?"

$$\begin{aligned}
A =& \Big\{\texttt{"Yoko Ono"},\texttt{"Paul McCartney"},\texttt{"John Lennon"},\\
&\texttt{"George Martin"}, \texttt{"Pete Best"}\Big\}\\
C =& \Big\{\texttt{"Ringo Starr"},\texttt{"Paul McCartney"},\texttt{"John Lennon"},\\
&\texttt{"George Harrison"}\Big\}\\
C \cap A =& \Big\{\texttt{"Paul McCartney"},\texttt{"John Lennon"}\Big\}\\
p =& \frac{2}{5} = 0.4\\
r =& \frac{2}{4} = 0.5\\
F_1 =& 2\frac{pr}{p+r} = 2 \cdot \frac{0.4 \cdot 0.5}{0.4 + 0.5} = 0.\overline{44}
\end{aligned}$$

AGGREGATION The average value of precision, recall and F_1-score can be calculated in multiple ways: The macro-average is the average of the individual measure results for each question, while the micro-average is attained by applying a measure once on the total numbers of correct answers and answers given by the system. Additionally, there is the global average, which includes all question and the local average, that only takes questions into account where the system returns at least one answer.

In this work, we report the global macro average precision as the arithmetic mean of the precision values:

$$\varnothing p = \frac{\Sigma_{i=1}^{n} p_n}{n}$$

The average recall and F_1 score is calculated analogously.

2.2.4 *SPARQL*

SPARQL Protocol and RDF Query Language (SPARQL) is the de facto standard language for querying RDF data. A simple SELECT query contains four main parts:[9]

PREFIXES URIs from the same knowledge base often share a significant amount of leading characters. This can be used to reduce verbosity by defining and using pre-

9 More features are described in the W3C recommendation Aranda et al. [8].

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>
SELECT ?city
WHERE
{
 ?city a               dbo:City;
       dbo:populationTotal  ?pop;
       dbo:country   dbr:Germany.
}
ORDER BY DESC(?pop)
LIMIT 1
```

Figure 2.3: Example of a SPARQL query on the public DBpedia SPARQL endpoint http://dbpedia.org/sparql. DBpedia [124] is an RDF transformation of real-world knowledge from Wikipedia. The result of the query is the German city with the highest population.

fixes. In the example in Fig. 2.3, the first two lines are prefix definitions. Triplestores may offer configurable default prefixes that can be used without specifying them in the query. The definition of prefixes is optional. Table 2.1 contains the default prefixes of queries in this work.

QUERY FORM SPARQL offers the four query forms of SELECT, ASK, CONSTRUCT and DESCRIBE. Each query form requires a slightly different query structure: The query form SELECT, as used in line 3 of Fig. 2.3, requires a list of variables and produces a result set of variable bindings. ASK queries are constructed similar to SELECT queries but they contain neither Solution Sequence Modifiers nor variables along with the query form.[10] They return a boolean value that indicates, whether there is at least one possible variable binding for the query. CONSTRUCT and DESCRIBE return RDF graphs and are thus not used in this work.

WHERE CLAUSE The WHERE clause (lines 4–9 in Fig. 2.3) contains basic graph patterns. A basic graph pattern is either a triples pattern or a filter.A triple pattern is an RDF triple, where subject, predicate and object each may be replaced by a variable. The triple pattern syntax is based on Turtle. Filters restrict the result set to those rows where the filter condition evaluates to true. The keyword WHERE in front of the WHERE clause can be omitted.

SOLUTION SEQUENCE MODIFIERS (SSMS) The result of a SPARQL query is a set of rows, the *result set*. Specifiying a LIMIT of n results in the first n rows of the result set. By default, the result set is given arbitrary order, so that there is no guarantee on which subset is returned. An ascending, respectively descending order can be imposed using ORDER BY ASC(t), respectively ORDER BY DESC(t), where t is a numerical term based on the variables contained in the query. Fig. 2.3 combines a descending order with a limit of 1 in lines 10 and 11 to find the German city with the highest population value.

10 Triple patterns in ASK queries may still contain variables.

As the standard language, *SPARQL* is the intermediate result of all other query approaches and is thus the most expressive one. Compared to using SPARQL directly, SQA frees users from two major requirements:

1. the mastery of the SPARQL query language
2. knowledge about the specific vocabularies of the knowledge base they want to query.

Besides SPARQL and SQA, there are other query approaches for RDF data, such as SPARQL query builders, faceted browsing, keyword search and controlled vocabularies. They present different tradeoffs between expressiveness, ease of use, initial implementation effort and ease of adaptation of an existing implementation to a new domain, see Table 2.3, and are described in the following:

Table 2.3: Comparison of approaches for querying RDF.

Approach	Expressiveness	Ease of Use	Ease of Implementation	Ease of Adaptation
SPARQL Query	+++	−−	++	++
Controlled Vocabulary	++	−	+	−−
Faceted Search [16]	−	++	−	+
Keyword Search	+	++	−−	−
Question Answering	++	++	−−−	−

2.2.5 *Controlled Vocabulary*

A controlled vocabulary for a knowledge base is a small subset of natural language that is mapped to SPARQL. While it is expressive and easier to implement than SQA, the vocabulary is tailored to a specific domain. Existing queries seem intuitive but users still need to learn the vocabulary and the allowed grammatical constructs. A controlled vocabulary, like SemBT [110], is thus best suited for frequent users of a single domain with a stable schema, such as a library.

2.2.6 *Faceted Search*

Faceted Search approaches, such as Stadler et al. [178], allow selecting from a set of entities based on their common properties. The user restricts values of those properties, which selects a subset of entities, such as smart phones that have Android OS and at least 4 GB of RAM. Faceted Browsing does not require knowledge about a vocabulary or about the entities, as the properties and their possible value ranges are clearly visible, which makes it intuitive and easy to use. It can only be used, however, if three preconditions are met, which significantly impacts its expressivity:

1. The entities need to be so homogeneous that they have enough common properties.
2. The user is looking for a set of entities[11].
3. Restrictions on property values are adequate representations of the information need.

2.2.7 *Keyword Search*

Of the query methods described above, SQA is the only one that is both expressive and intuitive to use. While natural language is expressive, it is also hard to process, which presents difficult challenges to SQA. *Keyword Search* on the other hand, where a set of words is given instead of a sentence, is less expressive but easier to implement. We did not rate one of the approaches as more intuitive, as there are arguments for both positions: while natural language statements are expressed in sentences, users of browsers and smartphones are used to keyword search engines. Regardless, we chose to include Keyword Search approaches in the survey in Chapter 4 as, other than processing natural language questions, they face similar challenges.

2.3 DATA CUBES

Unlike common data representations such as tables or relational databases, the *data cube*[12] formalism adequately represents multidimensional, numerical data.

THE DATA CUBE MODEL A data cube is a multidimensional array of *cells*. Each cell is uniquely identified by its associated dimension values and contains one or more numeric *measurement* values, such as an amount of money spent or received. The *dimensions* thus provide a context to the measurements, such as the purpose, department and time of a spending item. Each dimension has a range, which can be a number system ($\mathbb{N}, \mathbb{R}, \ldots$), dates or an enumeration of constants (code list). Data cubes are often sparse, i.e, for most combinations of dimension values there is no cell in the cube. Optional *attributes* further describe the measured value, such as the unit of the measurement.

THE RDF DATA CUBE (RDC) VOCABULARY Data cubes can be expressed in RDF using the RDF Data Cube Vocabulary, see Fig. 2.4. Each RDF Data Cube is called a data set and is modelled as an instance of `qb:DataSet` with an attached schema, the *data structure definition*, which specifies the *component properties*. Component properties are either dimensions, measures or attributes, whose range is defined either using data types, such as `xsd:dateTime`, or code lists. Measures, of which there has to be at least one, represent the measured quantities, while the dimensions and the optional attributes provide context. Because the RDC vocabulary is focused on statistical data, the cells of an RDF Data Cube are called *observations*. An observation contains exactly one value for each dimension and measure. Attribute values may be specified either in each observation or at the data cube level if the value is constant over the data set, such as the currency used.

11 In contrast to a value, an explanation, a timespan or something else.
12 Also called *Online Analytical Processing (OLAP) cube* or *hypercube*.

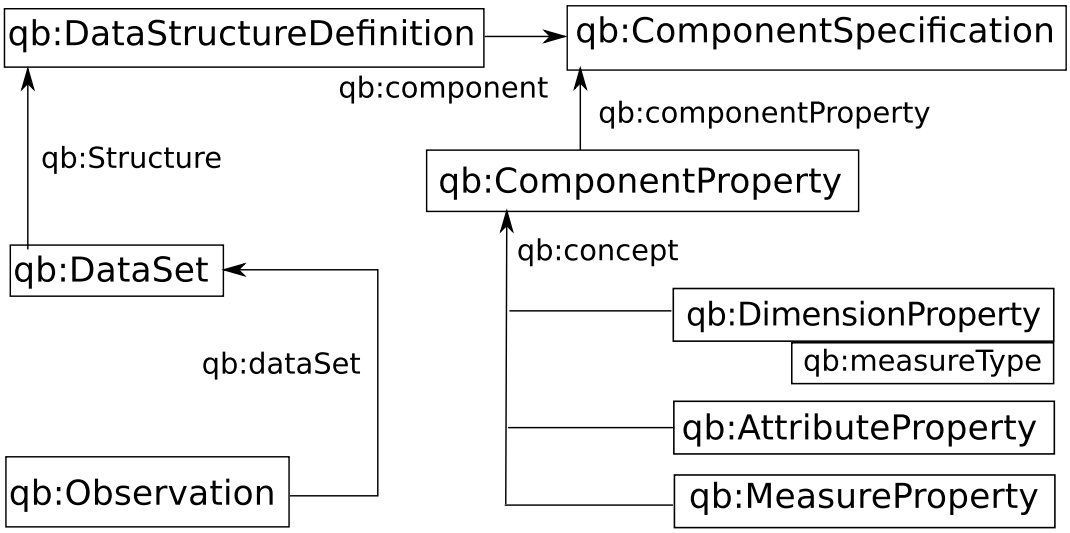

Figure 2.4: Simplified structure of the RDF Data Cube (RDC) vocabulary [50], which determines the triple patterns required for the SPARQL query. A more detailed explanation of the RDC vocabulary is presented in Höffner et al. [103]. Figure originally published in Höffner et al. [105].

3 RELATED WORK

3.1 SEMANTIC QUESTION ANSWERING

This section gives an overview of recent QA and SQA surveys and differences to the survey in this thesis, as well as QA and SQA evaluation campaigns, which quantitatively compare systems.

3.1.1 *Surveys*

Cimiano et al. [47] present a data-driven problem analysis of QA on the Geobase data set. The authors identify eleven challenges that QA has to solve and which inspired the problem categories of this survey: question types, language "light"[1], lexical ambiguities, syntactic ambiguities, scope ambiguities, spatial prepositions, adjective modifiers and superlatives, aggregation, comparison and negation operators, non-compositionality, and out of scope[2]. In contrast to our work, they identify challenges by manually inspecting user provided questions instead of existing systems. Mishra et al. [139] propose eight classification criteria, such as application domain, types of questions and types of data. For each criterion, the different classifications are given along with their advantages, disadvantages and exemplary systems.

SQA SURVEYS Athenikos et al. [10] give an overview of domain specific SQA systems for biomedicine. After summarising the state of the art for biomedical SQA systems in 2009, the authors describe different approaches from the point of view of medical and biological QA. In contrast to our survey, the authors do not sort the presented approaches by challenges, but by more broader terms such as "Non-semantic knowledge base medical QA systems and approaches" or "Inference-based biological QA systems and approaches".

Lopez et al. [128] present an overview similar to Athenikos et al. [10] but with a wider scope. After defining the goals and dimensions of QA and presenting some related and historic work, the authors summarize the achievements of SQA so far and the challenges that are still open.

Another related survey from 2012, Freitas et al. [78], gives a broad overview of the challenges involved in constructing effective query mechanisms for Web-scale data. The authors analyze different SQA systems, such as *Treo* [81], for five different challenges: usability, query expressivity, vocabulary-level semantic matching, entity recognition and improvement of semantic tractability. The same is done for architectural elements such as user interaction and interfaces and the impact on these challenges is reported.

Lopez et al. [127] analyze the SQA systems of the participants of the QALD 1 and 2 evaluation campaigns. While there is an overlap in the surveyed approaches

1 Semantically weak constructions.
2 Cannot be answered as the information required is not contained in the knowledge base.

Table 3.1: Other surveys by year of publication. Surveyed years are given except when a data set is theoretically analyzed. Approaches addressing specific types of data are also indicated.

QA Survey	Year	Coverage	Data
Cimiano et al. [47]	2010	—	geobase
Mishra et al. [139]	2015	2000–2014	general
SQA Survey	**Year**	**Coverage**	**Data**
Athenikos et al. [10]	2010	2000–2009	biomedical
Lopez et al. [128]	2010	2004–2010	general
Freitas et al. [78]	2012	2004–2011	general
Lopez et al. [127]	2013	2005–2012	general

between Lopez et al. [127] and our paper, our survey has a broader scope as it also analyzes approaches that do not take part in the QALD challenges.

In contrast to the surveys mentioned above, we do not focus on the overall performance or domain of a system, but on analyzing and categorizing methods that tackle specific problems, such as ambiguity. Additionally, we build upon the existing surveys and describe the new state of the art systems, which were published after the before mentioned surveys in order to keep track of new research ideas.

After the publication of our survey in Höffner et al. [108], new surveys followed: Diefenbach et al. [58] divide SQA into five tasks: "question analysis, phrase mapping, disambiguation, query construction and querying". The tasks are then mapped to challenges identified in the systems that took part in any of the QALD evaluation campaigns. Techniques for solving these challenges are then discussed. Chakraborty et al. [43] analyze SQA systems that employ neural networks to solve tasks such as query ranking, answer candidate classification and question translation.

3.1.2 *Evaluation Campaigns*

In contrast to QA surveys, which qualitatively compare systems, there are also evaluation campaigns, which quantitatively compare them using benchmarks [195]. Those campaigns show how different open-domain QA systems perform on realistic questions on real-world knowledge bases. This accelerates the evolution of QA in four different ways:

1. New systems do not have to include their own benchmark, shortening system development.
2. Standardized evaluation allows for better research resource allocation as it is easier to determine, which approaches are worthwhile to develop further.
3. The addition of new challenges to the questions of each new benchmark iteration motivates addressing those challenges.

4. The competitive pressure to keep pace with the top scoring systems compells emergence and integration of shared best practises.

On the other hand, evaluation campaign proceedings do not describe single components of those systems in great detail. By focussing on complete systems, research effort gets spread around multiple components, possibly duplicating existing efforts, instead of focussing on a single one.

The core task of QALD is open domain SQA on lexicographic facts of DBpedia [119]. Since its inception in 2011, the yearly benchmark has been made progressively more difficult. Additionally, the general core task has been joined by special tasks providing challenges like multilinguality, hybrid (textual and Linked Data) and its newest addition, RDCQA based on the CubeQA benchmark, see Section 6.7.1.

BioASQ [13, 14, 152, 185] is a benchmark challenge which ran until September 2015. It consists of both a semantic indexing and an SQA part on biomedical data. In the SQA part, systems are expected to be hybrids, returning matching triples as well as text snippets. Partial evaluation (text or triples only) is possible as well. The introductory task separates the process into annotation which is equivalent to named entity recognition (NER) and disambiguation (NED) as well as the answering itself. The second task combines these two steps.

TREC LiveQA, starting in 2015 [3], gives systems unanswered questions from *Yahoo Answers* intended for other humans. As such, the campaign contains the most realistic questions with the least restrictions, in contrast to the solely factual questions of QALD, BioASQ and TREC's old QA track [54].

3.1.3 *System Frameworks*

System frameworks provide an abstraction in which a generic functionality can be selectively added by third-party libraries. In document retrieval, there are many existing frameworks, such as Lucene [26], Solr [175] and Elastic Search [64]. For SQA systems, however, there is still a lack of tools to facilitate the implementation and evaluation processes of SQA systems.

Developing an SQA framework is a hard task because many systems combine Natural Language (NL) techniques with information retrieval methods. One application for Information Retrieval in SQA systems is detecting named entities at the retrieval stage. There are systems that retrieve the answer from an RDF knowledge base by translating the NL question into a SPARQL query at the first stage, a process also known as interpretation [132]. Systems that use the same strategy to obtain information at the retrieval stage may still differ in the analyzer stage. Systems such as Unger et al. [186] use NLP methods to obtain the question's syntactic parse tree to deduce the query intention. Others, such as *SINA* [169] group keywords into segments and map those to resources in a knowledge base. There are also hybrid systems that work both on structured and unstructured data [193] or on a combination of these systems [88]. These efforts have led to a new research sub field that focuses on SQA frameworks, that is, the design and development of common features for SQA systems.

openQA [132] is a modular open-source framework for implementing and instantiating SQA approaches. The framework's main workflow consists of four stages

(interpretation, retrieval, synthesis and rendering) and adjacent modules (context and service). The adjacent modules contain features, such as caching, that are used by multiple components of the main workflow. The framework proposes the *answer formulation* process similar to traditional document retrieval but replaces the query processing and ranking steps with the more general *interpretation* and *synthesis*. The interpretation step comprises all the pre-processing and matching techniques required to deduce the question whereas the synthesis is the process of ranking, merging and confidence estimation required to produce the answer. The authors claim that *openQA* enables a unification of different architectures and methods.

IBM Watson [142] is a massively parallel Question Answering system that integrates its responses among many different sources, including DBpedia [119], Wikipedia and WordNet. Instead of the standard approach, candidates are generated first using multiple interpretations and are then selected based on a combination of scores.

TBSL [186] combines a domain independent and a domain dependent lexicon, which already exists for knowledge bases such as DBpedia and can be adapted to others. First, the users supplies a natural language question or statement. Next, a tagger identifies parts of speech such as nouns and verbs and the two lexicons are used to parse the question. The parse structure along with the identified entities is used to contruct a semantic representation, which is then transformed to an incomplete SPARQL query with placeholders for the entities. Next, entities are identified. For resources and classes, this is done using an Apache Solr index. Solr is much faster than doing a reverse label lookup on a SPARQL endpoint and allows fuzzy matching to bridge the lexical gap. For properties, fuzzy matching alone is often not enough, because their expressions can vary wildly. For example, *married to* and *spouse of* have a very low string similarity. Setting the string similarity threshold of fuzzy matching to a value low enough to match them is not feasible because it results in an extremely high number of false positives and thus a very low precision. The *BOA* [86] pattern library tackles this problem by providing various phrases commonly used to refer to a certain property. The entities are then entered in the placeholders of templates, forming full SPARQL queries that are scored before the one with the highest score is executed. The answer is presented to the user as a list whose items can be marked as correct or incorrect in order to improve the SPARQL query using the AutoSPARQL [120] algorithm.

Treo [81] is a different approach that performs entity recognition and disambiguation using Wikipedia based semantic relatedness and spreading activation. It takes advantage of the context of a word in a sentence and the assumption, that all entities in a sentence are somehow related and thus similar concepts have a higher probability of being correctly identified.

Other approaches for querying RDF include facetted browsing approaches such as Broccoli [16] and Facete [178], which allow intuitive navigation from a certain starting resource of list of resources using property values.

3.2 QUESTION ANSWERING ON RDF DATA CUBES

RDCQA has not existed until recently, but non-semantic QA on multidimensional numerical data is implemented by Wolfram | Alpha, which queries several structured sources using the computational platform *Mathematica* [205], but the source code and algorithm are not published.

Next, we developed CubeQA and QALD6-T3 to stimulate further research, which led to the development of QA[3] and Sparklis (see Section 6.7.2).

CubeQA uses time intervals for handling dates, similar to the system in [184], which uses the *Clinical Narrative Temporal Relation Ontology (CNTRO)* to incorporate the time dimension in answering clinical questions. The ontology is based on Allen's Interval Based Temporal Logic [5] but it represents time points as well. The framework includes a reasoner for time inference, for example based on the transitivity of the *before* and *after* relations. The time dimension is used there to identify the direction of possible causality between different events.

Furthermore, CubeQA generates query templates recursively, which is similarly employed by Intui2 [59], which uses DBpedia and is based on *synfragments*, minimal parse subtrees of a question, that are combined based on syntactic and semantic characteristics to create the final query.

The motivation to develop RDCQA algorithms and their benefit rises with the quantity, quality and significance of available RDCs. On the flipside, we expect that the emergence and improvement of RDCQA algorithms increases the value of RDCs. Because of this interdependence, we summarize efforts to improve the quality of, create and publish RDF in general and RDCs in particular: RDCs are usually created by transforming databases or other structured data sources using either custom software or mapping languages like R2RML[3] and SML [179]. Eurostat—Linked Data[4] transforms tabular data of Eurostat[5], providing statistics for comparing the European countries and regions. LinkedSpending [105] uses the OpenSpending JavaScript Object Notation (JSON) API to provide finance data from countries around the world. A systematic review of Linked Data quality [214] provides a qualitative analysis over established approaches, tools and metrics.

3.3 RDF DATA CUBE DATA SETS

The *TWC Data-Gov Corpus* [61, 62] consists of linked government data from the Data-gov project. However, it only contains transactions made in the US and does not overlap with OpenSpending. The publicspending.gr project generates and publishes [196] public spending data from Greece based on the UK payment ontology and without using data cubes. The UK government expenditure data set COINS[6] is available as Linked Data[7]. *LOD Around-The-Clock (LATC)*[8] is a project, which was funded by the European Union (EU) and converted European open government data into RDF. One of its outcomes is the FTS[9] [131] project, which transforms and publishes financial transparency data on EU spending. In comparison with LinkedSpending, those projects also contribute linked government data but with a different or more limited scope. Furthermore, there is the Digital Agenda Scoreboard [130], which is an EU project that keeps track of the transformation of performance indicators of the EU and its member states to RDCs.

3 https://www.w3.org/TR/r2rml
4 http://eurostat.linked-statistics.org/
5 http://ec.europa.eu/eurostat
6 http://data.gov.uk/dataset/coins
7 http://openuplabs.tso.co.uk/sparql/gov-coins, in a beta version
8 http://latc-project.eu
9 http://ec.europa.eu/budget/fts

4 SYSTEMATIC SURVEY OF SEMANTIC QUESTION ANSWERING

In this chapter, we conduct a systematic survey of SQA systems. The survey is published in the Semantic Web Journal as Höffner et al. [108]. The collection and selection was shared by the first two authors while the description of the approaches was shared by the first four authors. The last two authors motivated the work and provided high-level insights and feedback.

We answer research question RQ1 "What are the current approaches for Semantic Question Answering?" by conducting a systematic survey.

The structure of the survey is as follows:

- Section 3.1 collects existing work that summarizes SQA approaches. We show that existing surveys contains significant gaps, which motivates the creation this survey. We also present evaluation campaigns, system frameworks and work outside the SQA field and contrast their different roles.
- Section 4.1 states the methodology used to find and filter surveyed publications. Applying the inclusion and exclusion criteria leads to to more than 72 publications describing more than 62 systems in the covered timeframe.
- Section 4.2 introduces the surveyed systems.
- Section 4.3 identifies challenges faced by SQA approaches and analyzes, in which way those challenges are tackled by the approaches (RQ1.1 and RQ1.2).
- Section 7.2 summarizes the efforts made to face challenges to SQA. We point to established techniques for each challenge and recommend avenues for future work (RQ1.3).

4.1 METHODOLOGY

This survey follows a strict discovery methodology; Objective inclusion and exclusion criteria are used to find and restrict publications on SQA.

4.1.1 *Inclusion Criteria*

Candidate articles for inclusion in the survey need to be part of relevant conference proceedings or searchable via the publication search engine Google Scholar (see Table 4.1). Candidates from the Google Scholar are all articles that contain "'Question Answering' AND ('Semantic Web' OR 'data web')" in their title, abstract or text body. The first 300 results that do not not meet the exclusion criteria are included.

Conference candidates are all publications in our examined time frame (see Section 4.1.2) in the proceedings of the major Semantic Web Conferences ISWC, ESWC, WWW, NLDB, and the proceedings which contain the annual QALD challenge participants.

4.1.2 *Exclusion Criteria*

Works published before November 2010[1] or after July 2015[2] are excluded, as well as those that are not related to SQA, determined in a manual inspection in the following manner: First, proceeding tracks are excluded that clearly do not contain SQA related publications. Second, publications both from proceedings and from Google Scholar are excluded based on their title and finally on their content.

NOTABLE EXCLUSIONS We exclude the following approaches since they do not fit our definition of SQA (see Chapter 2): *Swoogle* [63] is independent of any specific knowledge base but instead builds its own index and knowledge base using RDF documents found by multiple web crawlers. Discovered ontologies are ranked based on their usage intensity and RDF documents are ranked using authority scoring. *Swoogle* can only find single terms and is thus not a SQA system. *Wolfram | Alpha* is a natural language interface based on the computational platform *Mathematica* [205] and aggregates a large number of structured sources and a algorithms. However, it does not support Semantic Web knowledge bases and the source code and the algorithm are not published. Thus, we cannot identify whether it corresponds to our definition of a SQA system.

4.1.3 *Result*

The inspection of the titles of the Google Scholar results led to the discovery of 153 publications. 39 remained after inspecting the full text (see Table 4.1). The selected proceedings contain 1660 publications, which were narrowed down to 980 by excluding tracks that have no relation to SQA. Based on their titles, 62 of them were selected and inspected, resulting in 33 publications that were categorized and listed in this survey. Table 4.1 shows the number of publications in each step for each source. In total, 1960 candidates were found using the inclusion criteria in Google Scholar and conference proceedings and then reduced using track names (conference proceedings only, 1280 remaining), then titles (214) and finally the full text, resulting in 72 publications describing 62 distinct SQA systems.

4.2 SYSTEMS

The 72 surveyed publications describe 62 distinct systems or approaches. This section exemplifies some of them and their novelties to highlight current research questions, while the next section presents the contributions of all analyzed papers to specific challenges.

4.2.1 *Implementation*

The implementation of an SQA system can be very complex. To reduce the effort of development and increase the performance, several known techniques can be reused, especially in the analyzer stage. The query analyzer generates or formats the query that will be used to recover the answer at the retrieval stage. There are a wide variety of techniques that can be applied at the analyzer stage, such

1 The time before is already covered in Cimiano et al. [47].
2 The time at which the papers were collected.

as tokenisation, disambiguation, internationalization, logical forms, semantic role labels, question reformulation, coreference resolution, relation extraction and named entity recognition amongst others. For some of those techniques, such as NL parsing and POS tagging, mature all-purpose methods are available and commonly reused. Other techniques, such as the disambiguation between multiple possible answer candidates, are not available in a domain independent fashion. Thus, high quality solutions can only be obtained by the development of new components.

4.2.2 *Examples*

Hakimov et al. [91] propose a SQA system using syntactic dependency trees of input questions. The method consists of four main steps:

1. Triple patterns are extracted using the dependency tree and POS tags of the questions.
2. Entities, properties and classes are extracted and mapped to the underlying knowledge base. Recognized entities are disambiguated using page links between all spotted named entities as well as string similarity. Properties are disambiguated by using relational linguistic patterns from *PATTY* [143], which allows a more flexible mapping, such as "die" to `dbo:deathPlace`[3].
3. Question words are matched to the respective answer type, such as "who" to *person, organization or company* and "where" to *place*.
4. The results are ranked and the best result is returned as the answer.

PARALEX [70] only answers questions for subjects or objects of property-object or subject-property pairs, respectively. It contains phrase to concept mappings in a lexicon that is trained from a corpus of paraphrases, which is constructed from the question-answer site WikiAnswers[4]. If one of the paraphrases can be mapped to a query, this query is the correct answer for the paraphrases as well. By mapping phrases between those paraphrases, the linguistic patterns are extended. For example, "what is the r of e" leads to "how r is e ", so that "What is the population of New York City" can be mapped to "How big is NYC". There are a variety of other systems, such as Bordes et al. [30], that make use of paraphrase learning methods and integrate linguistic generalization with knowledge graph biases. They are however not included here as they do not query RDF knowledge bases and thus do not fit the inclusion criteria.

Xser [208] contains two independent steps: First, Xser determines the question structure solely based on a phrase level dependency graph. Secondly, it uses the target knowledge base to instantiate the generated template. Moving to another domain based on a different knowledge base thus only affects parts of the approach so that the adaptation effort is lessened.

QuASE [181] is a three stage open domain approach based on web search and the Freebase knowledge base[5]. First, QuASE uses entity linking, semantic feature construction and candidate ranking on the input question. Then, it selects the documents and according sentences from a web search with a high probability to match the question and presents them as answers to the user.

3 URL prefixes are defined in Table 2.1.
4 `http://wiki.answers.com/`
5 `https://www.freebase.com/`

DEV-NLQ [180] is based on lambda calculus and an event-based triple store[6] using only triple based retrieval operations. DEV-NLQ claims to be the only QA system able to solve chained, arbitrarily-nested, complex, prepositional phrases.

QAKiS [35, 37, 48] queries several multilingual versions of DBpedia at the same time by filling the produced SPARQL query with the corresponding language dependent properties and classes. Thus, it can retrieve correct answers even in cases of missing information in the language dependent knowledge base.

Freitas et al. [79] evaluate a distributional-compositional semantics approach that is independent from manually created dictionaries but instead relies on co-occurring words in text corpora. The vector space over the set of terms in the corpus is used to create a distributional vector space based on the weighted term vectors for each concept. An inverted Lucene index is adapted to the chosen model.

Instead of querying a specific knowledge base, Sun et al. [181] use web search engines to extract relevant text snippets, which are then linked to Freebase, where a ranking function is applied and the highest ranked entity is returned as the answer.

HAWK [193] is the first hybrid source SQA system which processes RDF as well as textual information to answer one input query. HAWK uses an eight-fold pipeline comprising part-of-speech tagging, entity annotation, dependency parsing and linguistic pruning heuristics. It performs an in-depth analysis of the natural language input, semantic annotation of properties and classes, the generation of basic triple patterns for each component of the input query as well as discarding queries containing not connected query graphs and ranking them afterwards.

SWIP (Semantic Web intercase using Pattern) [157] generates a pivot query, a hybrid structure between the natural language question and the formal SPARQL target query. Generating the pivot queries consists of three main steps:

1. named entity identification,
2. query focus identification and
3. sub query generation.

To formalize the pivot queries, the query is mapped to linguistic patterns, which are created by hand from domain experts. If there are multiple applicable linguistic patterns for a pivot query, the user chooses between them.

Hakimov et al. [92] adapt a semantic parsing algorithm to SQA that achieves a high performance. It relies on large amounts of training data, which is not practical when the domain is large or unspecified.

Several industry-driven SQA-related projects have emerged over the last years. For example, DeepQA [75] of IBM Watson [88], which was able to win the Jeopardy! challenge against human experts.

YodaQA [17] is a modular open source hybrid approach built on top of the Apache UIMA framework[76]. It is part of the Brmson platform and is inspired by DeepQA. YodaQA allows easy parallelization and leverage of pre-existing NLP UIMA components by representing each artifact (question, search result, passage, candidate answer) as a separate UIMA Common Analysis Structure. The Yoda pipeline is divided in five stages: Question Analysis, Answer Production, Answer Analysis, Answer Merging and Scoring as well as Successive Refining.

Further, KAIST's Exobrain[7] project aims to learn from large amounts of data while ensuring a natural interaction with end users. However, it is limited to Korean.

6 http://www.w3.org/wiki/LargeTripleStores
7 http://exobrain.kr/

4.2.3 *Answer Presentation*

Another, important part of SQA systems outside the SQA research challenges is result presentation. Verbose descriptions or plain URIs are uncomfortable for human reading. Entity summarization deals with different types and levels of abstractions.

Cheng et al. [44] propose a random surfer model extended by a notion of centrality, which involves the computation of the central elements involving similarity (or relatedness) between them as well as their informativeness. The similarity is given by a combination of the relatedness between their properties and their values.

Ngonga Ngomo et al. [148] present an approach that automatically generates natural language description of resources using their attributes. The rationale behind SPARQL2NL is to verbalize[8] RDF data by applying templates together with the metadata of the schema itself (label, description, type). Entities can have multiple types as well as different levels of hierarchy which can lead to different levels of abstractions. For example, the verbalization of the DBpedia entity `dbr:Microsoft` can vary depending on the type `dbo:Agent` rather than `dbo:Company`.

8 For example, "123"^^<http://dbpedia.org/datatype/squareKilometre> can be verbalized as *123 square kilometres*.

Table 4.1: Sources of publication candidates along with the number of publications in total, after excluding based on conference tracks, the title, and finally on the full text. Works that are found both in a conference's proceedings and in Google Scholar are only counted once, as selected for that conference. The QALD 2 proceedings are included in ILD 2012 [190], QALD 3 [36] and QALD 4 [191] in the CLEF 2013 [77] and 2014 [40] working notes, respectively.

Venue	Total	After exclusion on:	Track	Title	Full Text
Google Scholar Top 300	300		300	153	39
ISWC 2010 [155]	70		70	1	1
ISWC 2011 [9]	68		68	4	3
ISWC 2012 [49]	66		66	4	2
ISWC 2013 [4]	72		72	4	0
ISWC 2014 [136]	31		4	2	0
WWW 2011 [198]	81		9	0	0
WWW 2012 [199]	108		6	2	1
WWW 2013 [200]	137		137	2	1
WWW 2014 [201]	84		33	3	0
WWW 2015 [202]	131		131	1	1
ESWC 2011 [7]	67		58	3	0
ESWC 2012 [174]	53		43	0	0
ESWC 2013 [45]	42		34	0	0
ESWC 2014 [158]	51		31	2	1
ESWC 2015 [84]	42		42	1	1
NLDB 2011 [141]	21		21	2	2
NLDB 2012 [33]	36		36	0	0
NLDB 2013 [134]	36		36	1	1
NLDB 2014 [135]	39		30	1	2
NLDB 2015 [28]	45		10	2	1
QALD 1 [189]	3		3	3	2
ILD 2012 [190]	9		9	9	3
CLEF 2013 [77]	208		7	6	5
CLEF 2014 [40]	160		24	8	6
Σ(conference)	1660		980	61	33
Σ(all)	1960		1280	214	72

4.3 CHALLENGES

In this section, we address seven challenges that have to be faced by state-of-the-art SQA systems. All mentioned challenges are currently open research fields. For each challenge, we describe efforts mentioned in the 72 selected publications. Challenges that affect SQA, but that are not to be solved by SQA systems, such as speech interfaces, data quality and system interoperability, are analyzed in Shekarpour et al. [166].

4.3.1 *Lexical Gap*

In a natural language, the same meaning can be expressed in different ways. Natural language descriptions of RDF resources are provided by values of the `rdfs:label` property (*label* in the following). While synonyms for the same RDF resource can be modeled using multiple labels for that resource, knowledge bases typically do not contain all the different terms that can refer to a certain entity. If the vocabulary used in a question is different from the one used in the labels of the knowledge base, we call this the *lexical gap*[9] [92].

Because a question can usually only be answered if every referred concept is identified, bridging this gap significantly increases the proportion of questions that can be answered by a system. Table 4.2 shows the methods employed by the 72 selected publications for bridging the lexical gap along with examples. As an example of how the lexical gap is bridged outside of SQA, see Lee et al. [118].

Table 4.2: Different techniques for bridging the lexical gap along with examples of deviations to the word "running" that these techniques cover.

Technique	Detected deviation of "running"
Identity	running
Similarity Measure	runnign
Stemming/Lemmatizing	run
AQE—Synonyms	sprint
Pattern libraries	X made a break for Y

STRING NORMALIZATION AND SIMILARITY FUNCTIONS Normalizations, such as conversion to lower case or to base forms, such as "é" to "e", allow matching of slightly different forms and some simple mistakes, such as "Deja Vu" for "déjà vu", and are quickly implemented and executed. More elaborate normalizations use Natural Language Processing (NLP) techniques for stemming, which reduces, for example, both "running" and "ran" to "run".

If normalizations are not enough, the distance—and its complementary concept, similarity—can be quantified using a *similarity function* and a threshold. Common examples of similarity functions are Jaro-Winkler, an edit-distance that measures

9 In linguistics, the term *lexical gap* has a different meaning, referring to a word that has no equivalent in another language.

transpositions, and n-grams, which compares sets of substrings of length n of two strings. Also, one of the surveyed publications, Zhang et al. [216], uses the largest common substring, both between Japanese and translated English words. However, applying such similarity functions can carry harsh performance penalties. While an exact string match can be efficiently executed in a SPARQL triple pattern, similarity scores generally need to be calculated between a phrase and every entity label, which is infeasible on large knowledge bases [193]. There are however efficient indexes for some similarity functions. For instance, the edit distances of two characters or less can be mitigated by using the fuzzy query implementation of a Lucene index that implements a Levenshtein Automaton [164]. Furthermore, Ngonga Ngomo [146] provides a different approach to efficiently calculating similarity scores that could be applied to QA. It requires similarity measures to be *similarity metrics*, where the triangle inequality $\forall x, y, z : s(x, z) \leq s(x, y) + s(y, z)$ must hold. This inequality allows for a large portion of potential matches to be discarded early in the process. This solution is not as fast as using a Levenshtein Automaton but does not place such a tight limit on the maximum edit distance.

AUTOMATIC QUERY EXPANSION Synonyms, like *design* and *plan*, are pairs of words that, either always or only in a specific context, have the same meaning. While normalization and string similarity methods match different forms of the same word, they do not recognize synonyms. In hyper-hyponym-pairs, like *chemical process* and *photosynthesis*, the first word is less specific then the second one. To match both synonym- and hyper-hyponym-pairs, these word pairs are taken from lexical databases such as WordNet [137] and are used as additional labels in Automatic Query Expansion (AQE). AQE is commonly used in information retrieval and traditional search engines, as summarized in Carpineto et al. [41]. These additional surface forms allow for more matches and thus increase recall but lead to mismatches between related words and thus can decrease the precision.

In traditional document-based search engines with high recall and low precision, this trade-off is more common than in SQA. SQA is typically optimized for concise answers and a high precision, since a SPARQL query with an incorrectly identified concept mostly results in a wrong set of answer resources. However, AQE can be used as a backup method in case there is no direct match. One of the surveyed publications is an experimental study [167] that evaluates the impact of AQE on SQA. It has analyzed different lexical[10] and semantic[11] expansion features and used machine learning to optimize weightings for combinations of them. Both lexical and semantic features were shown to be beneficial on a benchmark data set consisting only of sentences where direct matching is not sufficient.

PATTERN LIBRARIES RDF individuals can be matched from a phrase to a resource with high accuracy using similarity functions and normalization alone. Properties however require further treatment, as

1. they determine the subject and object, which can be in different positions[12] and

10 lexical features include synonyms, hyper and hyponyms
11 semantic features rely on RDF graphs and the RDFS vocabulary, such as equivalent, sub- and super-classes
12 E.g., "X wrote Y" and "Y is written by X"

2. a single property can be expressed in many different ways, both as a noun and as a verb phrase which may not even be a continuous substring[13] of the question.

Because of the complex and varying structure of those linguistic patterns and the required reasoning and knowledge[14], libraries to overcome these issues have been developed.

PATTY [143] detects entities in sentences of a corpus and determines the shortest path between the entities. The path is then expanded with occurring modifiers and stored as a pattern. Thus, PATTY is able to build up a pattern library on any knowledge base with an accompanying corpus.

BOA [86] generates linguistic patterns using a corpus and a knowledge base. For each property in the knowledge base, sentences from a corpus are chosen containing examples of subjects and objects for this particular property. BOA assumes that each resource pair that is connected in a sentence exemplifies another label for this relation and thus generates a pattern from each occurrence of that word pair in the corpus.

PARALEX [70] contains phrase to concept mappings in a lexicon that is trained from a corpus of paraphrases from the QA site WikiAnswers. The advantage is that no manual templates have to be created as they are automatically learned from the paraphrases.

ENTAILMENT A corpus of already answered questions or linguistic question patterns can be used to infer the answer for new questions. A phrase A is said to *entail* phrase B, if B follows from A. Thus, entailment is directional: Synonyms entail each other, whereas hyper- and hyponyms entail in one direction only: "birds fly" entails "sparrows fly", but not the other way around. Ou et al. [151] generate possible questions for an ontology in advance and identify the most similar match to a user question based on a syntactic and semantic similarity score. The syntactic score is the cosine-similarity of the questions using bag-of-words. The semantic score also includes hypernyms, hyponyms and denorminalizations based on WordNet [137]. While the preprocessing is algorithmically simple compared to the complex pipeline of NLP tools, the number of possible questions is expected to grow superlinearly with the size of the ontology so the approach is more suited to specific domain ontologies. Furthermore, the range of possible questions is quite limited which the authors aim to partially alleviate in future work by combining multiple *basic questions* into a *complex question*.

DOCUMENT RETRIEVAL MODELS Blanco et al. [29] adapt entity ranking models from traditional document retrieval algorithms to RDF data. The authors apply BM25 as well as the tf-idf ranking function to an index structure with different text fields constructed from the title, object URIs, property values and RDF inlinks. The proposed adaptation is shown to be both time efficient and qualitatively superior to other state-of-the-art methods in ranking RDF resources.

COMPOSITE APPROACHES Elaborate approaches on bridging the lexical gap can have a high impact on the overall runtime performance of an SQA system. This can

13 E.g., "X wrote Y together with Z" for "X is a coauthor of Y".
14 E.g., "if X writes a book, X is called the author of it."

be partially mitigated by composing methods and executing each following step only if the one before did not return the expected results.

BELA [203] implements four layers. First, the question is mapped directly to the concept of the ontology using the index lookup. Second, the question is mapped based on Levenshtein distance to the ontology, if the Levenshtein distance of a word from the question and a property from an ontology exceed a certain threshold. Third, WordNet is used to find synonyms for a given word. Finally, BELA uses explicit semantic analysis (ESA, see Gabrilovich et al. [83]). The evaluation is carried out on the QALD 2 [190] test data set and shows that the more simple steps, like index lookup and Levenshtein distance, had the most positive influence on answering questions so that many questions can be answered with simple mechanisms.

Park et al. [153] answer natural language questions via regular expressions and keyword queries with a Lucene-based index. Furthermore, the approach uses DBpedia [122] as well as their own triple extraction method on the English Wikipedia.

4.3.2 *Ambiguity*

Ambiguity is the phenomenon of the same phrase having different meanings; this can be structural and syntactic (like "flying planes") or lexical and semantic (like "bank"). We distinguish between homonymy, where the same string accidentally refers to different concepts (as in money bank vs. river bank) and polysemy, where the same string refers to different but related concepts (as in bank as a company vs. bank as a building). We also distinguish between synonymy and taxonomic relations such as metonymy and hypernymy. In contrast to the lexical gap, which impedes the recall of a SQA system, ambiguity negatively effects its precision. Ambiguity is the flipside of the lexical gap. This problem is aggravated by the very methods used for overcoming the lexical gap. The more loose the matching criteria become (increase in recall), the more candidates are found which are generally less likely to be correct than closer ones.

DISAMBIGUATION Disambiguation is the process of selecting one of multiple candidate concepts for an ambiguous phrase. We differentiate between two types of disambiguation based on the source and type of information used to solve this mapping.

CORPUS-BASED Corpus-based methods are traditionally used and rely on counts, often used as probabilities, from unstructured text corpora. Such statistical approaches [173] are based on the *distributional hypothesis*, which states that "difference of meaning correlates with difference of [contextual] distribution" [95]. The *context* of a phrase is identified here as its central characteristic [138]. Common context features used are word co-occurrences, such as left or right neighbours, but also synonyms, hyponyms, POS-tags and the parse tree structure. More elaborate approaches also take advantage of the context outside of the question, such as past queries of the user [172] .

RESOURCE-BASED In SQA, resource-based methods exploit the fact that the candidate concepts are RDF resources. Resources are compared using different scoring schemes based of their properties and the connections between them. The assumption is that high scores between all the resources chosen in the mapping

imply a higher probability of those resources being related, and that this implies a higher propability of those resources being correctly chosen.

APPROACHES *RVT* [87] uses Hidden Markov Models (HMMs) to select the proper ontological triples according to the graph nature of DBpedia.

CASIA [97] employs *Markov Logic Networks (MLN)*: First-order logic statements are assigned a numerical penalty, which is used to define hard constraints, like "each phrase can map to only one resource", alongside soft constraints, like "the larger the semantic similarity is between two resources, the higher the chance is that they are connected by a relation in the question".

Unger et al. [188] employ *underspecification* to discard certain combinations of possible meanings before the time consuming querying step, by combining restrictions for each meaning. Each term is mapped to a *Dependency-based Underspecified Discourse REpresentation Structure (DUDE [46])*, which captures its possible meanings along with their class restrictions.

Treo [80, 81] performs entity recognition and disambiguation using Wikipedia-based semantic relatedness and spreading activation. Semantic relatedness calculates similarity values between pairs of RDF resources. Determining semantic relatedness between entity candidates associated to words in a sentence allows to find the most probable entity by maximizing the total relatedness.

EasyESA [42] is based on distributional semantic models which allow to represent an entity by a vector of target words and thus compresses its representation. The distributional semantic models allow to bridge the lexical gap and resolve ambiguity by avoiding the explicit structures of RDF-based entity descriptions for entity linking and relatedness.

gAnswer [111] tackles ambiguity with *RDF fragments*, i.e., star-like RDF subgraphs. The number of connections between the fragments of the resource candidates is then used to score and select them. The lexical gap for relations is covered by a dictionary, which is automatically built in advance.

Wikimantic [31] can be used to disambiguate short questions or even sentences. It uses Wikipedia article interlinks for a generative model, where the probability of an article to generate a term is set to the terms relative occurrence in the article. Disambiguation is then an optimization problem to locally maximize each article's (and thus DBpedia resource's) term probability along with a global ranking method.

Shekarpour et al. [170, 171] disambiguate resource candidates using segments consisting of one or more words from a keyword query. The aim is to maximize the high textual similarity of keywords to resources along with relatedness between the resources (classes, properties and entities). The problem is cast as a HMM with the states representing the set of candidate resources extended by Web Ontology Language (OWL) reasoning. The transition probabilities are based on the shortest path between the resources. The Viterbi algorithm generates an optimal path though the HMM that is used for disambiguation.

DEANNA [209, 210] manages phrase detection, entity recognition and entity disambiguation by formulating the SQA task as an integer linear programming (ILP) problem. It employs *semantic coherence* which measures co-occurrence of resources in the same context. DEANNA constructs a disambiguation graph, which encodes the selection of candidates for resources and properties. The chosen objective function maximizes the combined similarity while constraints guarantee that the selections are valid. The resulting problem is NP-hard (non-deterministic polynomial-time

hard) but it is efficiently solvable in approximation by existing ILP solvers. The follow-up approach [211] uses DBpedia and Yago with a mapping of input queries to semantic relations based on text search. At QALD 2, it outperformed almost every other system on factoid questions and every other system on list questions. However, the approach requires detailed textual descriptions of entities and only creates basic graph pattern queries.

LOD-Query [165] is a keyword-based SQA system that tackles both ambiguity and the lexical gap by selecting candidate concepts based on a combination of a string similarity score and the connectivity degree. The string similarity is the normalized edit distance between a label and a keyword. The connectivity degree of a concept is approximated by the occurrence of that concept in all the triples of the knowledge base.

Pomelo [93] answers biomedical questions on the combination of Drugbank, Diseasome and Sider using `owl:sameAs` links between them. Properties are disambiguated using predefined rewriting rules which are categorized by context. Rani et al. [160] use fuzzy logic co-clustering algorithms to retrieve documents based on their ontology similarity. Possible senses for a word are assigned a probability depending on the context. Zhang et al. [216] translate RDF resources to the English DBpedia. It uses feedback learning in the disambiguation step to refine the resource mapping

USER INPUT Instead of trying to resolve ambiguity automatically, some approaches let the user clarify the exact intent, either in all cases or only for ambiguous phrases:

SQUALL [72, 73] defines a controlled vocabulary based on English that is enhanced with knowledge from a given triple store. While this ideally results in a high performance, it moves the problem of the lexical gap and disambiguation fully to the user. As such, it covers a middle ground between SPARQL and full-fledged SQA with the author's intent that learning the grammatical structure of this proposed language is easier for a non-expert than to learn SPARQL.

A cooperative approach that places less of a burden on the user is proposed in Melo et al. [133], which transforms the question into a discourse representation structure and starts a dialogue with the user for all occurring ambiguities.

CrowdQ [57] is a SQA system that decomposes complex queries into simple parts (keyword queries) and uses crowdsourcing for disambiguation. It avoids excessive usage of crowd resources by creating general templates as an intermediate step.

FREyA (Feedback, Refinement and Extended VocabularY Aggregation) [51] represents phrases as potential ontology concepts which are identified by heuristics on the syntactic parse tree. Ontology concepts are identified by matching their labels with phrases from the question without regarding its structure. A consolidation algorithm then matches both potential and ontology concepts. In case of ambiguities, feedback from the user is asked. Disambiguation candidates are created using string similarity in combination with WordNet synonym detection. The system learns from the user selections, thereby improving the precision over time.

TBSL [186] uses both a domain independent and a domain dependent lexicon so that it performs well on specific topic but is still adaptable to a different domain. It uses AutoSPARQL [120] to refine the learned SPARQL using the *QTL* algorithm for supervised machine learning. The user marks certain answers as correct or incorrect and triggers a refinement. This is repeated until the user is satisfied with the result.

An extension of *TBSL* is *DEQA* [123], which combines Web extraction with OXPath [82], interlinking with *LIMES* [147] and SQA with *TBSL*. It can thus answer complex questions about objects which are only available as HTML. Another extension of *TBSL* is *ISOFT* [154], which uses explicit semantic analysis to help bridging the lexical gap.

NL-Graphs [66] combines SQA with an interactive visualization of the graph of triple patterns in the query which is close to the SPARQL query structure yet still intuitive to the user. Users that find errors in the query structure can either reformulate the query or modify the query graph.

KOIOS [27] answers queries on natural environment indicators and allows the user to refine the answer to a keyword query by faceted search. Instead of relying on a given ontology, a schema index is generated from the triples and then connected with the keywords of the query. Ambiguity is resolved by user feedback on the top ranked results.

ANSWER TYPES A different way to restrict the set of answer candidates and thus handle ambiguity is to determine the expected answer type of a factual question. The standard approach to determine this type is to identify the focus of the question and to map this type to an ontology class. In the example "Which books are written by Dan Brown?", the focus is "books", which is mapped to `dbo:Book`. There is however a long tail of rare answer types that are not as easily alignable to an ontology, which, for instance, Watson [88] tackles using the TyCor [116] framework for type coercion. Instead of the standard approach, candidates are first generated using multiple interpretations and then selected based on a combination of scores. Besides trying to align the answer type directly, it is *coerced* into other types by calculating the probability of an entity of class A to also be in class B. DBpedia, Wikipedia and WordNet are used to determine link anchors, list memberships, synonyms, hyper- and hyponyms. The follow-up [204] compares two different approaches for answer typing. Type-and-Generate (TaG) approaches restrict candidate answers to the expected answer types using predictive annotation, which requires manual analysis of a domain. Tycor on the other hand employs multiple strategies using generate-and-type (GaT), which generates all answers regardless of answer type and tries to coerce them into the expected answer type. Experimental results hint that GaT outperforms TaG when the accuracy is higher than 50%. The significantly higher performance of TyCor when using GaT is explained by its robustness to incorrect candidates while there is no recovery from excluded answers from TaG.

4.3.3 *Multilingualism*

Knowledge on the Web is expressed in various languages. While RDF resources can be described in multiple languages at once using language tags, there is not a single language that is always used in Web documents. Additionally, users have different native languages. A more flexible approach is thus to have SQA systems that can handle multiple input languages, which may even differ from the language used to encode the knowledge.

GermanNLI [55] uses *GermaNet* [94], which is integrated into the multilingual knowledge base *EuroWordNet* [197], together with lemon-LexInfo [34] to answer German questions.

Aggarwal et al. [2] only need to successfully translate part of the query, after which the recognition of the other entities is aided using semantic similarity and relatedness measures between resources connected to the initial ones in the knowledge base.

QAKiS (Question Answering wiKiframework-based System) [48] automatically extends existing mappings between different language versions of Wikipedia, which is carried over to DBpedia.

4.3.4 *Complex Queries*

Simple questions can most often be answered by translation into a set of simple triple patterns. Problems arise when several facts have to be found out, connected and then combined. Queries may also request a specific result order or results that are aggregated or filtered.

YAGO-QA [1] allows nested queries when the subquery has already been answered, for example "Who is the governor of the state of New York?" after "What is the state of New York?" YAGO-QA extracts facts from Wikipedia (categories and infoboxes), WordNet and GeoNames. It contains different surface forms such as abbreviations and paraphrases for named entities.

PYTHIA [187] is an ontology-based SQA system with an automatically build ontology-specific lexicon. Due to the linguistic representation, the system is able to answer natural language question with linguistically more complex queries, involving quantifiers, numerals, comparisons and superlatives, negations and so on.

IBM Watson [88] handles complex questions by first determining the focus element, which represents the searched entity. The information about the focus element is used to predict the lexical answer type and thus restrict the range of possible answers. This allows for indirect questions and multiple sentences.

Shekarpour et al. [170, 171], see also Section 4.3.2, propose a model that combines knowledge base concepts with a HMM to handle complex queries.

Intui2 [59] is an SQA system based on DBpedia based on *synfragments* which map to a subtree of the syntactic parse tree. Semantically, a synfragment is a minimal span of text that can be interpreted as an RDF triple or complex RDF query. Synfragments interoperate with their parent synfragment by combining all combinations of child synfragments, ordered by syntactic and semantic characteristics. The authors assume that an interpretation of a question in any RDF query language can be obtained by the recursively interpretation of its synfragments. *Intui3* [60] replaces self-made components with robust libraries such as the neural networks-based NLP toolkit SENNA and the DBpedia Lookup service. It drops the parser determined interpretation combination method of its predecessor that suffered from bad sentence parses and instead uses a fixed order right-to-left combination.

GETARUNS [56] first creates a logical form out of a query which consists of a focus, a predicate and arguments. The focus element identifies the expected answer type. For example, the focus of "Who is the major of New York?" is "person", the predicate "be" and the arguments "major of New York". If no focus element is detected, a yes/no question is assumed. In the second step, the logical form is converted to a SPARQL query by mapping elements to resources via label matching. The resulting triple patterns are then split up again as properties are referenced by unions over both possible directions, as in (`{?x ?p ?o} UNION {?o ?p ?x}`) because

the direction is not known beforehand. Additionally, there are filters to handle additional restrictions which cannot be expressed in a SPARQL query, such as "Who has been the 5th president of the USA".

4.3.5 *Distributed Knowledge*

If concept information–which is referred to in a query–is represented by distributed RDF resources, information needed for answering it may be missing if only a single one or not all of the knowledge bases are found. In single data sets with a single source, such as DBpedia, however, most of the concepts have at most one corresponding resource. In case of combined data sets, this problem can be dealt with by creating `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty` links, respectively. However, interlinking while answering a semantic query is a separate research area and thus not covered here.

Some questions are only answerable using multiple knowledge bases and exising interlinks. The *ALOQUS* [115] system tackles this problem by using the *PROTON* [53] upper level ontology first to phrase the queries. The ontology is then aligned to those of other knowledge bases using the *BLOOMS* [114] system. Complex queries are decomposed into separately handled subqueries after coreferences[15] are extracted and substituted. Finally, these alignments are used to execute the query on the target systems. In order to improve the speed and quality of the results, the alignments are filtered using a threshold on the confidence measure.

CRM [98] searches for entities and consolidates results from multiple knowledge bases. Similarity metrics are used both to determine and rank result candidates of each datasource and to identify matches between entities from different datasources.

4.3.6 *Procedural, Temporal and Spatial Questions*

PROCEDURAL QUESTIONS Factual, list and yes-no questions are easiest to answer as they conform directly to SPARQL queries using SELECT and ASK. Others, such as why (causal) or how (procedural) questions require more additional processing. Procedural QA can currently not be solved by SQA, since, to the best of our knowledge, there are no existing knowledge bases that contain procedural knowledge. While it is not an SQA system, we describe the document-retrieval based *KOMODO* [38] to motivate further research in this area. Instead of an answer sentence, KOMODO returns a Web page with step-by-step instructions on how to reach the goal specified by the user. This reduces the problem difficulty as it is much easier to find a Web page which contains instructions on how to, for example, assemble an "Ikea Billy bookshelf" than it would be to extract, parse and present the required steps to the user. Additionally, there are arguments explaining reasons for taking a step and warnings against deviation. Instead of extracting the sense of the question using an RDF knowledge base, KOMODO submits the question to a traditional search engine. The highest ranked returned pages are then cleaned and procedural text is identified using statistical distributions of certain POS tags.

In basic RDF, each fact, which is expressed by a triple, is assumed to be true, regardless of circumstances. In the real world and in natural language however,

15 Such as "List the Semantic Web *people* and *their* affiliation.", where the coreferent *their* refers to the entity *people*.

the truth value of many statements is not a constant but a function of either the location or the time or both.

TEMPORAL QUESTIONS Tao et al. [184] answer temporal question on clinical narratives. They introduce the Clinical Narrative Temporal Relation Ontology (CNTRO), which is based on Allen's Interval Based Temporal Logic [5] but allows usage of time instants as well as intervals. This allows inferring the temporal relation of events from those of others, for example by using the transitivity of *before* and *after*. In CNTRO, measurement, results or actions done on patients are modeled as events whose time is either absolutely specified in date and optionally time of day or alternatively in relations to other events and times. The framework also includes an *SWRL* [109] based reasoner that can deduce additional time information. This allows the detection of possible causalities, such as between a therapy for a disease and its cure in a patient.

Melo et al. [133] propose to include the implicit temporal and spatial context of the user in a dialog in order to resolve ambiguities. It also includes spatial, temporal and other implicit information.

QALL-ME [71] is a multilingual framework based on description logics and uses the spatial and temporal context of the question. If this context is not explicitly given, the location and time are of the user posing the question are added to the query. This context is also used to determine the language used for the answer, which can differ from the language of the question.

SPATIAL QUESTIONS In RDF, a location can be expressed as 2-dimensional geo-coordinates with latitude and longitude, while three-dimensional representations (e.g. with additional height) are not supported by the most often used schema[16]. Alternatively, spatial relationships can be modeled which are easier to answer as users typically ask for relationships and not for exact geocoordinates.

Younis et al. [213] employ an inverted index for named entity recognition that enriches semantic data with spatial relationships such as crossing, inclusion and nearness. This information is then made available for SPARQL queries.

4.3.7 *Templates*

For complex questions, where the resulting SPARQL query contains more than one basic graph pattern, sophisticated approaches are required to capture the structure of the underlying query. Current research follows two paths, namely

1. template based approaches, which map input questions to either manually or automatically created SPARQL query templates or
2. template-free approaches that try to build SPARQL queries based on the given syntactic structure of the input question.

For the first solution, many (1) template-driven approaches have been proposed like *TBSL* [186] or *SINA* [170, 171]. Furthermore, *Casia* [96] generates the graph pattern templates by using the question type, named entities and POS tags. The generated graph patterns are then mapped to resources using WordNet, PATTY and similarity measures. Finally, the possible graph pattern combinations are used

16 see http://www.w3.org/2003/01/geo/wgs84_pos at http://lodstats.aksw.org

to build SPARQL queries. The system focuses on the generation of SPARQL queries that do not need filter conditions, aggregations and superlatives.

Ben Abacha et al. [18] focus on a narrow medical patients-treatment domain and use manually created templates alongside machine learning.

Damova et al. [52] return well formulated natural language sentences that are created using a template with optional parameters for the domain of paintings. Between the input query and the SPARQL query, the system places the intermediate step of a multilingual description using the Grammatical Framework [161], which enables the system to support 15 languages.

Rahoman et al. [159] propose a template based approach using keywords as input. Templates are automatically constructed from the knowledge base.

However, (2) template-free approaches require additional effort of making sure to cover every possible basic graph pattern [193]. Thus, only few SQA systems take this approach so far.

Xser [208] first assigns semantic labels, which are variables, entities, relations and categories, to phrases by casting them to a sequence labelling pattern recognition problem which is then solved by a structured perceptron. The perceptron is trained using features including n-grams of POS tags, Named Entity Recognition (NER) tags and words. Thus, Xser is capable of covering any complex basic graph pattern.

Going beyond SPARQL queries is TPSM, the open domain Three-Phases Semantic Mapping [85] framework. It maps natural language questions to OWL queries using Fuzzy Constraint Satisfaction Problems. Constraints include surface text matching, preference of POS tags and the similarity degree of surface forms. The set of correct mapping elements acquired using the FCSP-SM algorithm is combined into a model using predefined templates.

An extension of gAnswer [217] (see Section 4.3.2) is based on question understanding and query evaluation. First, their approach uses a relation mining algorithm to find triple patterns in queries as well as relation extraction, POS-tagging and dependency parsing. Second, the approach tries to find a matching subgraph for the extracted triples and scores them based on a confidence score. Finally, the top-k subgraph matches are returned. Their evaluation on QALD 3 [36] shows that mapping NL questions to graph patterns is not as powerful as generating SPARQL (template) queries with respect to aggregation and filter functions needed to answer several benchmark input questions.

Table 4.3: Number of publications per year per addressed challenge. Percentages are given for the fully covered years 2011–2014 separately and for the whole covered timespan, with 1 decimal place. For a full list, see Table 4.4.

Year	Total	Lexical Gap	Ambiguity	Multilingualism	Complex Operators	Distributed Knowledge	Procedural, Temporal or Spatial	Templates
				absolute				
2010	1	0	0	0	0	0	1	0
2011	16	11	12	1	3	1	2	2
2012	14	6	7	1	2	1	1	4
2013	20	18	12	2	5	1	1	5
2014	13	7	8	1	2	0	1	0
2015	6	5	3	1	0	1	0	0
all	70	46	42	6	12	4	6	11
				percentage				
2011		68.8	75.0	6.3	18.8	6.3	12.5	12.5
2012		42.9	50.0	7.1	14.3	7.1	7.1	28.6
2013		85.0	60.0	10.0	25.0	5.0	5.0	25.0
2014		53.8	61.5	7.7	15.4	7.7	7.7	0.0
all		65.7	60.0	8.6	17.1	5.7	8.6	15.7

Table 4.4: Surveyed publications from November 2010 to July of 2015, inclusive, along with the challenges they explicitely address and the approach or system they belong to. Additionally annotated is the use of templates. In case the system or approach is not named in the publication, a name is generated using the last name of the first author and the year of the first included publication.

Publication	System or Approach	Year	Lexical Gap	Ambiguity	Multilingualism	Complex Operators	Distributed Knowledge	Procedural, Temporal or Spatial	Templates
Tao et al. [184]	Tao10	2010						✓	
Adolphs et al. [1]	YAGO-QA	2011	✓	✓					
Blanco et al. [29]	Blanco11	2011	✓						
Canitrot et al. [38]	KOMODO	2011	✓	✓					
Damljanovic et al. [51]	FREyA	2011	✓	✓					
Ferrandez et al. [71]	QALL-ME	2011			✓			✓	
Freitas et al. [81]	Treo	2011	✓	✓		✓	✓		
Gao et al. [85]	TPSM	2011	✓	✓					
Kalyanpur et al. [116]	Watson	2011		✓					
Melo et al. [133]	Melo11	2011	✓	✓					
Moussa et al. [140]	QASYO	2011	✓						✓
Ou et al. [151]	Ou11	2011	✓	✓		✓			✓
Shen et al. [172]	Shen11	2011		✓					
Unger et al. [188]	Pythia	2011		✓					
Unger et al. [187]	Pythia	2011	✓	✓		✓		✓	
Bicer et al. [27]	KOIOS	2011	✓	✓					
Freitas et al. [80]	Treo	2011							
Ben Abacha et al. [18]	MM+/BIO-CRF-H	2012							✓
Boston et al. [31]	Wikimantic	2012		✓					
Gliozzo et al. [88]	Watson	2012				✓			
Joshi et al. [115]	ALOQUS	2012				✓	✓		

Publication	System or Approach	Year	Lexical Gap	Ambiguity	Multilingualism	Complex Operators	Distributed Knowledge	Procedural, Temporal or Spatial	Templates
Lehmann et al. [123]	DEQA	2012	✓	✓					✓
Yahya et al. [209]	DEANNA	2012							
Yahya et al. [210]	DEANNA	2012	✓	✓					
Shekarpour et al. [170]	SINA	2012	✓	✓					
Unger et al. [186]	TBSL	2012	✓	✓					✓
Walter et al. [203]	BELA	2012	✓	✓	✓				✓
Younis et al. [213]	Younis12	2012	✓					✓	
Welty et al. [204]	Watson	2012		✓					
Elbedweihy et al. [67]	Elbedweihy12	2012							
Cabrio et al. [35]	QAKiS	2012							
Demartini et al. [57]	CrowdQ	2013		✓		✓			
Aggarwal et al. [2]	Aggarwal13	2013	✓	✓	✓				
Deines et al. [55]	GermanNLI	2013	✓						
Dima [59]	Intui2	2013	✓			✓			
Fader et al. [70]	PARALEX	2013	✓	✓				✓	✓
Ferré [73]	SQUALL2SPARQL	2013	✓			✓			
Giannone et al. [87]	RTV	2013	✓						
Hakimov et al. [91]	Hakimov13	2013	✓	✓					
He et al. [96]	CASIA	2013	✓	✓		✓			✓
Herzig et al. [98]	CRM	2013	✓	✓			✓		
Huang et al. [111]	gAnswer	2013	✓	✓					
Pradel et al. [157]	SWIP	2013	✓						
Rahoman et al. [159]	Rahoman13	2013							✓
Shekarpour et al. [171]	SINA	2013	✓	✓					✓
Shekarpour et al. [167]	SINA	2013	✓						

Publication	System or Approach	Year	Lexical Gap	Ambiguity	Multilingualism	Complex Operators	Distributed Knowledge	Procedural, Temporal or Spatial	Templates
Shekarpour et al. [165]	SINA	2013	✓	✓					✓
Delmonte [56]	GETARUNS	2013	✓	✓		✓			
Cojan et al. [48]	QAKiS	2013			✓				
Yahya et al. [211]	SPOX	2013	✓	✓					
Zhang et al. [216]	Kawamura13	2013	✓	✓					
Carvalho et al. [42]	EasyESA	2014	✓	✓					
Rani et al. [160]	Rani14	2014		✓					
Zou et al. [217]	Zhou14	2014		✓					
Stewart [180]	DEV-NLQ	2014				✓			
Höffner et al. [103]	CubeQA	2014	✓	✓				✓	
Cabrio et al. [37]	QAKiS	2014			✓				
Freitas et al. [79]	Freitas14	2014	✓	✓					
Dima [60]	Intui3	2014	✓			✓			
Hamon et al. [93]	POMELO	2014		✓					
Park et al. [154]	ISOFT	2014	✓						
He et al. [97]	CASIA	2014	✓	✓					
Xu et al. [208]	Xser	2014	✓						
Elbedweihy et al. [66]	NL-Graphs	2014		✓					
Sun et al. [182]	QuASE	2015	✓	✓					
Park et al. [153]	Park15	2015	✓	✓					
Damova et al. [52]	MOLTO	2015			✓				
Sun et al. [181]	QuASE	2015	✓	✓					
Usbeck et al. [193]	HAWK	2015	✓				✓		
Hakimov et al. [92]	Hakimov15	2015	✓						

5 QUESTION ANSWERING ON RDF DATA CUBES

This chapter presents CubeQA, the first algorithm for SQA on RDF Data Cubes. The approach is motivated in Höffner et al. [103], which is published in the proceedings of the International Conference on Semantic Systems of 2014 and implemented in Höffner et al. [104], which is published in the proceedings of the International Semantic Web Conference of 2016. The whole motivation, implementation and benchmark creation (and the evaluation in Chapter 6) were carried out by the author.

We answer research question RQ2 "How can SQA be applied to RDCs?" as follows: Section 5.1 presents the corpus of user questions (RQ2.1), which is analyzed to determine typical information needs in Section 5.2 (RQ2.2). Section 5.3 explains datacube operations and uses them to answer an example question. Section 5.4 describes the CubeQA algorithm following from those considerations (RQ2.3). Section 3.2 shows existing SQA approaches and their workflow and differentiates, which of their parts can be reused and which ones have to be adapted for statistical data.

5.1 QUESTION CORPUS

A corpus of 50 typical user questions to a hypothetical RDCQA system is individually provided by six researchers in the field of the Semantic Web. The participants were asked to provide questions that are typical for their numerical information needs with a focus on governement financial spending and budget data. Some of the questions contain grammatical errors or are missing a question mark, which should be taken into account when developing a RDCQA system. Twelve questions are given as examples in Table 5.1 while the full 50 questions are shown in appendix A.

1a	What was the average student grade per semester in year 2010?
1b	How many diseases have a rate of >100 deaths per year?
2a	How much money, does Leipzig and Dresden spend on child care in relation to the birth rate in comparison to the average in Saxony.
2b	How much money does Leipzig get from Saxony for education compared to other major cities in Saxony.
3a	What is the average monthly income of a German citizen?
3b	What was the average inflation in Germany over the last 10 years?
4a	How much money was invested to fight bicycle thefts in Leipzig?
4b	Which were the top 10 funded research institutions in Europe in 2013?
5a	How many citizens live in a <certain area>?
5b	How much money spend <X> on <Y>?
6a	How many professor positions per students in a university in Germany?
6b	How many spin-off companies were created from Government budget?

Table 5.1: The first two questions of of the six survey participants.

5.2 CORPUS ANALYSIS

question word	expected answer type	*f*
how much	uncountable	19
what	uncountable, countable, count, temporal, location, entity	12
how many	countable, count	11
which	temporal, location, entity	3
where	location or purpose	2
how is	uncountable, countable, count, temporal, location, entity	1
relate	comparison or visualization	1
none	uncountable, countable, count, temporal, location, entity	1
total		50

Table 5.2: Frequency of question words in the corpus along with their potential expected answer types. "Countable" and "uncountable" quantities are refered to in the grammatical sense.

Table 5.2 presents the different question words in the corpus, which give information about the expected answer type. Some SQA approaches, such as IBM Watson [142], use the expected answer type to filter out wrong answer candidates and thus improve the precision of the answer. In RDCQA, the dimensionality of the answer needs to be taken into account, which can only be known after executing the query in full or, to get an upper bound, determining the data set and relating its model with the dimensions which are fixed in the query. As such, additional steps need to be taken in order to determine the expected presentation type of the answer, see Table 5.2. Table 5.3 shows the frequencies of the expected presentation types in the corpus. The most common one is that of a single (0-dimensional) value for a certain measurement, which can be presented as a simple text sentence, such as "The Frankfurt city budget of bus transportation in 2014 is 36 million euro". If all dimensions are fixed by the question, there is only one answer and that value is contained in a single observation. When the result contains multiple answers and one dimension, a list of all values can be shown in data cubes it is often not an intuitive answer for the user, as the number of results can be very large. This can require an aggregation even if it is not explicitly mentioned in the question. Multiple free dimension require either an aggregation or a visualization. In fact, visualizations are explicitly mentioned only in two of the 19 cases ("display it on a map" and "what does [...] look like"). The other 17 cases are treated as expected visualization because the result is multidimensional. In 9 of the 12 questions where single values are expected, an aggregate is necessary in order to generate this value (see Table 5.4). In 5 cases, the aggregate type is explicitly mentioned ("average", "total", "the biggest") while in 4 questions it has to be inferred ("How many kids are born in Berlin on a single day?"). If multiple aggregations are possible (e.g. arithmetic mean or the median), they can all be presented to the user at the same

expected presentation type	f
visualization	19
single measurement value	12
percentage value	4
entity or set of entities	4
correlation statement	1
unknown	10
total	50

Table 5.3: Frequency of expected presentation types of questions in the corpus.

time. Users often don't mention the name of a measure but instead its unit, e.g. "How much money was invested to fight bicycle thefts in Leipzig?". In this case, the attributes describing the unit will be used to select the correct measure. Most

phrase	aggregate	f
average	arith. mean	3
total	sum	1
(on) a <timespan>	arith. mean	2
how much does . . . a <class> <measure>	arith. mean	2
the biggest	max	1
total		9

Table 5.4: References to aggregates in the corpus.

used, with 18 occurrences, is "how much", each of those referencing the amount of money spent in some context. Generalizing, we map "how much" to a value of a measure in a specific observation in the given context. If the context refers to multiple observations, an aggregate (see Table 5.4) is used.

EXAMPLE: "HOW MUCH MONEY WAS INVESTED TO FIGHT BICYCLE THEFTS IN LEIPZIG?" The verb used in the example is "invested", which is not guaranteed to match the label of a typical measure property, which could be called "amount spent". This examplifies the traditional difficulties of Question Answering approaches in matching properties which translates to difficulties matching measure and dimension properties in QA over RDCs. Additionally, the example references the data type ("money") itself, which helps in limiting the set of possible measures. As the aggregate is not explicitly specified in the example, we use the sum over all values. Other possible aggregations include the average, minimum and maximum value, the median and the geometric mean.

5.3 DATA CUBE OPERATIONS

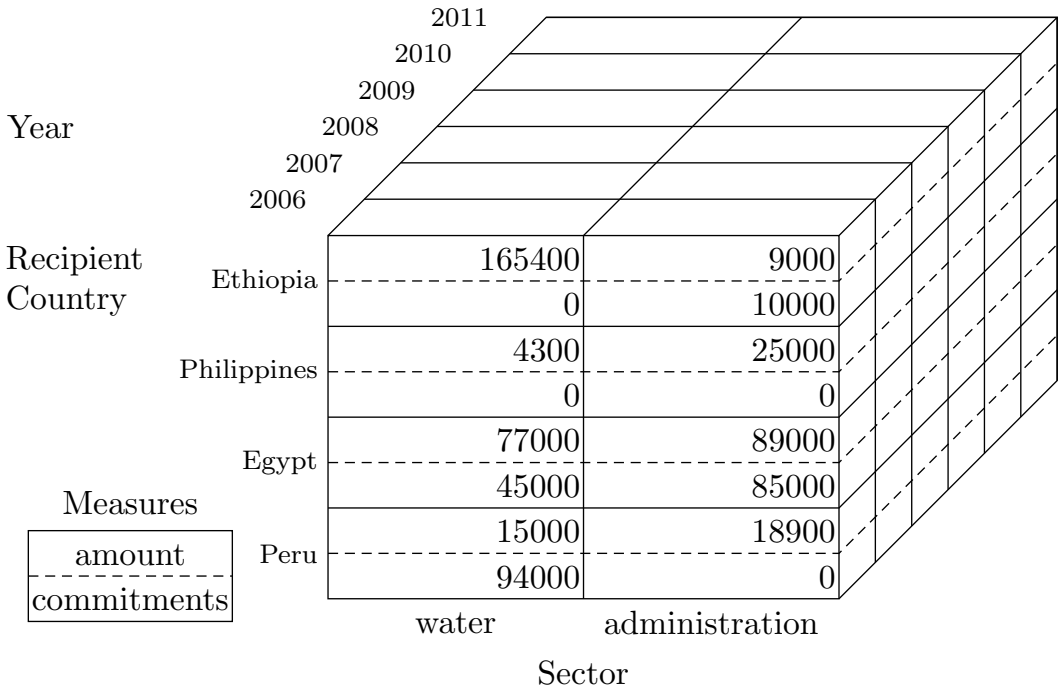

Figure 5.1: Simplified excerpt of the LinkedSpending (see Chapter 6) RDC *Finland Aid Data*. Measure units are provided by the *currency* attribute in each cell (omitted for brevity).

Figure 5.1 gives an simplified example of an RDC before any operations are applied. The RDC has the three dimensions *year*, *sector* and *recipient country* and the two measures *amount* and *commitments*. To answer the running example question *How much did the Philippines receive in the years of 2007 to 2008?*, the following operations are needed:

1. *Dicing* a data cube creates a subcube by constraining a dimension to a subset of its values. Figure 5.2 shows the result of a dice of Figure 5.1 on the *year* values of *2007* and *2008*.

2. *Slicing* a data cube reduces its dimensionality by one by constraining a dimension to one specific value, see Fig. 5.4. Figure 5.4 shows the result of a slice of Figure 5.2 on the *recipient country* of *the Philippines*.

3. *Rolling Up* a data cube means summarizing measure values along a dimension, such as a sum, count, or arithmetic mean. The sum of all *amount* values of Figure 5.4 is a roll-up that answers our question.

   Figure 5.5 shows the combination of those operations in a single SPARQL query.

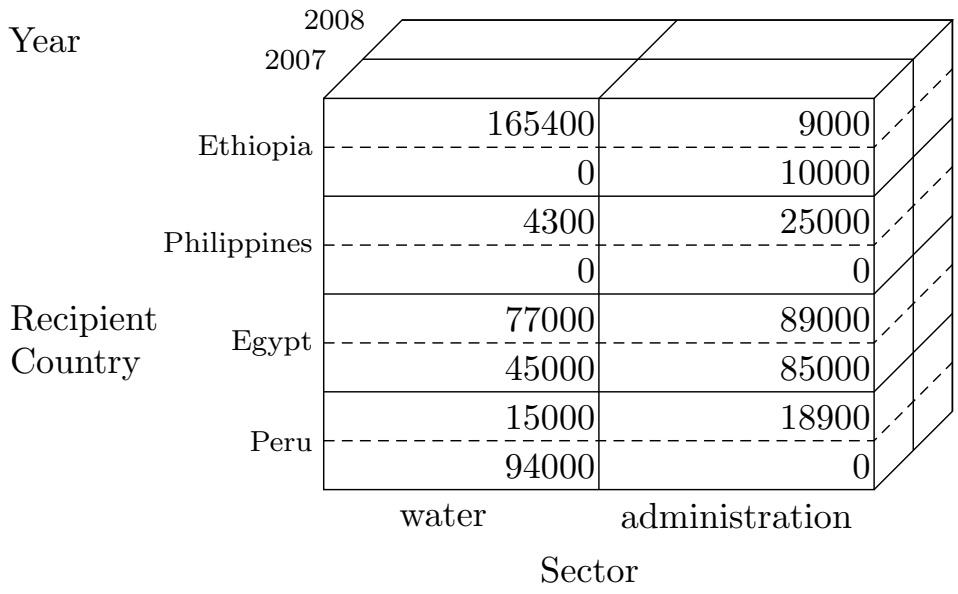

Figure 5.2: A *dice* of Fig. 5.1 created by constraining the *year* dimension to 2007 and 2008.

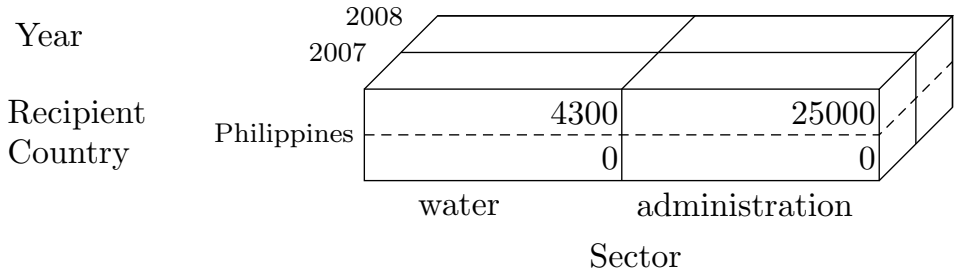

Figure 5.3: A *slice* of Fig. 5.2 created by constraining the *Recipient Country* dimension to the Philippines.

Figure 5.4: Example of a data cube and its operations.

```
SELECT sum(xsd:decimal(?amount))
FROM <http://linkedspending.aksw.org/finland-aid>
{
 ?o a qb:Observation.
 ?o lso:finland-aid-amount ?amount.
 ?o lso:finland-aid-recipient-country
       <https://openspending.org/finland-aid/recipient-country/ph>.
 ?o lso:refYear ?y.
 filter(func{year}{?y}=2007 OR func{year}{?y}=2008)
}
```

Figure 5.5: A SPARQL query for answering the question "How much did the Philippines receive in the years of 2007 to 2008?"

5.4 ALGORITHM

Basic facts can be queried using SPARQL queries as follows:

```
SELECT ?y WHERE {<RESOURCE> <PROPERTY> ?y.}
```

RDF Data Cubes, however, are based on observations and thus need a different query structure, such as:

```
SELECT ?o WHERE
{
 ?o a qb:Observation.
 ?o qb:dataSet ?d.
 ?d qb:structure ?model.
 ?model qb:component <DIMENSION>.
}
```

Such a single query is not enough to answer typical RDCQA questions, however, because their answers need to be derived from many observations and the answer can be multidimensional. To account for these requirements, the CubeQA algorithm uses the following pipeline architecture, see Fig. 5.6: First, the preprocessing step indexes the target data sets, extracts simple constraints and creates the parse tree used by the following steps. Figure 5.7 presents the parse tree for the running example question. Next, the matching step recursively traverses the parse tree downwards until it identifies reference candidates at each branch. Starting at those candidates, the combination step merges those candidates upwards until it creates a final template in the root of the parse tree. Finally, in the execution step, the template is converted to a SPARQL query that is executed to generate the result set containing the answer. In the following, those steps are described in detail:

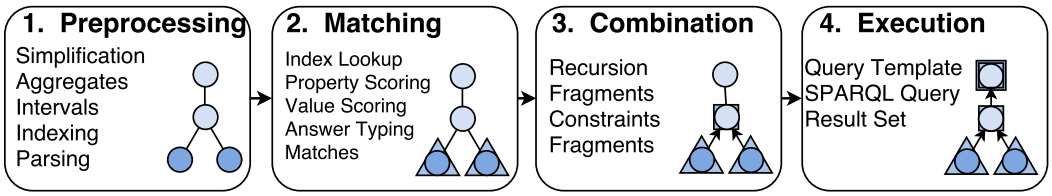

Figure 5.6: The CubeQA pipeline.

5.4.1 *Preprocessing*

NUMBER NORMALIZATION. First, numbers are normalized by multiplying them with factors assigned to textual references to numbers, such as "hundred" and "thousand", if they exist. This is necessary because the other components do not recognize textual references to numbers.

KEYPHRASE DETECTION. In this step, phrases referring to data cube operations (slice, dice and roll-up) are detected. These operations are typically referenced by certain keyphrases and are thus detected using regular expressions during

Element	SPARQL	Example Phrase
dice	filter	2007 to 2008
roll-up	aggregate	in total
slice	filter	in 2008
modifier	ORDER LIMIT	the 5 highest amounts

Table 5.5: Data cube operations extracted in the preprocessing step.

preprocessing, see Table 5.5. Those keyphrases, as well as the following entity recognition and disambiguation steps, are domain independent, so that CubeQA can be used with any set of RDCs. Nevertheless, the keyphrase to operation mapping can be extended for a specialized vocabulary to increase the recall on a particular domain.

If a roll-up is not explicitly expressed in the question, a default aggregation is assumed for some answer types. A SPARQL aggregate rolls up all dimensions that are not bound by query variables. The roll-up aggregates *sum*, *arithmetic mean* and *count* are handled differently then *minimum* and *maximum*. The former aggregates return new values, so that they can be mapped to the SPARQL aggregation keywords SUM, AVG and COUNT. Minimum and maximum, however, choose a value among the existing ones, which allows identification of a cell and thus of a different component value. For example, in "Which company has the highest research spending?", the user probably asks for the total, which can be achieved by a roll-up with addition followed by selecting the company with the highest sum. On the other hand, in "Which athlete jumped the highest?", the user implies the highest *maximum* jump height, not a sum of all jump heights. This is obvious to humans but not to machines, as domain knowledge is needed for that decision. Thus, it is hard to determine automatically, which aggregates make sense in a certain context. Sometimes, even a combination of multiple aggregates is needed. Because of both the difficulty and significance of this task, we propose implied aggregations as a good candidate for future research.

DATASET DETECTION CubeQA uses a data set index that is initialized once with the set of available RDCs. It is implemented as a Lucene index with fields for the labels, comments and property labels of each RDC. The title of the data set is not sufficient because questions often do not have words in common with that title. For example, in Fig. 5.5, the data set titled *finland-aid* is not mentioned but "the Philippines", "2007" and "2008" will all be found by the index.

PARSING At the end of the preprocessing step, a syntactic parse tree is generated for the modified question. This tree structure is traversed for matching nodes as described in Section 5.4.2.

Range	Scorer	Answer Type
`xsd:integer`	Numeric	countable
`xsd:float`, `xsd:double`	Numeric	uncountable
`xsd:gYear`, `xsd:date`	Temporal	temporal
no match	String	entity

Table 5.6: Component property scorer and answer type assignment. Integers include datatypes derived by constraint.

5.4.2 *Matching*

Our query model starts with the whole RDC. The question is then split into phrases that are mapped to constraints, which exclude cells from the target datacube. To increase accuracy and resolve ambiguity, however, phrases of the question are first mapped to potential observation values in the matching step, based on the following definitions, building upon Chapter 2:

Definition 5 (Values) *The* values *of a component property p for an RDC c and a knowledge base k are defined as:*
$V_{p,c,k} = \left\{ v | \exists o : \{(o, p, v), (o, \texttt{qb:dataset}, c), (o, \texttt{rdf:type}, \texttt{qb:Observation})\} \subset k \right\}$.
The numerical values $D_{p,c,k}$ *are the values representing numbers, converted to double values. The* temporal values $T_{p,c,k}$ *are the union of all values representing dates or years, converted to time intervals $\tau(v)$:* $T_{p,c,k} = \bigcup_{v \in V_{p,c,k}} \tau(v)$.

Example: $\texttt{:ph} \in V_{\texttt{:recipient-country,finland-aid,linkedspending}}$.

Definition 6 (Scorer) *A* scorer *for a component property p is represented formally by a partial function $d_p : \Sigma^* \to (L \cup U) \times (0, 1]$, Table 5.7 shows the three types of scorers, which are assigned to a property based on its range (see Table 5.6). Informally, the scorer of p returns the value with the closest match to a given phrase and its estimated probability.*

For the query template, the scorer results are converted to constraints. A naive approach is to create a value reference for the highest scored property of a phrase but this penalizes short phrases and suffers from ambiguity as it does not take the context of the phrase into account. Accordingly, CubeQA inserts an intermediate step: the *match*, which represents the possible references to component properties and their values.

Definition 7 (Match) *A* match *m is represented formally by a pair (ρ, γ), where ρ is the partial* property scoring function, *$\rho : P \to (0, 1]$ and γ is the partial* value scoring function, *$\gamma : P \to (L \cup U) \times (0, 1]$.*

5.4.3 *Combining Matches to Constraints*

The recursive combination process is used because (1) it favours longer phrases over shorter ones, giving increased coverage of the question and (2) it favours combination of phrases that are nearby in the question parse tree.

Type	Scoring Function $d_p(a)$
String	$\Big(\underset{b\in\beta_p(a)}{\operatorname{argmax}}\big(\operatorname{ngram}(a,b)\big), \underset{b\in\beta_p(a)}{\max}\big(\operatorname{ngram}(a,b)\big)\Big)$,
	$\beta_p(a) := \{b \in V_{p,c,k} \mid \operatorname{lev}(a,b) \leq 2\}$
	using bigram similarity [117] and a Levenshtein-Automaton[164] (lev).
Numeric	$(a,1)$, if $a \in \big[\min(D_{p,c,k}), \max(D_{p,c,k})\big]$, otherwise undefined
Temporal	$(a,1)$, if $T_{p,c,k} \cap \tau(a) \neq \varnothing$, otherwise undefined,

Table 5.7: Definitions of the different types of scorers. The String Scorer uses both the Levenshtein distance, to quickly find candidates, and bigrams, for a more accurate scoring. All three scorers are partial functions whose result is undefined if no value is found. Only String Scorers return scores < 1, as they can correct for typographical errors in the input, while Numeric and Temporal Scorers are either undefined or return the input number, respectively time interval, with a score of 1. Type casting and conversion is omitted for brevity, e.g. in Fig. 5.5 the phrase "Philippines" is equated to the language tagged label `"Philippines"@en` and the phrase "2007" to the year `2007^^xsd:gYear`.

Definition 8 (Constraint) *A* constraint *c is represented by a tuple (G, ω, λ), where:*

- *G is a set of SPARQL* triple patterns *and* filters *as defined in [8]*
- *ω is an optional* order by *modifier, $\omega \in \big(\{\text{ASC}, \text{DESC}\} \times P\big) \cup \{null\}$*
- *λ is an optional* limit *modifier, $\lambda \in \mathbb{N}^+ \cup \{null\}$*

Constraints are based on three different criteria:

1. A **Value Constraint** can be applied to any component property to confine it to an exact value, which can be a string, a number or a URI. It only contains triple patterns:
   $c_v = \Big(\big\{(?o, p, v), (?o, \texttt{qb:DataSet}, d), (?o, a, \texttt{qb:Observation})\big\}, \text{null}, \text{null}\Big)$,
   with $p \in P$ and $v \in L \cup U$.

2. An **Interval Constraint** confines a value to a numeric or temporal interval. Accordingly, it can only apply to a component property whose range is an XSD numeric or temporal data type. It consists of a SPARQL triple pattern and a filter:
   $$c_i = \Big(\big\{(?o, p, ?x), \text{filter}\big((?x > x_1) \texttt{ AND } (?x < x_2)\big)\big\}, \text{null}, \text{null}\Big),$$
   with $p \in P$, the lower limit x_1 and an upper limit x_2. Example:
   $\Big(\big\{(?o, :refYear, ?y), \text{filter}\big((?y \geq 2007) \texttt{ AND } (?y \leq 2008)\big)\big\}, \text{null}, \text{null}\Big)$.
   Closed or half-bounded intervals are defined analogously.

3. **Top/Bottom n Constraints** place an upper limit on the number of selected cells. They consist of three parts: The order (ascending or descending), the limit and the numeric component property whose values imply the order. Formally, $c_t = \Big(\varnothing, (\text{DESC}, p), n\Big)$, $c_b = \Big(\varnothing, (\text{ASC}, p), n\Big)$

To identify Value Constraints, each component property has a scorer (Definition 6), which tries to find a value similar to an input phrase. For example, "How much total aid was given to the regional FLEG programme in Mekong?", could refer to a dimension "programme" with a value of "FLEG" and a dimension "region" with a value of "Mekong". Equally possible would be a data set description of "aid to Mekong" and a dimension "target" with a value of "FLEG programme". The other types of constraints are matched in the preprocessing step because they are identified by certain keyphrases, such as "the 5 highest X".

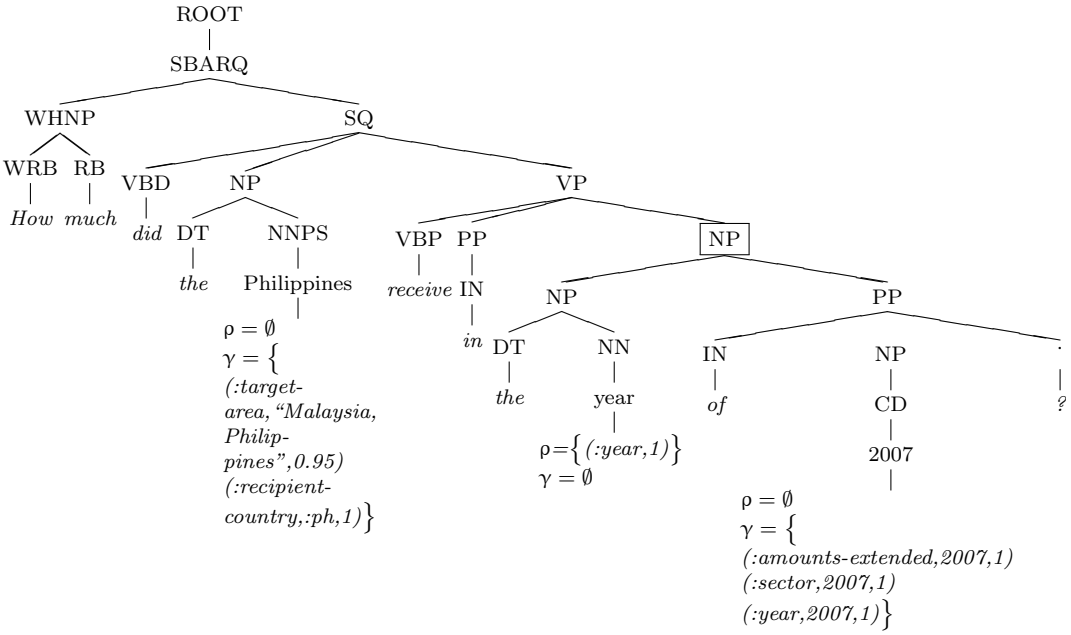

Figure 5.7: Syntactical parse tree for the question "How much did the Philippines receive in the year of 2007?" along with simplified property and value scoring functions ρ and γ, given in set notation.

In the example question, "How much did the Philippines receive in the year of 2007?", there are multiple candidates for the number "2007". The candidates can be disambiguated using the property scoring function of the "year" node by upward combination in the boxed "NP" node in Fig. 5.7. As a *match* only holds the information collected from a single node in the question parse tree, there is additional information needed to represent a whole subtree. This extended representation is called a *fragment* and holds: (1) multiple matches collected in the recursive merge and (2) constraints extracted from fitting matches.

Definition 9 (Fragment) *Formally, a* fragment f *is a pair* (M, R)*, where* M *is a set of matches (see Definition 7) and* R *is a set of constraints.*

Algorithm 1 describes the process that combines the fragments of a list of child nodes into the fragment for their parent node.

5.4.4 *Execution*

Algorithm 1 combines the fragments of child nodes to create a fragment for the parent node. When this recursive process reaches the root node, Algorithm 2

Algorithm 1: Fragment Combination.

Input: A list of fragments $F = \{(M_1, R_1), \ldots, (M_n, R_n)\}$, with $M_i = (\rho_i, \gamma_i)$
Output: The combined fragment $f = (M, R)$
$R \leftarrow \cup_{i=1}^{n} R_i$;
$P' \leftarrow \big(P \setminus \delta(R)\big) \cap \big(\bigcup_{i=1}^{n}(\mathrm{dom}(\rho_i)) \cup \big(\bigcup_{x \in \mathrm{dom}(\gamma_i)} \pi_1(x)\big)\big)$;
$M \leftarrow \{M_1, \ldots, M_n\}$;
foreach $p' \in P'$ **do**
  $m_{\text{property}} \leftarrow \underset{(\rho,\gamma)\in M'}{\mathrm{argmax}}\, \rho(p')$;
  $m_{\text{value}} \leftarrow \underset{(\rho,\gamma)\in M' \setminus \{m_{\text{property}}\}}{\mathrm{argmax}}\, \pi_2(\gamma(p'))$;
  $\gamma \leftarrow \pi_2(m_{\text{value}})$;
  $v \leftarrow \pi_1(\gamma(p))$;
  $g \leftarrow (?o, p, v)$;
  $R \leftarrow R \cup \{(\{g\}, \mathrm{null}, \mathrm{null})\}$;
  $M \leftarrow M \setminus \{m_{\text{property}}, m_{\text{value}}\}$;
return (M, R)

$\pi_i(t)$ is the projection on the i-th element of the tuple t. The domain $\mathrm{dom}(\mathrm{f})$ is the set of all elements for which the (partial) function f is defined. $\delta(R)$ is the set of all component properties that occur in at least one triple pattern in R.

transforms the fragment that results from the successive combination up to that point into a template (see Definition 10). All leftover value references whose property has not been referenced yet over a certain score threshold are transformed into additional Value Constraints. Other name and value references are discarded. All constraints, as well as the aggregate, if available, are then used to construct a SPARQL *select* query.

Definition 10 (Template) *A* template *t is a tuple (R, a, α), where R is defined in Definition 9, $a \in P$ is the answer property and α is an optional aggregate function, $\alpha(X) \in \Big\{\min(X), \max(X), \sum_{x\in X} x, |X|, \sum_{x\in X} \frac{x}{|X|}, null\Big\}$.*

Next, the values of the answer properties are requested. If the set of answer properties is empty, the default measure of the data set is used as an answer property to determine the properties. Executing the SPARQL query on the target knowledge base results in the set of answers requested by the user. The algorithm implementation is publicly available under an open license at https://github.com/AKSW/cubeqa.

Question Word	Expected Answer Type	f
what	uncountable, countable, count, temporal, location, entity	35
how much	uncountable	33
which	temporal, location, entity	19
how many	countable, count	6
when	temporal	4
none, other	uncountable, countable, count, temporal, location, entity	3
total		100

Table 5.8: Mapping m of a question word to a set of expected answer types E, along with the frequency of each question word in the benchmark. When the question word is unknown or not found, or the unspecific "what" is used, all 6 answer types are possible.

Algorithm 2: Fragment to Template Conversion.

Input: A fragment $f = (M, R)$
A threshold $\theta \in (0, 1]$
An optional aggregate function α identified in preprocessing
The set of expected answer types E is defined in Table 5.8
answerType(p) is defined by Table 5.6
Output: A template $t = (R', a, \alpha')$
$R' = R$;
$P' \leftarrow \big(P \setminus A \setminus \delta(R)\big)$;
foreach $(\rho, \gamma) \in M'$ **do**
  $p_{\max} = \underset{p \in (\mathrm{dom}(\gamma) \cap P')}{\mathrm{argmax}} \Big(\pi_2\big(\gamma(p)\big)\Big)$;
  if $\Big(\big(p_{max} \neq null\big) \wedge \big(\pi_2(\gamma(p_{max})) \leq \theta\big)\Big)$ **then**
    $R' \leftarrow R' \cup \Big(\big\{(?o, p_{\max}, \pi_1(\gamma(p_{\max})))\big\}, \mathrm{null}, \mathrm{null}\Big)$;
$A \leftarrow \bigcup_{(\rho,\gamma) \in M)} \mathrm{dom}(\rho)$;
$A' \leftarrow \{p \in A | \mathrm{answerType}(p) \in E\}$;
if $A' = \varnothing$ **then**
  $a \leftarrow$ DEFAULT_MEASURE;
else
  $a \leftarrow \underset{(\rho,\gamma) \in M, p \in A'}{\mathrm{argmax}} \rho(p)$;
return (R', a, α)

6 LINKEDSPENDING: OPENSPENDING BECOMES LINKED OPEN DATA

In this chapter, we present the conversion of OpenSpending *to* LinkedSpending, *which provides more than five million financial transactions in more than 600 data sets from all over the world as* RDF Data Cubes. *This chapter is based on Höffner et al. [105], which is published in the Semantic Web Journal. The evaluation is based on Höffner et al. [104], which is published in the proceedings of the International Semantic Web Conference of 2016. The author designed and implementated the conversion algorithm, wrote most of the paper and published the resulting data sets.*

In this chapter we answer research question RQ3: *How can we transform a large amount of relevant data cubes to the RDF Data Cube vocabulary?*

The structure of this chapter is as follows:

- Section 6.1 motivates the choice of converting government spending data in general and OpenSpending in particular (RQ3.1).
- Section 6.2 describes OpenSpending, which is the source of the data, and its data model.
- Section 6.3 explains the target RDF Data Cube vocabulary and the transformation process to it.
- Section 6.4 describes, how and where the data set is published and in which way users can access the data.
- Section 6.5 gives an overall view of the data sets, gives details about the licence used and describes the data sets it is interlinked to.
- The last section discusses known shortcomings of the data sets and future work. The prefixes used throughout this publication are defined in Table 2.1.

In order to save space, prefixes are used even when technically incorrect, such as in `ls:berlin_de/model`

6.1 CHOICE OF SOURCE DATA

6.1.1 *Government Spending*

A World Wide Web Consortium (W3C) design issue [21] motivates making government data available online as Linked Data for three reasons:

1. "Increasing citizen awareness of government functions to enable greater accountability";
2. "Contributing valuable information about the world;" and
3. "Enabling the government, the country, and the world to function more efficiently."

Increasing the transparency of government spending specifically is in high demand from the public. For instance, in the survey publication [156], "Public access to records is crucial to the functioning government" was rated with a mean of 4.14 (1 = disagree completely, 5 = agree completely). Open spending data can reduce corruption by increasing accountability and strengthening democracy because

voters can make better informed decisions. Furthermore, an informed and trusting public also strengthens the government itself because it is more likely to commit to large projects (see [6] for details).

Several States and Unions are bound to financial transparency by law, such as the European Union[1] with its *Financial Transparency System (FTS)*[2] [131]. Government spending amounts are however often much higher than the sums ordinary people are used to dealing with but even for policy makers it is hard to understand whether a certain amount of money spent is too high or normal. Comparing data sets and finding those which are similar to another one helps separating common values from outliers which should be further investigated. For example, if another country has a similar budget structure but spends way less on health care with a similar health level, it should be investigated whether that discrepancy is caused by inherent differences such as different minimum wages or a different climate or if it is due to preventable factors such as inefficiencies or corruption. Public spending services satisfy basic information needs, but in their current form they do not allow queries, which go further than simple keyword search or which cannot be answered with data from one system alone. Converting this Data to RDCs solves those problems by providing a unified format, a powerful query language and the possibility of integration with Linked Data sets from other services.

6.1.2 *OpenSpending*

OpenSpending.org is an open platform that provides public finance data from governments around the world. OpenSpending provides several hundreds of data sets, which can be searched, and it allows browsing and visualization of any single one, but it does not provide a comparison function between data sets. Because of the mechanism to identify equivalent properties (see Section 6.3), SPARQL queries can compare different data sets, e.g. between similar structures in different countries. Query 9 in Table 6.4 shows a simple query to detect data sets which are most similar to any particular data set. This is done by calculating the number of common measures, attributes and dimensions.

ECONOMIC ANALYSIS LinkedSpending is represented in Linked Open Data, which facilitates data integration. Currencies from DBpedia and countries from LinkedGeoData are already integrated. Financial data offers further integration candidates, such as political or other policy-influencing data such as health care. This allows queries such as query 7 in Table 6.4, which asks for data sets with currencies whose inflation rates are greater than 10 %.

LinkedSpending can also be used to compute economic indicators across several data sets. A possible indicator is a country's spending on education per person where the population size can be taken from the LinkedGeoData countries linked from one or more budget data sets. One such data set is `ugandabudget`, which contains the Uganda Budget and Aid to Uganda, 2003–2006. LinkedSpending serves as a hub for the integration of those data sets and their provenance information. More data sets can be integrated with similarity-based interlinking tools such as LIMES [145].

1 "2. The Commission shall make available, in an appropriate and timely manner, information on recipients, as well as the nature and purpose of the measure financed from the budget[...]" [162]

2 http://ec.europa.eu/budget/fts

6.2 OPENSPENDING SOURCE DATA

The domain model of OpenSpending is a data cube, where each cell corresponds to an instance of spending or revenue that contains as a measurement the amount of money spent or received.

OpenSpending[3] is a project which aims to track and analyze public spending worldwide. Data Sets can be submitted and modified by anyone but they have to pass a sanity check from the OpenSpending Data Team which also cleans the data before publishing.[4] OpenSpending hosts transactional as well as budgetary data with a focus on government finance. It contains this data in structured form stored in database tables and provides searching and filtering as well as visualizations and a JSON REST interface. The data sets differ in granularity and type of accompanying information, but they share the same meta model.

```
"main-programme":
{
 "label": "Main-programme",
 "type": "compound"
},
"sub-programme":
{
 "label": "Sub-programme",
 "type": "compound",
},
"amount":
{
 "datatype": "float",
 "label": "Total",
 "type": "measure",
}
```

Figure 6.1: Simplified excerpt of an OpenSpending *model*.

Figure 6.1 shows an excerpt from the model of the OpenSpending data set *eu-budget* with the dimension *sub-programme* and the measure *amount*. Figure 6.2 shows

3 https://openspending.org/
4 http://community.openspending.org/contribute/data/, accessed 2019-07-04

```
"main-programme":
{
 "name":"citizenship-freedom-security-and-justice"
},
"sub-programme":
{
 "name":"security-and-safeguarding-liberties"
},
"amount": 41.2
```

Figure 6.2: Simplified excerpt from an OpenSpending *entry*.

an entry that contains the actual values for the dimension and the measure of the observation.

PROBLEMS While the data is well-structured and thus suitable for conversion without data cleaning or extensive preprocessing, it still poses problems that need to be taken into account:

1. New data sets are frequently added (approximately 50 per month) and, less often, existing data sets are modified.
2. Some data sets do not specify a value for all properties in all observations.
3. There are properties with the same name in different data sets where it is unknown if they specify the same property.
4. Data Cube is a meta model. The deep structure of the data sets is heterogeneous and described only shallowly.
5. The language of literals is varying between and even within data sets but the language used is not specified.

Points 1 to 3 are addressed in the next section while points 4 and 5 are discussed in Section 7.4.

6.3 CONVERSION OF OPENSPENDING TO RDF

The RDF DataCube vocabulary [50], i.e. an RDF variant of the previously explained data cube model, is an ideal fit for the transformed data.

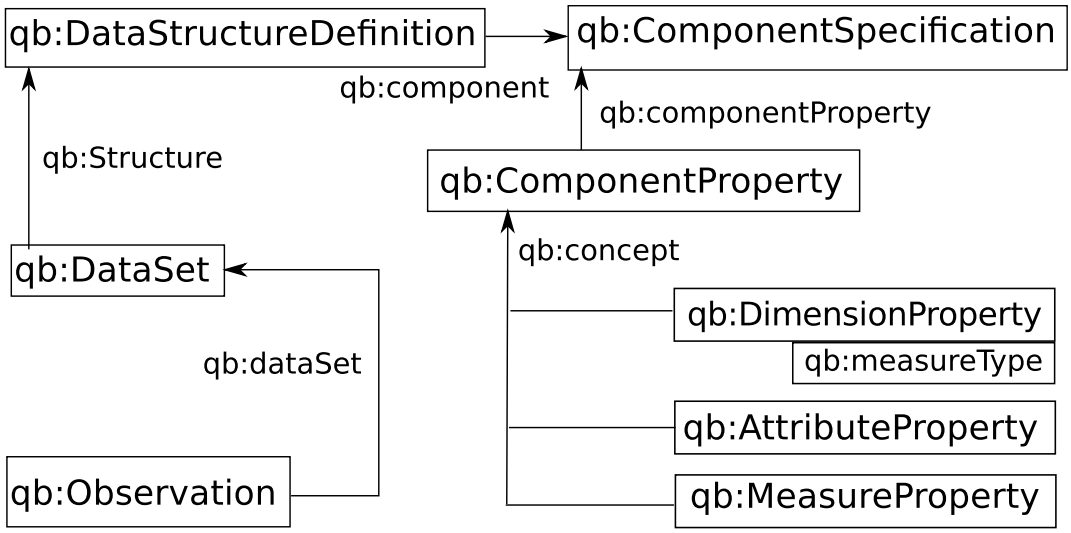

Figure 6.3: Used RDF DataCube concepts and their relationships[5].

First and foremost, this vocabulary provides the backbone structure for every LinkedSpending data set, see Fig. 6.3. Each data set is represented by an instance of `qb:DataSet` and an associated instance of `qb:DataStructureDefinition` which includes *component specifications* (see Fig. 6.4 for an example). Each component

5 Simplified version of the structure described in [50].

```
ls:berlin_de
 rdf:type qb:DataSet;
 rdfs:label "Berlin Budget";
 dc:source  os:berlin_de;
 qb:structure ls:berlin_de/model;
 qb:slice ls:berlin_de/views/nach-einzelplan.

ls:berlin_de/model
 rdf:type qb:DataStructureDefinition;
 qb:component
   lso:CountryComponentSpecification,
   lso:DateComponentSpecification,
   lso:Einzelplan-spec,
   lso:amount-spec.

lso:CountryComponentSpecification
 rdf:type qb:ComponentSpecification;
 rdfs:label "country";
 qb:attribute sdmxa:refArea;
 qb:componentRequired "false"^^xsd:boolean;
 qb:componentAttachment
   qb:DataSet,qb:Observation.
```

Figure 6.4: RDF DataCube vocabulary modelling excerpt of data set `ls:berlin_de` (some properties and values omitted).

Table 6.1: Conversion of OpenSpending to LinkedSpending classes and instances.
Each *class used by LinkedSpending* uses the data from the given *source URL* to create the instances of those classes.
In case there are multiple instances described at one URL, a *JSON path* expression is given, that locates the corresponding subnodes.
JSON path is a a query language for selecting nodes from a JSON document, similar to XPath for XML.
The *instance scheme describes* the form of the resulting LinkedSpending URLs. For example, the OpenSpending URL `os:berlin_de/model` contains the node `$.mapping.amount` which has a type value of "attribute" and is, thus, transformed to the OpenSpending instance `lso:berlin_de-amount` of the class `qb:AttributeProperty`.

	Source URL	JSON Path	LinkedSpending class	LS instance scheme
I	`os:`*name*`.json`		`qb:DataSet`	`ls:`*name*
II	`os:`*name*`/model`		`qb:DataStructureDefinition`	`ls:`*name*`/model`
III	`os:`*name*`/model`	`$.mapping.*`	`os:{Country,Time}Component Specification` or `qb:ComponentSpecification`	`lso:`*name-propertyname*`-spec`
IV	`os:`*name*`/model`	`$.mapping.*[?(@.type="compound")]`	`qb:DimensionProperty`	`lso:`*name-propertyname*
V		`$.mapping.*[?(@.type="date")]`	`qb:DimensionProperty`	
VI		`$.mapping.*[?(@.type="measure")]`	`qb:MeasureProperty`	
VII		`$.mapping.*[?(@.type="attribute")]`	`qb:AttributeProperty`	
VIII	`os:`*name*`/entries.json`	`$.results[*].dataset`	`qb:Observation`	`ls:observation-`*data set name-hashvalue*

specification is associated to a *component property* which can be either a *dimension*, an *attribute* or a *measure*. Commonly used concepts are specified in the model of the *Statistical Data and Metadata eXchange (SDMX)* initiative [39]. The RDF Data Cube vocabulary is supported by the LOD2 Statistical Office Workbench[6] which is part of the Linked Data Stack (an advanced version of the LOD2 Stack [12]). The workbench includes a DataCube validator, a split and merge component and a CKAN Publisher. The OntoWiki [11], which manages several parts of the Linked Data Lifecycle [12], such as Storage/Querying and Search/Browsing/Exploration offers a CSV import plugin for the format as well as a faceted RDF Data Cube browser, CubeViz. Data cubes may contain slices, which are presets for certain dimension values, effectively selecting a subset of a cube. Users may create and visualize their own slices using the OntoWiki CubeViz plugin. Furthermore, the RDF DataCube vocabulary allows the persistence of slices which is used to represent preconfigured slices from OpenSpending.

TRANSFORMATION All of the OpenSpending data sets describe observations referring to a specific point or period in time and thus undergo only minor changes. New data sets however, are frequently added. Because of this, the huge number of data sets and their size, an automatic, repeatable transformation is required. This is realized by a program[7] which fetches a list of data sets on execution and only transforms the ones who are not transformed yet. Each data set is transformed separately. Table 6.1 shows the exact rules of the mapping between OpenSpending URLs and JSON identifiers to LinkedSpending instances. Equivalent component properties (dimensions, attributes and measures) are identified as follows: A configuration file optionally specifies the mapping of data set and property name to an entity in the LinkedSpending ontology. By default, the property URI is derived from the property name. Properties with the same name in different data sets not having a mapping entry that states otherwise are assumed to represent the same concept and thus given the same URL.[8]

USE OF ESTABLISHED VOCABULARIES In addition to the standard vocabularies, RDF, RDF Schema (RDFS), OWL and XML Schema (XSD), the Dublin Core Metadata Initiative (DCMI) vocabulary is used for source and generation time metadata. The data sets are modelled, first and foremost, according to the RDF Data Cube vocabulary, which specifies the structure of a data cube. LinkedSpending follows the RDF Data Cube recommendation to make heavy use of the Statistical Data and Metadata eXchange (SDMX) model for measures, attributes and dimensions.The data sets are very heterogeneous but there are some properties which are commonly specified and thus modelled with established vocabularies. The year and date, a data set and an observation refers to, respectively, is expressed by `sdmxd:refPeriod` and XSD.

Currencies are taken from DBpedia [124] and countries are represented using the vocabulary of LinkedGeoData [177], a hub for spatial Linked Data. Some amount of data is imported from LinkedGeoData countries and DBpedia currencies. Because of the limited number of countries and currencies, and properties values imported per country and currency, the amount of data is too small to consider federated

6 http://demo.lod2.eu/lod2statworkbench

7 Written in Java, available as open source at https://github.com/AKSW/openspending2rdf.

8 Although that has the possibility of mismatches, such a mismatch has not been spotted yet. Still, evaluating and, if necessary, improving the automatic matching is part of future work.

querying. As most countries and currencies are stable in the medium term, this data needs to be updated only infrequently.

INTERLINKING There are two possibilities to align entities to another vocabulary: 1) to use the entities directly and 2) to create an own RDF resource with interlinks, like `owl:sameAs`, to that vocabulary. We generally preferred the first approach because a higher amount of reuse provides easier integration, better understandability and tool support.

While we did not find *sameAs* link targets on observation level, i.e. exactly the same observations described in other data sets, there are many possibilities for interlinks between data sets or dimension values and concepts they refer to. Using the labels of those data sets and dimension values, it is possible, for example, to link values of the dimension "region" of a federal budget, and thus indirectly also the observations which use those values, to the cities in DBpedia or LinkedGeoData whose labels are contained in the label of the region value URI.

ERROR HANDLING To prevent timeouts and to reduce the impact of disrupted connections, the source data set is downloaded in several parts with a maximum number of entries. These parts are then merged so that each file corresponds to exactly one data set. Data sets without observations are removed and the remaining data sets are transformed, noting the missing values for all component properties. If the first 1000 values are all missing, the transformation is aborted, otherwise a `lso:completeness` value $c = \frac{|\text{existing values}|}{|\text{observations}||\text{component properties}|}$ is attached to the data set. Besides empty or non-existing data sets, there were no other types of error observed. The chosen approach is to regard as equal all properties with exactly the same name.

SUSTAINABILITY The data conversion process is controlled by a web application[9], which constantly checks for added and modified data sets from OpenSpending, which are automatically queued for conversion but can also be manually managed. Updates do not interrupt the accessibility of the SPARQL endpoint and the services building on it. On average, about 50 new data sets became available on each month between September 2013 and March 2014. A service monitor constantly checks the state of the application and reports errors.

PERFORMANCE The transformation of a data set takes less than an hour on average on a 2 GHz virtual machine, using less then 2 GB of RAM.

6.4 PUBLISHING

The data is published using OntoWiki [11]. The interface for human and machine consumption is available at `http://linkedspending.aksw.org`. Depending on the actor and needs, OntoWiki provides various abilities to gather the published RDF data. It can be explored by viewing the properties of a resource, its values and by following links to other resources (see Figure 6.5).

9 `http://linkedspending.aksw.org/api`

Table 6.2: Technical details of the LinkedSpending data set.

URL	http://linkedspending.aksw.org
Version date and number	2013-8-14, 0.1
	2014-4-11, 2014-3
License	PDDL 1.0[11]
SPARQL endpoint	http://linkedspending.aksw.org/sparql
Compressed N-Triples Dump	http://linkedspending.aksw.org/extensions/page/page/export/lscomplete20143.tar.gz
DataHub entry	http://old.datahub.io/dataset/linkedspending
GitHub Repository	https://github.com/SmartDataAnalytics/openspending2rdf
Ontology	https://raw.githubusercontent.com/SmartDataAnalytics/openspending2rdf/master/schema/ontology.ttl

Using the SPARQL endpoint[12] provided by the underlying *Virtuoso Triple Store*[13], actors are able to satisfy complex information needs.

12 http://linkedspending.aksw.org/sparql
13 http://virtuoso.openlinksw.com

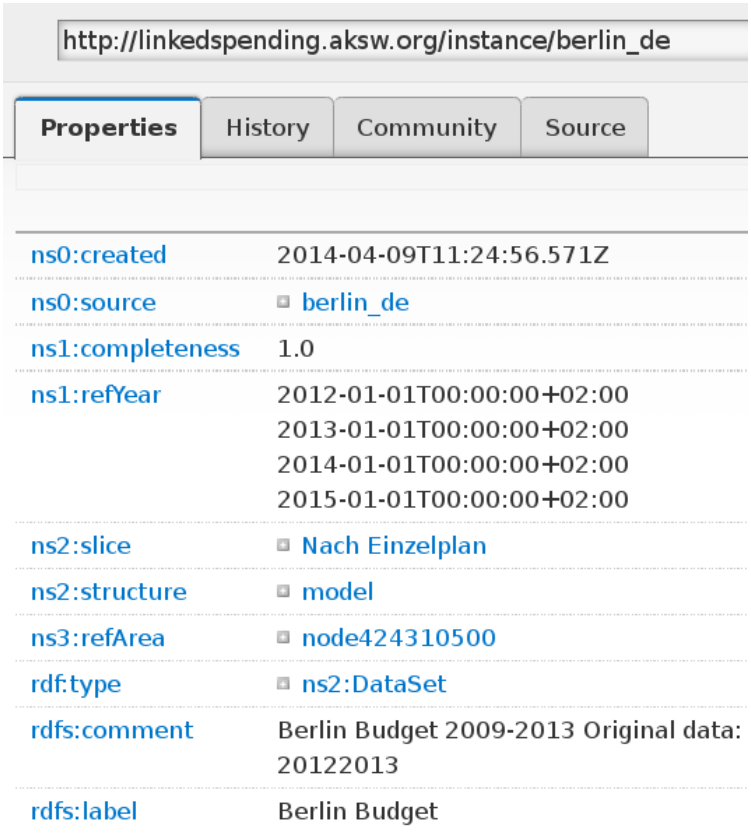

Figure 6.5: View of the data set `berlin_de` in the OntoWiki.

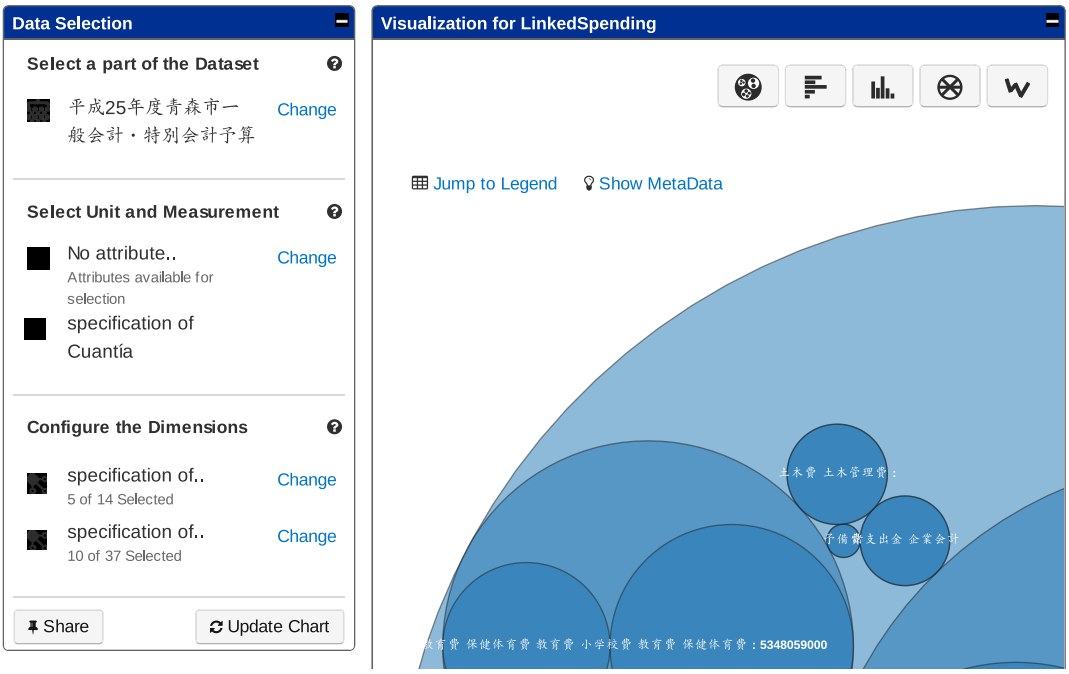

Figure 6.6: Faceted browsing in CubeViz by restricting values of dimensions.

Faceted search offers a selection of values for certain properties and thus slice and dice of the data set according to the interests on the fly. For example, depicted in Figure 6.6 is all Greek police spending in a certain region. Visualization supports discovery of underlying patterns and gain of new insights about the data, for example about the relative proportions of a budget (see Figure 6.7). We set up the RDF DataCube Browser CubeViz Salas et al. [163] as part of the human consumption interface.

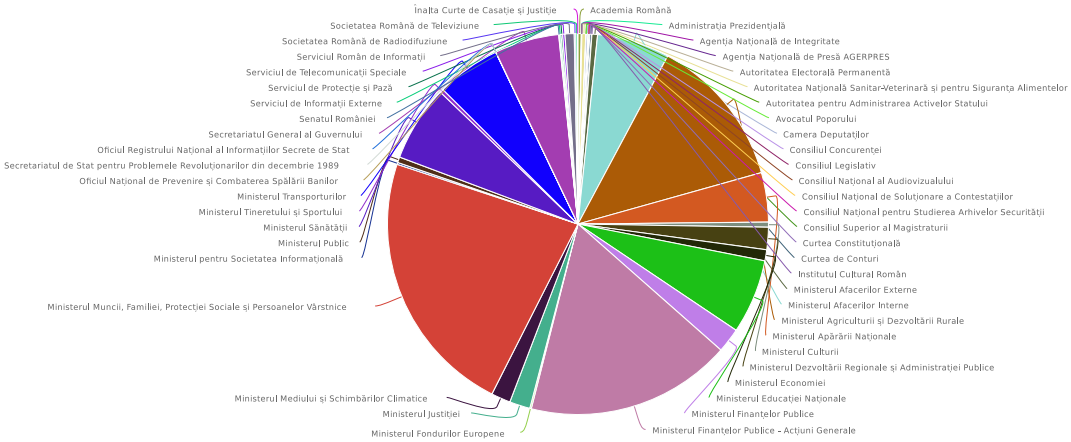

Figure 6.7: CubeViz visualization of the Romanian budget of 2013.

LICENSING All published data is openly licensed under the ODC Public Domain Dedication and Licence (PDDL) 1.0. in accordance with the open definition[14].

13 http://opendatacommons.org/licenses/pddl/1.0/
14 http://opendefinition.org/

6.5 OVERVIEW OVER THE DATA SETS

LinkedSpending consists of 955 data sets with more than five million observations total. The amount of observations of the individual data sets varies considerably between two (spendings in Prague of about 5000 CZK for an unknown purpose) and 242 209 ("Spending from ministries under the Danish government"). Table 6.3 details the average and total amount of data in bytes, triples, and observations as well as the number of links to external data sets, which, for the presented version of 2014-3, amounts to more than 9 million links to LinkedGeoData countries and 1.5 million links to DBpedia currencies.[15] Figure 6.8 shows the distribution of the numbers of measures, attributes and dimensions of the data sets.[16] Measures represent the quantity that an observation describes. All data sets have at least one measure which is the amount of money spent or received. For most of them (217) that is the only one but there are data sets with up to 7 measures. Attributes give further context to the measurement. The number of attributes is more varied, ranging from 2 to 26, with all data sets having at least a currency and a country, and most of them additionally the time the observations refer to. While the number of dimensions ranges from none[17] to 32, almost all of the data sets have between 1 and 6 dimensions, the most common ones being the year and the time the data set and the observations refers to, respectively. Technical details about the data sets are described in Table 6.2.

Table 6.3: Quantitative characteristics of version 2014-3. All values are rounded to the nearest integer.

Characteristic	Total	Average
number of data sets	955	
file size (N-Triples) in MB	24 585	39
triples	113 640 534	181 245
observations	5 026 393	8017
links to external data sets	10 696 614	17 060

EXAMPLE QUERIES Table 6.4 contains queries for common use cases: Queries 1–6 are basic queries. Query 7 uses the interlinking to DBpedia currencies by querying over two different graphs.[18] Query 8 uses the custom vocabulary[19] which is available for each data set.

15 The number of links is inflated as some are duplicated among observations instead of originating at data sets, because this allows better querying and tool support.

16 This analysis relates to version 0.1, which contains less data sets.

17 There is only one data set with no dimensions which a test data set on OpenSpending, as a data cube with no dimensions is not useful.

18 Parts of DBpedia and LinkedGeoData describing countries and currencies have been integrated in the SPARQL endpoint. With federated querying however, nearly the whole LOD cloud can be queried.

19 In this case, the "Hauptfunktion" and "Oberfunktion" are unique to the `berlin_de` data set.

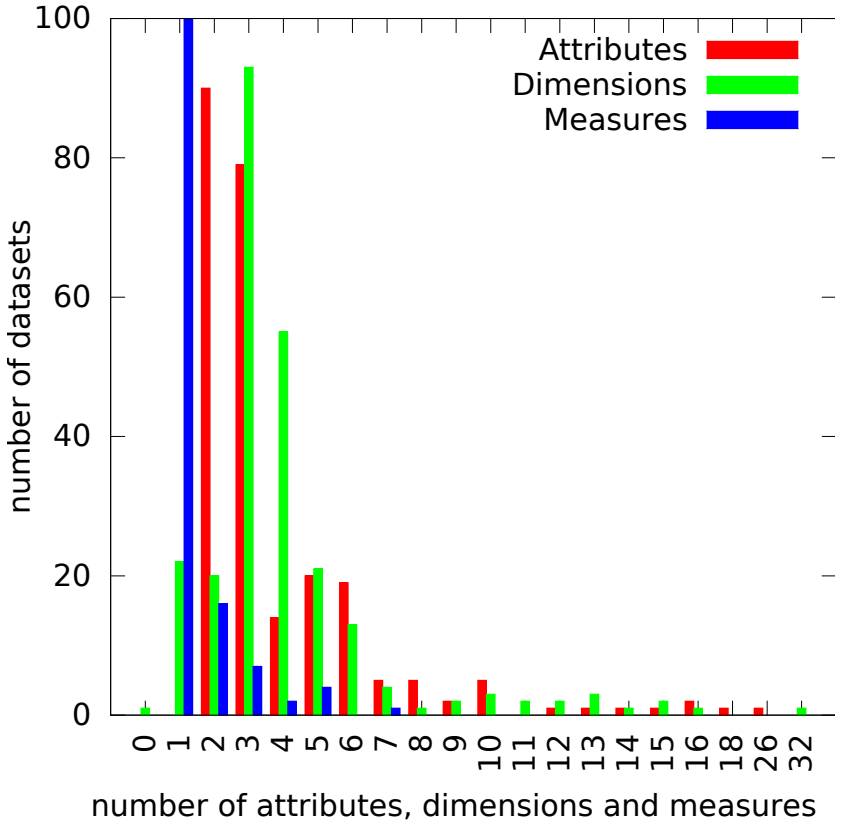

Figure 6.8: Histogram of measures, attributes and dimensions (version 0.1). 217 data sets have exactly one measure (clipped bar).

6.6 DATA SET QUALITY ANALYSIS

A recent data quality survey [215] identified a core set of data quality dimensions for Linked Data. Our analysis focusses on dimensions relevant to LinkedSpending. Security, for example, is an irrelevant indicator for the data sets because of the open license. Because spending data is historical, currency, volatility and timeliness are not applicable as well. Objectivity, accuracy, completeness, trustworthiness and trust are out of scope because they need to be evaluated on the source data.

6.6.1 *Intrinsic dimensions*

VALIDITY-OF-DOCUMENTS The validity of the RDF syntax has been successfully verified by RDF parsing and serializing utilities.

CONSISTENCY For each observation only one value per measure is generated, so that inconsistencies cannot occur.

CONCISENESS In RDF Data Cubes, attributes can be attached to a data set, a slice, a measure property or to an observation. While attributes which take different values for each observations necessarily have to be attached at the observation level, there are conflicting arguments for attributes which have the same value throughout. Attaching them at the observations ensures that all relevant information concerning

an observation is in one place. Thus users can satisfy their information needs with less complex SPARQL queries or by resolving a single URL. Attaching them at the data set however removes redundancy and thus improves the conciseness. Furthermore, queries which select certain data sets, are actually easier to formulate with the latter approach. However, it is easier to create mash-ups and to use RDF DataCube visualization tools such as CubeViz [163], when all attributes are attached at the observation level, so that we attach all attributes directly to the observations. RDF however is flexible and permits attaching the countries additionally at the data sets. Thus data sets of a certain country are simpler and the overhead in data size is negligible.

6.6.2 *Representational Dimensions*

VERSATILITY Multiple data access modes are available, see Table 6.2.

REPRESENTATIONAL CONCISENESS AND UNDERSTANDABILITY Because the source data contains labels for all data sets, attributes, dimensions and measures, those labels are also provided for the transformed data sets. The resource URLs[21] are concise and easily understandable, because they are generated from the names of the entities they describe. This facilitates writing SPARQL queries (see Table 6.4) and interpreting the results.

6.7 EVALUATION

6.7.1 *Experimental Setup and Benchmark*

As there was no existing benchmark for RDCQA, we created a benchmark based on a statistical question corpus and included it in the QALD-6[22] evaluation challenge.

QALD6T3-TRAIN We used the existing corpus and significantly extended it to 100 questions, forming the training set QALD6T3-train. While keeping a similar structure, we adapted it to 50 of the, at this time, 983 financial data sets of Linked-Spending [105]. Chosen are the first 50 data sets that are manually confirmed as English from a list of all data sets. The list was sorted in descending order by their proportion of English labels (having at least 100 labels) as determined by automatic language detection. The data sets contain in total 158 dimensions, 81 measures, 176 attributes, 950 149 observations and 16 359 532 triples.

QALD6T3-TEST Using the same 50 data sets, the test set QALD6T3-test was created in the same way, but with slightly less complex questions. The questions, correct SPARQL queries, correct answers and our evaluation results are available online.[23]

CHALLENGES QALD6T3 provides several challenges that are supported by the CubeQA algorithm. These are implied aggregations, intervals, implied or differently referenced measures and numerical values that are contained in several component

21 except the observations whose last part is a hash value
22 http://www.sc.cit-ec.uni-bielefeld.de/qald/
23 https://github.com/AKSW/cubeqa/blob/master/benchmark/

properties. It also includes questions that require features not provided by CubeQA, such as SPARQL subqueries. The performance of CubeQA on the benchmark is measured using precision (Definition 2), recall (Definition 3) and F_1-score (Definition 4). The arithmetic mean of the F_1 scores is calculated using $p = 0$ for empty answers.

RESULTS Of the 100 questions, 82 resulted in a nonempty answer, with an average precision of 0.401, a recall of 0.324 and an F_1 score of 0.392. Expected Answer Typing positively impacts the performance, as its removal results in a significant decrease in all three scores. Due to the cube index, many questions can be answered even if they do not specify their target data set. With all the 50 data sets as candidates, the performance drops even more than without using answer typing, but the index chooses the data set correctly for the majority of the questions (74 of 100). Answering the 100 questions on a PC with an Intel Core i5-3230M CPU, hosting both the SPARQL endpoint and the system implementation, took 87.45 s, 63.29 s and 63.33 s on three consecutive runs with preexisting index structures.[24] Table 6.5 shows the runtime distributions for the core tasks. Without preexisting index structures, the runs took 228.11 s, 228.77 s and 224.73 s, respectively.

6.7.2 *Discussion*

COMPARISON We believe that CubeQA will be a strong baseline in this new research subfield. As QALD6T3 was launched prior to submitting this publication to attract further research, two additional systems emerged: the yet unpublished QA^3 RDCQA system and the Sparklis [74] query builder (see Table 6.7). A query builder lets the user construct queries visually by selecting and combining SPARQL features and knowledge base resources. It enables users to create SPARQL queries and, if they build those queries correctly, achieves high accuracies. As such it occupies a middle ground, both in accuracy and usability, between RDCQA and manually creating SPARQL queries. QA^3 achieves a 9 % higher f-score than CubeQA but due to its purely template-base approach, it is unclear how it performs on open domain questions.

LIMITATIONS CubeQA does not support query structures that require SPARQL subqueries, express negations of facts or unions of concepts. Ambiguities and lexical gaps are hard challenges that are not solved yet [107]. Nevertheless, they occur in almost every question and must be adressed by every SQA system to avoid massive penalties to precision and recall. Table 6.5 categorizes the different errors that prevented CubeQA from returning a correct result to a question.

AMBIGUITY The most common cause is ambiguity, which mainly results from a high number of similar resources or equal numbers in the observation values. In benchmark question 86, “How much was budgeted for general services for the Office of the President of Sierra Leone in 2013?”, two different properties contain the literal "Office of the President". Because only the property value and not the property name is referenced, the algorithm cannot determine which property is correct. SQA systems like TBSL [186] resolve ambiguity by template scoring, so that the user chooses among the top n, where candidate combinations are ranked highest

24 The higher initial time is assumed to be caused by cache warmup both in the system and the SPARQL endpoint.

that maximize textual and semantic relatedness between the candidates [170]. But this approach is not applicable to RDCQA because of the RDC meta model, where component properties are not directly connected. Instead, CubeQA relies on references consisting of a name reference as well as a value reference, as in "the year 2008", where the name-value pair with the maximal score product of the name reference and the value reference is chosen. In case such a two-part reference does not occur, it is alleviated by giving temporal dimensions priority to others. For example, "2008" gets mapped to the year, if it exists, rather than the more improbable measurement value.

LEXICAL GAP The second most common cause is the lexical gap, where a reference could not be mapped to an entity due to the differences in surface forms. It is caused, among others, by different capitalization, typing errors, word transpositions ("extended amount", "amount extended") and different word forms ("committed", "commitments"). Another issue with the lexical gap is that a measurement can be referenced using a quantity reference ("amount"), a unit ("How many dollars are given"), or the type ("aid"), of which only the first one guarantees a match. Thus, CubeQA matches the range of a property as well as a its label. The RDC vocabulary provides `sdmx-attribute:unitMeasure` to specify units of measurement, but it does not support multiple measures so that the fallback has the same effect. In case of future vocabulary specification updates, we plan to integrate measurement units into our approach.

INDEXES All of those, except typing errors, occur in the benchmark. As these mentioned causes occur in document retrieval and Web search as well, full text indexes have been developed that robustly handle those problems. The Lucene index cannot overcome the lexical gap in some cases that are not recognized by the stemmer and where the edit distance is too large for the fuzzy index as well. Sometimes a concept is implicitly required but there is no explicit reference at all. Implicit references are part of future work and include aggregates.

Table 6.4: Exemplary SPARQL queries for typical use cases.

information need	SPARQL Query
1 list of all data sets	`SELECT * {?d a qb:DataSet}`
2 all measures of the data set `berlin_de`	`SELECT ?m { ls:berlin_de qb:structure ?s. ?s qb:component ?c. ?c qb:measure ?m.}`
3 all years which have observations in the de-bund data set from 2020 onwards	`SELECT distinct ?year {?o a qb:Observation. ?o qb:dataSet ls:de-bund. ?o lso:refYear ?year.FILTER (xsd:date(?year) >= "2020-1-1"^^xsd:date) }`
4 spendings of more than 100 billion €	`SELECT * {?o lso:amount ?a. ?o dbo:currency dbr:Euro. FILTER(xsd:integer(?a)>"1E11"^^xsd:integer) }`
5 data sets with multiple years	`SELECT ?d count(?y) as ?count { ?d a qb:DataSet. ?d lso:refYear ?y. } group by ?d having (count(?y)>1)`
6 sums of amounts for each reference year of `berlin_de`	`SELECT ?y (sum(xsd:integer(?amount)) as ?sum) {?o qb:dataSet ls:berlin_de. ?o lso:refYear ?y . ?o lso:amount ?amount.} group by ?y`
7 data sets with currencies whose inflation rate is $> 10\,\%$	`SELECT distinct ?d ?c ?r {?o qb:dataSet ?d. ?o dbo:currency ?c. ?c dbp:inflationRate ?r . filter(?r > 10)}`
8 Berlin city subsectors of research and education that have had their budget reduced from 2012 to 2013 (data set version 0.1)	`SELECT ?l (sum(xsd:integer(?amount12)) as ?sum12) (sum(xsd:integer(?amount13)) as ?sum13) {?o qb:dataSet ls:berlin_de. ?o lso:Hauptfunktion <http://openspending.org/berlin_de/Hauptfunktion/1>. ?o lso:Oberfunktion ?of. ?of rdfs:label ?l. {?o lso:refYear "2012"^^xsd:gYear. ?o lso:amount ?amount12.} UNION {?o lso:refYear "2013"^^xsd:gYear. ?o lso:amount ?amount13.}} group by ?l having (sum(xsd:integer(?amount12)) > sum(xsd:integer(?amount13)))`
9 data sets ordered by their number of properties in common with `2012_tax` (having at least one such common property)	`SELECT ?d (count(?c) as ?count) {ls:2012_tax qb:structure ?s. ?s qb:component ?c. ?d qb:structure ?s2. ?s2 qb:component ?c. FILTER(?d!=ls:2012_tax)} group by ?d order by desc(?count)`

Algorithm 3: JSON to RDF Transformation (see Table 6.1 for I–VIII).

```
m ← new JenaModel()
D ← list of data set names[20]
for each d ∈ D do
  E ← list of entries of d^VIII
  if |E| > 0 then
    create data set for d [I]
    create data set description for d^II
    for each property p_m in the mapping^III do
      switch (type(p_m))
      case "compound":
        P := P∪ new DimensionProperty()^IV
      case "date":
        P := P∪ new DimensionProperty()^V
      case "measure":
        P := P∪ new MeasureProperty()^VI
      case "attribute":
        P := P∪ new AttributeProperty()^VII
      end switch
    end for
    for each e ∈ E do
      o ← new Observation()
      for each property p in P do
        m.add(new Triple(o,p,e.get(p)))
        switch (type(p_m))
        case "compound":
          P := P∪ new DimensionProperty()
        case "date":
          P := P∪ new DimensionProperty()
        case "measure":
          P := P∪ new MeasureProperty()
        case "attribute":
          P := P∪ new AttributeProperty()
        end switch
      end for
    end for
  end if
end for
```

task	t (ms)
SPARQL querying	25 489
scoring	23 326
index lookup	16 070
parsing	5066
detectors	466
answer typing	13
total	61 673

Table 6.5: Runtimes of the core tasks on QALD6T3-train with preexisting cache structures. SPARQL querying, scoring and index lookup are intersecting and not all tasks are measured, so that the times do not add up to the total.

error cause	n
ambiguity	30
lexical gap	18
query structure	17
unknown	1
no error	34
total errors	66

Table 6.6: Categorization of errors of CubeQA on QALD6T3-train (at most one error per question), including the categories automatically excluded before the evaluation.

Algorithm	Benchmark	$\varnothing p$	$\varnothing r$	$\varnothing F_1$
CubeQA	train	0.40	0.32	0.32
QA^3	test	0.59	0.62	0.53
CubeQA	test	0.49	0.41	0.44
Sparklis	test	0.96	0.94	0.95

Table 6.7: QALD-6T3 test set performance [192], indicated by arithmetic mean of precision (counting no answers as precision of 0), recall, and F_1-score, rounded to 2 decimal places. CubeQA is also evaluated with the training set, which contains 100 harder questions compared to 50 in the test set. The correct target RDC was predefined for the training set, as the cube index is evaluated separately, with 74 of 100 correct choices by CubeQA.

7 CONCLUSION

7.1 RESEARCH QUESTION SUMMARY

A brief summary of the initial research questions is as follows:

RQ2.1: Typical user questions are collected in the corpus presented in Section 5.1. The full 50 questions are shown in appendix A.

RQ2.2: Most of the corpus questions expect countable[1] or uncountable quantities or counts. Other information needs include entities with the highest or lowest amount according to some criteria and comparisons, see Section 5.2.

RQ2.3: User questions can be transformed to SPARQL queries by matching parts of a parse tree of the questions to measures, values and component properties of the RDF Data Cube target data sets and by identifying the appropriate aggregations and order limit modifiers, see Section 5.4.

RQ2.4: The performance of a SQA system can be evaluated by creating a benchmark based on the corpus, see Section 6.7.

RQ2.5: CubeQA is sufficiently powerful to be applied on challenging questions over multidimensional data and we believe it will be a strong baseline for future research.

RQ2.6: Precision is higher than recall, similar to general SQA systems, on QALD6T3-train, but similar to the recall on QALD6T3-test.

RQ2.7: The three most common causes of problems for CubeQA on QALD6T3-train are ambiguity with errors on 30 questions, followed by the lexical gap with 18 and query structure errors on 17 of the 100 questions. Ambiguity errors are caused by an incorrect choice between multiple possible meanings of a natural language expression. Lexical gap errors occur when an entity is referred by a different phrase in the question than in the knowledge base and when methods to bridge this gap, such as sets of synonyms, cannot match this phrase as well. Query structure errors occur when grammatical constructs of the query represent operations, such as negation or subqueries, that the algorithm does not support.

RQ2.8: CubeQA achieves a precision of 0.49, a recall of 0.41 and a global F_1 score of 0.44 on the QALD6T3-test benchmark. On the more difficult QALD6T3-train benchmark, it achieves a precision of 0.40, a recall of 0.32 and a global F_1 score of 0.32. The QA^3 algorithm improves on those results with a precision of 0.59, a recall of 0.62 and a global F_1 score of 0.53, using a template-based algorithm. Those results are discussed in Section 6.7.2.

1 in the grammatical sense

7.2 SQA SURVEY

The survey in Chapter 4 analyzes 62 systems and their contributions to seven challenges for SQA systems. SQA is an active research field with many existing and diverse approaches covering a multitude of research challenges, domains and knowledge bases.

LIMITATIONS We only cover QA on the Semantic Web, that is, approaches that retrieve resources from RDF knowledge bases. As similar challenges are faced by QA unrelated to the Semantic Web, we refer to Section 3.1.1 and Section 3.1.2. We choose to not go into detail for approaches that do not retrieve resources from RDF knowledge bases. Moreover, our consensus can be found in Table 7.1 for best practices. To cover the field of SQA in depth, we exluded works solely about similarity [15] or paraphrases [19]. The existence of common SQA challenges implies that a unifying architecture can improve the precision as well as increase the number of answered questions [132]. Research into such architectures, includes openQA [132], OAQA [212], QALL-ME [71] and QANUS [144] (see Section 3.1.3). Our goal, however, is not to quantify submodule performance or interplay. That will be the task of upcoming projects of large consortiums. A new community[2] is forming in that field and did not find a satisfying solution yet.[3]

RESEARCH DRIFT Overall, the authors of this survey cannot observe a research drift to any of the challenges. The number of publications in a certain research challenge does not decrease significantly, which can be seen as an indicator that none of the challenges is solved yet—see Table 4.3. Naturally, since only a small number of publications addressed each challenge in a given year, one cannot draw statistically valid conclusions. The challenges proposed by Cimiano et al. [47] and reduced within this survey appear to be still valid. The following sections discuss each of the seven research challenges and give a short overview of already established as well as future research directions per challenge, see Table 7.1.

7.2.1 *Lexical Gap*

The lexical gap has to be bridged by every SQA system in order to retrieve results with a high recall. For named entities, this is commonly achieved using a combination of the reliable and mature natural language processing algorithms for string similarity and either stemming or lemmatization, see Table 7.1. AQE, for example with *WordNet* synonyms, is prevalent in information retrieval but only rarely used in SQA. Despite its potential negative effects on precision[4], we consider it a net benefit to SQA systems. Current SQA systems duplicate already existing efforts or fail to decide on the right technique. Thus, reusable libraries to lower the entrance effort to SQA systems are needed. Mapping to RDF properties from verb phrases is much harder, as they show more variation and often occur at multiple places of a question. Pattern libraries, such as BOA [86], can improve property identification, however they are still an active research topic and are specific to a knowledge base.

2 https://www.w3.org/community/nli/
3 http://eis.iai.uni-bonn.de/blog/2015/11/
4 Synonyms and other related words almost never have exactly the same meaning.

Table 7.1: Techniques for solving each challenge that are established and either actively researched or envisioned.

Challenge	Established	Future
Lexical Gap	stemming, lemmatization, string similarity, synonyms, vector space model, indexing, pattern libraries, explicit semantic analysis	combined efforts, reuse of libraries
Ambiguity	user information (history, time, location), underspecification, machine learning, spreading activation, semantic similarity, crowdsourcing, Markov Logic Network	holistic, knowledge-base aware systems
Multilingualism	translation to core language, language-dependent grammar	usage of multilingual knowledge bases
Complex Operators	reuse of former answers, syntactic tree-based formulation, answer type orientation, HMM, logic	non-factual questions, domain-independence
Distributed Knowledge and Procedural, Temporal, Spatial	temporal logic	domain specific adaptors, procedural SQA
Templates	fixed SPARQL templates, template generation, syntactic tree based generation	complex questions

7.2.2 *Ambiguity*

The next challenge, ambiguity, is addressed by the majority of the publications but the percentage does not increase over time, presumably because of use cases with small knowledge bases, where its impact is minuscule. For systems intended for longtime usage by the same persons, we regard as promising the integration of previous questions, time and location, as is already common in web of document search engines. There is a variety of established disambiguation methods, which use the context of a phrase to determine the most likely RDF resource, some of which are based on unstructured text collections and others on RDF resources. As we could make out no clear winner, we recommend system developers to make their decisions based on the resources (such as query logs, ontologies, thesauri) available to them. Many approaches reinvent disambiguation efforts and thus–like for the lexical gap–holistic, knowledge-base aware, reusable systems are needed to facilitate faster research.

7.2.3 *Multilingualism*

Despite its inclusion since QALD 3 [36] and following, publications dealing with multilingualism remain a small minority. Automatic translation of parts of or the whole query requires the least development effort, but suffers from imperfect translations. A higher quality can be achieved by using components, such as parsers and synonym libraries, for multiple languages. A possible future research direction is to make use of various language versions at once to use the power of a unified graph [48]. For instance, DBpedia [122] provides a knowledge base in more than 100 languages, which could form the base of a multilingual SQA system.

7.2.4 *Complex Operators*

Complex operators seem to be used only in specific tasks or factual questions. Most systems either use the syntactic structure of the question or some form of knowledge-base aware logic. Future research will be directed towards domain-independence as well as non-factual queries.

7.2.5 *Distributed Knowledge*

Systems using distributed knowledge remain niches. The common approach of the analyzed systems is to align the query or parts of it to several target knowledge bases and then to rank them.

7.2.6 *Procedural, Temporal and Spatial Data*

Procedural SQA does not exist yet as present approaches return unstructured text in the form of already written step-by-step instructions. While we consider future development of procedural SQA as feasible with the existing techniques, as far as we know there is no RDF vocabulary for and knowledge base with procedural knowledge yet.

7.2.7 *Templates*

The templates challenge which subsumes the question of mapping a question to a query structure is still unsolved. Although the development of template based approaches seems to have decreased in 2014, presumably because of their low flexibility on open domain tasks, this still presents the fastest way to develop a novel SQA system but the limitiation to simple query structures has yet to be overcome.

7.2.8 *Future Research*

Future research should be directed at more modularization, automatic reuse, self-wiring and encapsulated modules with their own benchmarks and evaluations. Thus, novel research directions can be tackled by reusing already existing parts and focusing on the core problem. A step in this direction is QANARY [32], which describes how to modularize QA systems by providing a core QA vocabulary

against which existing vocabularies are bound. Another research direction is SQA systems as aggregators or framework for other systems or algorithms to benefit of the set of existing approaches. Furthermore, benchmarking will move to single algorithmic modules instead of benchmarking a system as a whole. The target of local optimization is benchmarking a process at the individual steps, but global benchmarking is still needed to measure the impact of error propagation across the chain. A Turing-test-like spirit would suggests that the latter is more important, as the local measure are never fully representative. Additionally, we foresee the move from factual benchmarks over common sense knowledge to more domain specific questions without purely factual answers. Thus, there is a movement towards multilingual, multi-knowledge-source SQA systems that are capable of understanding noisy, human natural language input.

7.3 CUBEQA

We introduce RDCQA and design the CubeQA algorithm, provide a benchmark based on real data, and evaluate the results. In future work, we plan to continue contributing to the yearly QALD evaluation campaign by providing progressively more challenging benchmarks. The next iteration of CubeQA will answer questions that require the consolidation of several RDCs. We will also investigate how to integrate RDCQA techniques with SQA frameworks, such as OpenQA [132], so that all-purpose systems can also answer questions on RDCs. On the flipside, we also plan to integrate general SQA into RDCQA, to answer questions on RDCs that require world knowledge.

We also identified the following improvements:

- Implement selection filters as logical formula of constraints instead of flat sets, including negations and unions.
- Support SPARQL subqueries to handle nested information dependencies.
- Support languages other than English using language detection components as well as fitting parsers, indexes and preprocessing templates.
- Add synonym handling through the lexical database *WordNet* [137].
- Incorporate measurement units if the RDC vocabulary adds support for them for multiple measures. For elaborate phrase patterns, like "How many people live in" for "population", there are pattern libraries like BOA [86] which need to be adapted to RDCs by retraining on a comprehensive question corpus.

Overall, we believe to have opened a novel research subfield within SQA, which will increase in importance due to the rise of both the volume of RDCs and the usage of QA approaches in everyday life.

7.4 LINKEDSPENDING

As shown in Section 6.3, we converted several hundreds of financial data sets to RDF and, as shown in Section 6.4, we published them as Linked Open Data in several ways. However, we recognise a few shortcomings and our goal is to enrich the meta data with the help of domain experts and to refine the structure of the individual data sets. Furthermore, we plan to improve the automatic configuration of CubeViz.

7.4.1 *Shortcomings*

MULTILINGUALITY RDF itself provides support for multilingualism, which is one of its key advantages to other representation formats. The source data does not contain language tags, however, and the languages used do not always match the country that the data refers to. Automatic language detection on single labels did not yield a satisfying precision and it is not possible to increase the precision of the language detection by combining the estimates about several labels of an observation as their language is not always identical. We plan statistical examinations of the relations between labels of different entities and more complex schemes based on those examinations, which can achieve language detection with a higher precision. Additionally, we plan to automatically translate all literals to several languages.

INDIVIDUAL MODELLING Because the source data is already structured, the transformation of all the data sets without the need of text extraction and in an automatic way was feasible. On a deep level however, there is much unmodelled structure that is unique to each data set or at most shared between several of them, for instance the categorization of spending into several specific "plans" in German budgets. Because of the amount of data sets, modelling all details, and thus also improving the internal and external connectivity, requires either a large-scale cooperation or a crowd-driven approach, which we did not perform.

DRILLDOWNS Because of the hierarchical organization of the different coded properties "groups" and "functions", the visualizations on `openspending.org` permit "zooming" (drilldown) in and out of the different levels of the data. The RDF Data Cube vocabulary mention the use of *concept schemes* or *hierarchical code lists* but neither variant is fully specified yet and it is not clear, which of those modelling possibilities will become standard. Thus, LinkedSpending does not contain hierarchical code lists and does not support drilldown.

7.4.2 *Future Work*

INTERLINKING Extensive interlinking of referenced entities to the all-purpose knowledge base of DBpedia provides additional context. Coded property values, such as the budget areas healthcare and public transportation, can be interlinked with their respective DBpedia concepts. This enables the usage of type hierarchies and thus new ways of structuring the data and provides more meaningful aggregations and new insights.

REUSE Because of the large amount of data sets, the conversion process is necessarily automatic. While for some of the often used component properties, like date and time, resources from existing vocabularies such as SDMX are used, in the general case, new resources are created for all values. A thorough manual inspection might yield in an increased usage of existing vocabulary. Also some of the dimensions may be stable over different data sets and thus might be shared, which decreases the amount of redundancy and increases the benefit of those resources, should they be referenced from other resources.

BIBLIOGRAPHY

[1] Peter Adolphs, Martin Theobald, Ulrich Schäfer, Hans Uszkoreit, and Gerhard Weikum. "YAGO-QA: Answering Questions by Structured Knowledge Queries." In: *ICSC 2011 IEEE Fifth International Conference on Semantic Computing*. doi:10.1109/ICSC.2011.30. Los Alamitos, USA: IEEE Computer Society, 2011, pp. 158–161.

[2] Nitish Aggarwal, Tamara Polajnar, and Paul Buitelaar. "Cross-Lingual Natural Language Querying Over the Web of Data." In: *18th International Conference on Applications of Natural Language to Information Systems, NLDB 2013*. Ed. by Elisabeth Métais, Farid Meziane, Mohamad Saraee Vijayan Sugumaran, and Sunil Vadera. doi:10.1007/978-3-642-38824-8_13. Berlin Heidelberg, Germany: Springer, 2013, pp. 152–163.

[3] Eugene Agichtein, David Carmel, Dan Pelleg, Yuval Pinter, and Donna Harman. "Overview of the TREC 2015 LiveQA Track." In: *The Twenty-Fourth Text REtrieval Conference (TREC 2015) Proceedings*. Ed. by Ellen M. Voorhees and Angela Ellis. National Institute of Standards and Technology (NIST), 2015.

[4] Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, eds. *The Semantic Web–ISWC 2013*. doi:10.1007/978-3-642-41335-3. Berlin Heidelberg, Germany: Springer, 2013.

[5] James F Allen. "Maintaining knowledge about temporal intervals." In: *Communications of the ACM* 26 (1983). doi:10.1145/182.358434, pp. 832–843.

[6] James E. Alt, David Dreyer Lassen, and David Skilling. *Fiscal Transparency, Gubernatorial Popularity, and the Scale of Government: Evidence from the States*. Tech. rep. Economic Policy Research Unit (EPRU), University of Copenhagen, 2001. URL: http://ideas.repec.org/p/kud/epruwp/01-16.html.

[7] G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, P. de Leenheer, and J. Z. Pan, eds. *The Semantic Web: Research and applications*. Vol. 6643. Lecture Notes in Computer Science. doi:10.1007/978-3-642-21034-1. Berlin Heidelberg, Germany: Springer, 2011.

[8] Carlos Buil Aranda et al. *SPARQL 1.1 Overview*. W3C Recommendation. W3C, 2013. URL: http://www.w3.org/TR/2013/REC-sparql11-overview-20130321/.

[9] Lora Aroyo, Harith Welty Chris Alani, Jamie Taylor, Abraham Bernstein, Lalana Kagal, Natasha Noy, and Eva Blomqvist, eds. *The Semantic Web–ISWC 2011*. doi:10.1007/978-3-642-25073-6. Berlin Heidelberg, Germany: Springer, 2011.

[10] Sofia Athenikos and Hyoil Han. "Biomedical Question Answering: A Survey." In: *Computer methods and programs in biomedicine* 99 (2010), pp. 1–24. DOI: 10.1016/j.cmpb.2009.10.003.

[11] Sören Auer, Sebastian Dietzold, Jens Lehmann, and Thomas Riechert. "OntoWiki: A Tool for Social, Semantic Collaboration." In: *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge (CKC 2007) at the 16th International World Wide Web Conference (WWW2007) Banff, Canada, May 8, 2007*. Ed. by Natalya Fridman Noy, Harith Alani, Gerd Stumme, Peter Mika, York Sure, and Denny Vrandecic. Vol. 273. CEUR Workshop Proceedings. CEUR-WS.org, 2007. URL: http://ceur-ws.org/Vol-273/paper_91.pdf.

[12] Sören Auer et al. "Managing the life-cycle of Linked Data with the LOD2 Stack." In: *Proceedings of International Semantic Web Conference (ISWC 2012)*. 22% acceptance rate. 2012. URL: http://iswc2012.semanticweb.org/sites/default/files/76500001.pdf.

[13] Georgios Balikas, Aris Kosmopoulos, Anastasia Krithara, Georgios Paliouras, and Ioannis Kakadiaris. "Results of the BioASQ Track of the Question Answering Lab at CLEF 2014." In: *CLEF 2014 Working Notes*. Vol. 1180. CEUR Workshop Proceedings. Online Working Notes. 2014.

[14] Georgios Balikas, Aris Kosmopoulos, Anastasia Krithara, Georgios Paliouras, and Ioannis Kakadiaris. "Results of the BioASQ tasks of the Question Answering Lab at CLEF 2015." In: *CEUR Workshop Proceedings*. Vol. 1391. CEUR Workshop Proceedings. Online Working Notes. 2015.

[15] Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. "UKP: Computing Semantic Textual Similarity by Combining Multiple Content Similarity Measures." In: **SEM 2012: The First Joint Conference on Lexical and Computational Semantics* (Montréal, Canada). Ed. by Suresh Manandhar and Deniz Yuret. Stroudsburg, USA: Association for Computational Linguistics, 2012, pp. 435–440.

[16] Hannah Bast, Florian Bäurle, Björn Buchhold, and Elmar Haussmann. "Broccoli: Semantic Full-Text Search at Your Fingertips." In: *arXiv preprint arXiv: 1207.2615* (2012).

[17] Petr Baudiš and Jan Šedivỳ. "Modeling of the Question Answering Task in the YodaQA System." In: *Experimental IR Meets Multilinguality, Multimodality, and Interaction*. doi:10.1007/978-3-319-24027-5_20. Berlin Heidelberg, Germany: Springer, 2015, pp. 222–228.

[18] Asma Ben Abacha and Pierre Zweigenbaum. "Medical Question Answering: Translating Medical Questions Into SPARQL Queries." In: *IHI '12: Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*. New York, USA: Association for Computing Machinery, 2012, pp. 41–50.

[19] Jonathan Berant and Percy Liang. "Semantic Parsing via Paraphrasing." In: *Proceedings of the 52nd Annual Meeting*. doi:10.3115/v1/P14-1133. Stroudsburg, USA: Association For Computational Linguistics, 2014, pp. 1415–1425.

[20] Tim Berners-Lee. *Linked Data—Design Issues*. W3C design issue. 2009. URL: http://www.w3.org/DesignIssues/LinkedData.html.

[21] Tim Berners-Lee. *Putting Government Data online—Design Issues*. W3C design issue. 2009. URL: http://www.w3.org/DesignIssues/GovData.html.

[22] Tim Berners-Lee, Roy T. Fielding, and Larry Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. Tech. rep. 3986. Internet Engineering Task Force (IETF), Network Working Group, Jan. 2005. URL: https://tools.ietf.org/html/rfc3986.

[23] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper San Francisco, 1999.

[24] Tim Berners-Lee, James Hendler, Ora Lassila, et al. "The Semantic Web." In: *Scientific American* 284.5 (2001), pp. 28–37.

[25] Tim Berners-Lee, Larry Masinter, and Mark McCahill. *Uniform Resource Locators (URL)*. Tech. rep. 1738. Internet Engineering Task Force (IETF), Network Working Group, Dec. 1994. URL: https://tools.ietf.org/html/rfc1738.

[26] Andrzej Białecki, Robert Muir, Grant Ingersoll, and Lucid Imagination. "Apache Lucene 4." In: *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*. Ed. by Andrew Trotman, Charles L. A. Clarke, Iadh Ounis, J. Shane Culpepper, Marc-Allen Cartright, and Shlomo Geva. 2012, p. 17.

[27] Veli Bicer, Thanh Tran, Andreas Abecker, and Radoslav Nedkov. "KOIOS: Utilizing semantic search for easy-access and visualization of structured environmental data." In: *The Semantic Web–ISWC 2011*. doi:10.1007/978-3-642-25093-4_1. Berlin Heidelberg, Germany: Springer, 2011, pp. 1–16.

[28] Chris Biemann, Siegfried Handschuh, André Freitas, Farid Meziane, and Elisabeth Métais. *Natural Language Processing and Information Systems*. Vol. 9103. Lecture Notes in Computer Science. doi:10.1007/978-3-319-19581-0. New York, USA: Springer Publishing, 2015.

[29] Roi Blanco, Peter Mika, and Sebastiano Vigna. "Effective and Efficient Entity Search in RDF Data." In: *Proceedings of the 10th International Conference on The Semantic Web—Volume Part I* (Bonn, Germany). Ed. by Lora Aroyo, Chris Welty, Harith Alani, Jamie Taylor, and Abraham Bernstein. Vol. 7031. Lecture Notes in Computer Science. doi:10.1007/978-3-642-25073-6_6. Berlin Heidelberg, Germany: Springer, 2011, pp. 83–97.

[30] Antoine Bordes, Jason Weston, and Nicolas Usunier. "Open Question Answering with weakly supervised embedding models." In: *Machine Learning and Knowledge Discovery in Databases*. Vol. 8724. Lecture Notes in Computer Science. doi:10.1007/978-3-662-44848-9_11. Berlin Heidelberg, Germany: Springer, 2014, pp. 165–180.

[31] Christopher Boston, Sandra Carberry, and Hui Fang. "Wikimantic: Disambiguation for Short Queries." In: *17th International Conference on Applications of Natural Language to Information Systems, NLDB 2012*. Ed. by G. Bouma, A. Ittoo, E. Métais, and H. Wortmann. doi:10.1007/978-3-642-31178-9_13. Berlin Heidelberg, Germany: Springer, 2012, pp. 140–151.

[32] Andreas Both, Dennis Diefenbach, Kuldeep Singh, Saedeeh Shekarpour, Didier Cherix, and Christoph Lange. "Qanary–An Extensible Vocabulary for Open Question Answering Systems." In: *The Semantic Web. Latest Advances and New Domains*. Ed. by Harald Sack, Eva Blomqvist, Mathieu d'Aquin, Chiara Ghidini, Simone Paolo Ponzetto, and Christoph Lange. Vol. 9678.

Lecture Notes in Computer Science. Berlin Heidelberg, Germany: Springer, 2016.

[33] Gosse Bouma, Ashwin Ittoo, Elisabeth Métais, and Hans Wortmann, eds. *Natural Language Processing and Information Systems*. Vol. 7337. Lecture Notes in Computer Science. doi:10.1007/978-3-642-31178-9. Berlin Heidelberg, Germany: Springer, 2012.

[34] Paul Buitelaar, Philipp Cimiano, Peter Haase, and Michael Sintek. "Towards Linguistically Grounded Ontologies." In: *The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009*. Ed. by L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvönen, R. Mizoguchi, E. Oren, M. Sabou, and E. Simperl. Vol. 6643. Lecture Notes in Computer Science. doi:10.1007/978-3-642-02121-3_12. Berlin Heidelberg, Germany: Springer, 2009, pp. 111–125.

[35] Elena Cabrio, Alessio Palmero Aprosio, Julien Cojan, Bernardo Magnini, Fabien Gandon, and Alberto Lavelli. "QAKiS @ QALD-2." In: *Interacting with Linked Data (ILD 2012)*. 2012, p. 61.

[36] Elena Cabrio, Philipp Cimiano, Vanessa Lopez, Axel-Cyrille Ngonga Ngomo, Christina Unger, and Sebastian Walter. "QALD-3: Multilingual Question Answering over Linked Data." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[37] Elena Cabrio, Julien Cojan, Fabien Gandon, and Amine Hallili. "Querying multilingual DBpedia with QAKiS." In: *The Semantic Web: ESWC 2013 Satellite Events*. doi:10.1007/978-3-642-41242-4_23. Berlin Heidelberg, Germany: Springer, 2013, pp. 194–198.

[38] Marc Canitrot, Thomas de Filippo, Pierre-Yves Roger, and Patrick Saint-Dizier. "The KOMODO System: Getting Recommendations On How to Realize an Action Via Question-Answering." In: *IJCNLP 2011, Proceedings of the KRAQ11 Workshop: Knowledge and Reasoning for Answering Questions*. 2011, pp. 1–9.

[39] Sarven Capadisli, Sören Auer, and Axel-Cyrille Ngonga Ngomo. "Linked SDMX data." In: *Semantic Web Journal* 6.2 (2015), pp. 105–112.

[40] Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, eds. *CLEF 2014 Working Notes* (Sheffield, UK). Vol. 1180. CEUR Workshop Proceedings. Online Working Notes. 2014. URL: http://ceur-ws.org/Vol-1180.

[41] Claudio Carpineto and Giovanni Romano. "A Survey of Automatic Query Expansion in Information Retrieval." In: *ACM Computing Surveys* 44 (2012). doi:10.1145/2071389.2071390, 1:1–1:50.

[42] Danilo Carvalho, Cagatay Calli, André Freitas, and Edward Curry. "EasyESA: A Low-effort Infrastructure for Explicit Semantic Analysis." In: *Proc. of 13th International Semantic Web Conf*. 2014, pp. 177–180.

[43] Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. "Introduction to Neural Network based Approaches for Question Answering over Knowledge Graphs." In: *CoRR* abs/1907.09361 (2019). arXiv: 1907.09361. URL: http://arxiv.org/abs/1907.09361.

[44] Gong Cheng, Thanh Tran, and Yuzhong Qu. "RELIN: Relatedness and Informativeness-Based Centrality for Entity Summarization." In: *The Semantic Web—ISWC 2011*. Ed. by Lora Aroyo, Chris Welty, Harith Alani, Jamie Taylor, Abraham Bernstein, Lalana Kagal, Natasha Fridman Noy, and Eva Blomqvist. Vol. 7031. Lecture Notes in Computer Science. doi:10.1007/978-3-642-25073-6_8. Berlin Heidelberg, Germany: Springer, 2011, pp. 114–129.

[45] P. Cimiano, O. Corcho, V. Presutti, L. Hollink, and S. Rudolph, eds. *The Semantic Web: Semantics and Big Data*. Vol. 7882. Lecture Notes in Computer Science. doi:10.1007/978-3-642-38288-8. Berlin Heidelberg, Germany: Springer, 2013.

[46] Philipp Cimiano. "Flexible semantic composition with DUDES." In: *Proceedings of the Eighth International Conference on Computational Semantics*. doi:10.3115/1693756.1693786. Stroudsburg, USA: Association for Computational Linguistics, 2009, pp. 272–276.

[47] Philipp Cimiano and Michael Minock. "Natural Language Interfaces: What Is the Problem?—A Data-Driven Quantitative Analysis." In: *Natural Language Processing and Information Systems, 15th International Conference on Applications of Natural Language to Information Systems, NLDB 2010*. Vol. 6177. Lecture Notes in Computer Science. doi:10.1007/978-3-642-12550-8_16. Berlin Heidelberg, Germany: Springer, 2010, pp. 192–206.

[48] Julien Cojan, Elena Cabrio, and Fabien Gandon. "Filling the gaps among DBpedia multilingual chapters for Question Answering." In: *Proceedings of the 5th Annual ACM Web Science Conference*. doi:10.1145/2464464.2464500. New York, USA: Association for Computing Machinery, 2013, pp. 33–42.

[49] Philippe Cudré-Mauroux et al., eds. *The Semantic Web–ISWC 2012*. Berlin Heidelberg, Germany: Springer, 2012. DOI: 10.1007/978-3-642-35176-1.

[50] Richard Cyganiak and Dave Reynolds. *The RDF Data Cube Vocabulary*. Recommendation. World Wide Web Consortium (W3C), 2014. URL: http://www.w3.org/TR/2014/REC-vocab-data-cube-20140116/.

[51] Danica Damljanovic, Milan Agatonovic, and Hamish Cunningham. "FREyA: An Interactive Way of Querying Linked Data Using Natural Language." In: *Proceedings of the 1st Workshop on Question Answering Over Linked Data (QALD-1), Co-located with the 8th Extended Semantic Web Conference*. Ed. by Christina Unger, Philipp Cimiano, Vanessa Lopez, and Enrico Motta. 2011, pp. 10–23.

[52] Mariana Damova, Dana Dannells, Ramona Enache, Maria Mateva, and Aarne Ranta. "Natural language interaction with Semantic Web knowledge bases and LOD." In: *Towards the Multilingual Semantic Web* (2013).

[53] Mariana Damova, Atanas Kiryakov, Kiril Simov, and Svetoslav Petrov. "Mapping the Central LOD Ontologies to PROTON Upper-Level Ontology." In: *Proceedings of the 5th International Workshop on Ontology Matching (OM-2010) collocated with the 9th International Semantic Web Conference (ISWC-2010)*. Ed. by Pavel Shvaiko, Jérôme Euzenat, Fausto Giunchiglia, Heiner Stuckenschmidt, Ming Mao, and Isabel Cruz. Vol. 689. CEUR Workshop Proceedings. 2010. URL: http://ceur-ws.org/Vol-689/.

[54] Hoa Trang Dang, Diane Kelly, and Jimmy J. Lin. "Overview of the TREC 2007 Question Answering Track." In: *Proceedings of The Sixteenth Text REtrieval Conference, TREC 2007, Gaithersburg, Maryland, USA, November 5-9, 2007*. Ed. by Ellen M. Voorhees and Lori P. Buckland. Vol. Special Publication 500-274. National Institute of Standards and Technology (NIST), 2007.

[55] Irina Deines and Dirk Krechel. "A German Natural Language Interface for Semantic Search." In: *Semantic Technology*. Ed. by Hideaki Takeda, Yuzhong Qu, Riichiro Mizoguchi, and Yoshinobu Kitamura. Vol. 7774. Lecture Notes in Computer Science. doi:10.1007/978-3-642-37996-3_19. Berlin Heidelberg, Germany: Springer, 2013, pp. 278–289.

[56] Rodolfo Delmonte. *Computational Linguistic Text Processing–Lexicon, Grammar, Parsing and Anaphora Resolution*. New York, USA: Nova Science Publishers, 2008.

[57] Gianluca Demartini, Beth Trushkowsky, Tim Kraska, and Michael J. Franklin. "CrowdQ: Crowdsourced Query Understanding." In: *CIDR 2013, Sixth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 6-9, 2013, Online Proceedings*. www.cidrdb.org, 2013.

[58] Dennis Diefenbach, Vanessa Lopez, Kamal Singh, and Pierre Maret. "Core techniques of question answering systems over knowledge bases: a survey." In: *Knowledge and Information Systems (KAIS)* 55.3 (201), pp. 529–569.

[59] Corina Dima. "Intui2: A Prototype System for Question Answering Over Linked Data." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[60] Corina Dima. "Answering natural language questions with Intui3." In: *CLEF 2014 Working Notes*. Ed. by Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij. 2014.

[61] Li Ding, Dominic DiFranzo, Alvaro Graves, James R. Michaelis, Xian Li, Deborah L. McGuinness, and James A. Hendler. "TWC data-gov corpus: incrementally generating linked government data from data.gov." In: *WWW '10: Proceedings of the 19th International Conference on World Wide Web* (Raleigh, North Carolina, USA). New York, NY, USA: ACM, 2010. ISBN: 978-1-60558-799-8. DOI: 10.1145/1772690.1772937. URL: http://doi.acm.org/10.1145/1772690.1772937.

[62] Li Ding, Dominic DiFranzo, Alvaro Graves, James Michaelis, Xian Li, Deborah L. McGuinness, and Jim Hendler. "Data-gov Wiki: Towards Linking Government Data." In: *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*. Vol. 10. Atlanta, Georgia: AAAI Press, 2010, p. 1.

[63] Li Ding, Tim Finin, Anupam Joshi, Rong Pan, R. Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi, and Joel Sachs. "Swoogle: A Search and Metadata Engine for the Semantic Web." In: *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management* (Washington, D.C., USA). CIKM '04. New York, NY, USA: ACM, 2004, pp. 652–659. DOI: 10.1145/1031171.1031289.

[64] Bharvi Dixit, Rafal Kuc, Marek Rogozinski, and Saurabh Chhajed. *Elasticsearch: A Complete Guide*. Packt Publishing Ltd, 2017.

[65] Mohnish Dubey, Sourish Dasgupta, Ankit Sharma, Konrad Höffner, and Jens Lehmann. "AskNow: A Framework for Natural Language Query Formalization in SPARQL." In: *The Semantic Web. Latest Advances and New Domains*. 2016, pp. 300–316. URL: http://jens-lehmann.org/files/2016/eswc_asknow.pdf.

[66] Khadija Elbedweihy, Suvodeep Mazumdar, Stuart N Wrigley, and Fabio Ciravegna. "NL-Graphs: A Hybrid Approach toward Interactively Querying Semantic Data." In: *The Semantic Web: Trends and Challenges*. Ed. by Valentina Presutti, Claudia d'Amato, Fabien Gandon, Mathieu d'Aquin, Steffen Staab, and Anna Tordai. Vol. 8465. Lecture Notes in Computer Science. doi:10.1007/978-3-319-07443-6_38. New York, USA: Springer Publishing, 2014, pp. 565–579.

[67] Khadija Elbedweihy, Stuart N Wrigley, and Fabio Ciravegna. "Improving Semantic Search Using Query Log Analysis." In: *Interacting with Linked Data (ILD 2012)*. 2012, p. 61.

[68] Ivan Ermilov, Konrad Höffner, Jens Lehmann, and Dmitry Mouromtsev. "kOre: Using Linked Data for OpenScience Information Integration." In: *SEMANTiCS 2015*. 2015. URL: http://svn.aksw.org/papers/2015/SEMANTICS_ITMOLOD_DEMO/public.pdf.

[69] Ivan Ermilov, Jens Lehmann, Michael Martin, and Sören Auer. "LODStats: The Data Web Census Dataset." In: *Proceedings of 15th International Semantic Web Conference—Resources Track (ISWC'2016)*. 2016. URL: http://svn.aksw.org/papers/2016/ISWC_LODStats_Resource_Description/public.pdf.

[70] Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. "Paraphrase-Driven Learning for Open Question Answering." In: *Proceedings of the 51nd Annual Meeting*. Vol. 1. Stroudsburg, USA: Association For Computational Linguistics, 2013, pp. 1608–1618.

[71] Oscar Ferrandez, Christian Spurk, Milen Kouylekov, Iustin Dornescu, Sergio Ferrandez, Matteo Negri, Ruben Izquierdo, David Tomas, Constantin Orasan, Guenter Neumann, et al. "The QALL-ME Framework: A Specifiable-Domain Multilingual Question Answering Architecture." In: *Journal of Web Semantics* 9 (2011). doi:10.1016/j.websem.2011.01.002, pp. 137–145.

[72] Sébastien Ferré. "SQUALL: A controlled natural language as expressive as SPARQL 1.1." In: *Natural Language Processing and Information Systems*. doi:10.1007/978-3-642-38824-8_10. Berlin Heidelberg, Germany: Springer, 2013, pp. 114–125.

[73] Sébastien Ferré. "SQUALL2SPARQL: A Translator From Controlled English to Full SPARQL 1.1." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[74] Sébastien Ferré. "Sparklis: An expressive query builder for SPARQL endpoints with guidance in natural language." In: *Semantic Web Journal* (2015).

[75] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, John Prager, et al. "Building Watson: An overview of the DeepQA project." In: *AI magazine* 31.3 (2010), pp. 59–79.

[76] David Ferrucci and Adam Lally. "UIMA: An architectural approach to unstructured information processing in the corporate research environment." In: *Natural Language Engineering* 10.3-4 (2004), pp. 327–348.

[77] Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro, eds. *CLEF2013 Working Notes* (Valencia, Spain). Vol. 1179. CEUR Workshop Proceedings. Online Working Notes. 2013. URL: http://ceur-ws.org/Vol-1179.

[78] A. Freitas, E. Curry, J.G. Oliveira, and S. O'Riain. "Querying Heterogeneous Datasets on the Linked Data Web: Challenges, Approaches, and Trends." In: *IEEE Internet Computing* 16 (2012). Ed. by Michael Rabinovich. doi:10.1109/MIC.2011.141, pp. 24–33.

[79] Andre Freitas and Edward Curry. "Natural language queries over heterogeneous Linked Data graphs: A distributional-compositional semantics approach." In: *Proceedings of the 19th international conference on Intelligent User Interfaces*. doi:10.1145/2557500.2557534. New York, USA: Association for Computing Machinery, 2014, pp. 279–288.

[80] André Freitas, Joao Gabriel Oliveira, Edward Curry, Seán O'Riain, and Joao Carlos Pereira da Silva. "Treo: Combining entity-search, spreading activation and semantic relatedness for querying Linked Data." In: *Proceedings of the 1st Workshop on Question Answering Over Linked Data (QALD-1), Co-located with the 8th Extended Semantic Web Conference*. Ed. by Christina Unger, Philipp Cimiano, Vanessa Lopez, and Enrico Motta. 2011, pp. 24–37.

[81] André Freitas, João Gabriel Oliveira, Seán O'Riain, Edward Curry, and João Carlos Pereira Da Silva. "Querying Linked Data Using Semantic Relatedness: A Vocabulary Independent Approach." In: *Natural Language Processing and Information Systems, 16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011*. Ed. by Rafael Munoz, Andres Montoyo, and Elisabeth Metais. Vol. 6716. Lecture Notes in Computer Science. doi:10.1016/j.datak.2013.08.003. Berlin Heidelberg, Germany: Springer, 2011, pp. 40–51.

[82] Tim Furche, Georg Gottlob, Giovanni Grasso, Christian Schallhart, and Andrew Sellers. "OXPath: A language for scalable data extraction, automation, and crawling on the Deep Web." English. In: *The VLDB Journal* 22 (2013). doi:10.1007/s00778-012-0286-6, pp. 47–72.

[83] Evgeniy Gabrilovich and Shaul Markovitch. "Computing Semantic Relatedness Using Wikipedia-Based Explicit Semantic Analysis." In: *IJCAI–07, Proceedings of the 20th International Joint Conference on Artificial Intelligence*. Ed. by Manuela M. Veloso. Vol. 7. Palo Alto, USA: AAAI Press, 2007, pp. 1606–1611.

[84] F. Gandon, M. Sabou, H. Sack, C. d'Amato, P. Cudré-Mauroux, and A. Zimmermann, eds. *The Semantic Web. Latest Advances and New Domains*. Vol. 9088. Lecture Notes in Computer Science. doi:10.1007/978-3-319-18818-8. Cham, Switzerland: Springer Publishing, 2015.

[85] Mingxia Gao, Jiming Liu, Ning Zhong, Furong Chen, and Chunnian Liu. "Semantic mapping from natural language questions to OWL queries." In: *Computational Intelligence* 27 (2011), pp. 280–314.

[86] Daniel Gerber and Axel-Cyrille Ngonga Ngomo. "Extracting Multilingual Natural-Language Patterns for RDF Predicates." In: *Knowledge Engineering and Knowledge Management*. doi:10.1007/978-3-642-33876-2_10. Berlin Heidelberg, Germany: Springer, 2012, pp. 87–96.

[87] Christina Giannone, Valentina Bellomaria, and Roberto Basili. "A HMM-Based Approach to Question Answering Against Linked Data." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[88] Alfio Massimiliano Gliozzo and Aditya Kalyanpur. "Predicting Lexical Answer Types in Open Domain QA." In: *International Journal On Semantic Web and Information Systems* 8.3 (2012), pp. 74–88. DOI: 10.4018/jswis.2012070104.

[89] Bert F Green Jr, Alice K Wolf, Carol Chomsky, and Kenneth Laughery. "Baseball: An automatic question-answerer." In: *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference*. ACM. 1961, pp. 219–224.

[90] Thomas R. Gruber. "A translation approach to portable ontology specifications." In: *Knowledge Acquisition* 5 (1993), pp. 199–220.

[91] Sherzod Hakimov, Hakan Tunc, Marlen Akimaliev, and Erdogan Dogdu. "Semantic Question Answering System Over Linked Data Using Relational Patterns." In: *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. New York, USA: Association for Computing Machinery, 2013, pp. 83–88. DOI: 10.1145/2457317.2457331.

[92] Sherzod Hakimov, Christina Unger, Sebastian Walter, and Philipp Cimiano. "Applying Semantic Parsing to Question Answering Over Linked Data: Addressing the Lexical Gap." In: *Natural Language Processing and Information Systems: 20th International Conference on Applications of Natural Language to Information Systems, NLDB 2015, Passau, Germany, June 17-19, 2015, Proceedings*. Ed. by Chris Biemann, Siegfried Handschuh, André Freitas, Farid Meziane, and Elisabeth Métais. doi:10.1007/978-3-319-19581-0_8. Cham, Switzerland: Springer Publishing, 2015, pp. 103–109.

[93] Thierry Hamon, Natalia Grabar, Fleur Mougin, and Frantz Thiessard. "Description of the POMELO System for the Task 2 of QALD-4." In: *CLEF 2014 Working Notes*. Ed. by Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij. 2014.

[94] Birgit Hamp and Helmut Feldweg. "GermaNet—A lexical-semantic net for German." In: *Proceedings of the ACL/EACL '97 Workshop on Intelligent Scalable Text Summarization*. Ed. by I. Mani and M. Maybury. Stroudsburg, USA: Association for Computational Linguistics, 1997, pp. 9–15.

[95] Zellig S Harris. *Papers in structural and transformational linguistics*. Berlin Heidelberg, Germany: Springer, 2013. DOI: 10.1007/978-94-017-6059-1.

[96] Shizhu He, Shulin Liu, Yubo Chen, Guangyou Zhou, Kang Liu, and Jun Zhao. "CASIA@QALD-3: A Question Answering System Over Linked Data." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[97] Shizhu He, Yuanzhe Zhang, Kang Liu, and Jun Zhao. "CASIA@V2: A MLN-based Question Answering System over Linked Data." In: *CLEF 2014 Working Notes*. Ed. by Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij. 2014.

[98] Daniel M Herzig, Peter Mika, Roi Blanco, and Thanh Tran. "Federated Entity Search Using On-the-Fly Consolidation." In: *The Semantic Web–ISWC 2013*. doi:10.1007/978-3-642-41335-3_11. Berlin Heidelberg, Germany: Springer, 2013, pp. 167–183.

[99] Gerhard Heyer. "Elements of a natural language processing technology." In: *Language Engineering*. Ed. by Gerhard Heyer and Hans Haugeneder. Vieweg, 1995, pp. 15–32.

[100] Lynette Hirschman and Robert Gaizauskas. "Natural Language Question Answering: The View From Here." In: *Natural Language Engineering* 7 (2001). doi:10.1017/S1351324901002807, pp. 275–300.

[101] Pascal Hitzler, Markus Krotzsch, and Sebastian Rudolph. *Foundations of Semantic Web Technologies*. Chapman and Hall/CRC, 2009.

[102] Konrad Höffner, Franziska Jahn, Christian Kücherer, Barbara Paech, Birgit Schneider, Martin Schöbel, Sebastian Stäubert, and Alfred Winter. "Technical Environment for Developing the SNIK Ontology of Information Management in Hospitals." In: *Studies in Health Technology and Informatics*. Vol. 243. 2017, pp. 122–126.

[103] Konrad Höffner and Jens Lehmann. "Towards Question Answering on Statistical Linked Data." In: *Proceedings of the 10th International Conference on Semantic Systems*. doi:10.1145/2660517.2660521. New York, USA: Association for Computing Machinery, 2014, pp. 61–64. DOI: 10.1145/2660517.2660521.

[104] Konrad Höffner, Jens Lehmann, and Ricardo Usbeck. "CubeQA—Question Answering on RDF Data Cubes." In: *The Semantic Web – ISWC 2016*. Vol. 9981. Lecture Notes in Computer Science. Springer International Publishing, 2016, pp. 325–340.

[105] Konrad Höffner, Michael Martin, and Jens Lehmann. "LinkedSpending: OpenSpending becomes Linked Open Data." In: *Semantic Web Journal* 7.1 (2016), pp. 95–104.

[106] Konrad Höffner, Christina Unger, Lorenz Bühmann, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, Daniel Gerber, and Phillip Cimiano. "User Interface for a Template Based Question Answering System." In: *Knowledge Engineering and Semantic Web—4th International Conference*. 2013, pp. 258–264. URL: http://svn.aksw.org/papers/2013/KESW_AutoSparqlTbsl_Demo/public.pdf.

[107] Konrad Höffner, Sebastian Walter, Edgard Marx, Ricardo Usbeck, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. "Survey on Challenges of Question Answering in the Semantic Web." In: *Submitted to the Semantic Web Journal* (2016).

[108] Konrad Höffner, Sebastian Walter, Edgard Marx, Ricardo Usbeck, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. "Survey on Challenges of Question Answering in the Semantic Web." In: *Semantic Web Journal* 8.6 (2017), pp. 895–920.

[109] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean. *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. Tech. rep. W3C, 2004.

[110] Dimitar Hristovski, Dejan Dinevski, Andrej Kastrin, and Thomas C Rindflesch. "Biomedical Question Answering using semantic relations." In: *BMC bioinformatics* 16 (2015). doi:10.1186/s12859-014-0365-3, p. 6.

[111] Ruizhe Huang and Lei Zou. "Natural Language Question Answering Over RDF Data." In: *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*. doi:10.1145/2463676.2463725. New York, USA: Association for Computing Machinery, 2013, pp. 1289–1290.

[112] Konrad Höffner, Franziska Jahn, Anna Lörke, Thomas Pause, Birgit Schneider, Elske Ammenwerth, and Alfred Winter. "Open and Linkable Knowledge About Management of Health Information Systems." In: *MEDINFO 2019: Health and Wellbeing e-Networks for All*. Ed. by Lucila Ohno-Machado and Brigitte Séroussi. Vol. 264. Studies in Health Technology and Informatics. International Medical Informatics Association (IMIA) and IOS Press, 2019, pp. 1678–1679.

[113] Franziska Jahn, Konrad Höffner, Birgit Schneider, Anna Lörke, Thomas Pause, Elske Ammenwerth, and Alfred Winter. "The SNIK Graph: Visualization of a Medical Informatics Ontology." In: *MEDINFO 2019: Health and Wellbeing e-Networks for All*. Ed. by Lucila Ohno-Machado and Brigitte Séroussi. Vol. 264. Studies in Health Technology and Informatics. International Medical Informatics Association (IMIA) and IOS Press, 2019, pp. 1941–1942.

[114] Prateek Jain, Pascal Hitzler, Amit P. Sheth, Kunal Verma, and Peter Z. Yeh. "Ontology Alignment for Linked Open Data." In: *Proceedings of ISWC* (Shanghai, China). doi:10.1007/978-3-642-17746-0_26. Berlin, Heidelberg: Springer, 2010, pp. 402–417.

[115] Amit Krishna Joshi, Prateek Jain, Pascal Hitzler, Peter Z Yeh, Kunal Verma, Amit P Sheth, and Mariana Damova. "Alignment-Based Querying of Linked Open Data." In: *On the Move to Meaningful Internet Systems: OTM 2012*. doi:10.1007/978-3-642-33615-7_25. Berlin Heidelberg, Germany: Springer, 2012, pp. 807–824.

[116] Aditya Kalyanpur, J William Murdock, James Fan, and Christopher Welty. "Leveraging Community-Built Knowledge for Type Coercion in Question Answering." In: *The Semantic Web–ISWC 2011*. doi:10.1007/978-3-642-25093-4_10. Berlin Heidelberg, Germany: Springer, 2011, pp. 144–156.

[117] Grzegorz Kondrak. "String Processing and Information Retrieval: 12th International Conference, SPIRE 2005, Buenos Aires, Argentina, November 2-4, 2005. Proceedings." In: Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. Chap. N-Gram Similarity and Distance, pp. 115–126. ISBN: 978-3-540-32241-2. DOI: 10.1007/11575832_13. URL: http://dx.doi.org/10.1007/11575832_13.

[118] Jung-Tae Lee, Sang-Bum Kim, Young-In Song, and Hae-Chang Rim. "Bridging Lexical Gaps Between Queries and Questions on Large Online Q&A Collections with Compact Translation Models." In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (Honolulu, Hawaii). EMNLP '08. doi:10.3115/1613715.1613768. Stroudsburg, USA: Association for Computational Linguistics, 2008, pp. 410–418.

[119] Jens Lehmann, Chris Bizer, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. "DBpedia - A Crystallization Point for the Web of Data." In: *Journal of Web Semantics* 7.3 (2009), pp. 154–165. DOI: doi:10.1016/j.websem.2009.07.002.

[120] Jens Lehmann and Lorenz Bühmann. "AutoSPARQL: Let Users Query Your Knowledge Base." In: *The Semantic Web: Research and Applications, 8th European Semantic Web Conference, ESWC 2011*. Vol. 6643. Lecture Notes in Computer Science. 2011.

[121] Jens Lehmann, Konrad Höffner, Sandra Prätor, Stephanie Lehmann, Axel-Cyrille Ngonga Ngomo, Alejandra Garcia-Rojas, and Spiros Athanasiou. "GeoKnow: Geo-Anwendungen im DatenWeb." In: *gis.Business* 5 (2013), pp. 48–51.

[122] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, et al. "DBpedia—A large-scale, multilingual knowledge base extracted from Wikipedia." In: *Semantic Web Journal* 5 (2014), pp. 1–29.

[123] Jens Lehmann et al. "DEQA: Deep Web Extraction for Question Answering." In: *The Semantic Web – ISWC 2012*. Springer Berlin Heidelberg, 2012, pp. 131–147. URL: http://jens-lehmann.org/files/2012/iswc_deqa.pdf.

[124] Jens Lehmann et al. "DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia." In: *Semantic Web Journal* 6.2 (2015), pp. 167–195. URL: http://jens-lehmann.org/files/2014/swj_dbpedia.pdf.

[125] Jens Lehmann et al. "Managing Geospatial Linked Data in the GeoKnow Project." In: Studies on the Semantic Web. Amsterdam, Netherlands: IOS Press, 2015. Chap. 4, pp. 51–78. URL: http://jens-lehmann.org/files/2015/ios_geoknow_chapter.pdf.

[126] Jens Lehmann et al. *The GeoKnow Handbook*. Tech. rep. Institute of Applied Informatics, University of Leipzig, 2015. URL: http://jens-lehmann.org/files/2015/geoknow_handbook.pdf.

[127] Vanessa Lopez, Christina Unger, Philipp Cimiano, and Enrico Motta. "Evaluating Question Answering over Linked Data." In: *Journal of Web Semantics* 21 (2013). doi:10.1016/j.websem.2013.05.006, pp. 3–13.

[128] Vanessa Lopez, Victoria Uren, Marta Sabou, and Enrico Motta. "Is Question Answering Fit for the Semantic Web?: A Survey." In: *Semantic Web Journal* 2 (2011). doi:10.3233/SW-2011-0041, pp. 125–155.

[129] Klaus Lyko, Konrad Höffner, René Speck, Axel-Cyrille Ngonga Ngomo, and Jens Lehmann. "SAIM—One Step Closer to Zero-Configuration Link Discovery." In: *The Semantic Web: ESWC 2013 Satellite Events*. Springer Berlin Heidelberg, 2013, pp. 167–172. URL: http://jens-lehmann.org/files/2013/eswc_demo_saim.pdf.

[130] Michael Martin, Bert van Nuffelen, Stefano Abruzzini, and Sören Auer. *The Digital Agenda Scoreboard: A Statistical Anatomy of Europe's way into the Information Age*. Tech. rep. University of Leipzig, 2012. URL: http://svn.aksw.org/papers/2012/SWJ-Scoreboard/public.pdf.

[131] Michael Martin, Claus Stadler, Philipp Frischmuth, and Jens Lehmann. "Increasing the Financial Transparency of European Commission Project Funding." In: *Semantic Web Journal* Special Call for Linked Dataset descriptions.2 (2013), pp. 157–164. URL: http://www.semantic-web-journal.net/system/files/swj435_0.pdf.

[132] Edgard Marx, Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Konrad Höffner, Jens Lehmann, and Sören Auer. "Towards an Open Question Answering Architecture." In: *Proceedings of the 10th International Conference on Semantic Systems*. 2014. URL: http://svn.aksw.org/papers/2014/Semantics_openQA/public.pdf.

[133] Dora Melo, Irene Pimenta Rodrigues, and Vitor Beires Nogueira. "Cooperative Question Answering for the Semantic Web." In: *Actas das Jornadas de Informática da Universidade de Évora 2011*. Ed. by Luís Rato and Teresa Gonçalves. Escola de Ciências e Tecnologia, Universidade de Évora, 2011, pp. 1–6.

[134] Elisabeth Métais, Farid Meziane, Mohamad Sararee, Vijayan Sugumaran, and Sunil Vadera, eds. *Natural Language Processing and Information Systems*. Vol. 7934. Lecture Notes in Computer Science. doi:10.1007/978-3-642-38824-8. Berlin Heidelberg, Germany: Springer, 2013.

[135] Elisabeth Métais, Mathieu Roche, and Maguelonne Teisseire, eds. *Natural Language Processing and Information Systems*. Vol. 8455. Lecture Notes in Computer Science. doi:10.1007/978-3-319-07983-7. New York, USA: Springer Publishing, 2014, pp. 103–109.

[136] Peter Mika, Tania Tudorache, Abraham Bernstein, Chris Welty, Craig Knoblock, Denny Vrandečić, Paul Groth, Natasha Noy, Krzysztof Janowicz, and Carole Goble, eds. *The Semantic Web–ISWC 2014*. Cham, Switzerland: Springer Publishing, 2014. DOI: 10.1007/978-3-319-11915-1.

[137] George A Miller. "WordNet: A Lexical Database for English." In: *Communications of the ACM* 38 (1995). doi:10.1145/219717.219748, pp. 39–41.

[138] George A Miller and Walter G Charles. "Contextual correlates of semantic similarity." In: *Language and cognitive processes* 6 (1991), pp. 1–28. DOI: 10.1080/01690969108406936.

[139] Amit Mishra and Sanjay Kumar Jain. "A survey on Question Answering systems with classification." In: *Journal of King Saud University-Computer and Information Sciences* (2015).

[140] Abdullah M. Moussa and Rehab F. Abdel-Kader. "QASYO: A Question Answering System for YAGO Ontology." In: *International Journal of Database Theory and Application* 4 (2011).

[141] Rafael Muñoz, Andres Montoyo, and Elisabeth Metais, eds. *Natural Language Processing and Information Systems*. Vol. 6716. Lecture Notes in Computer Science. doi:10.1007/978-3-642-22327-3. Berlin Heidelberg, Germany: Springer, 2011.

[142] J William Murdock, Aditya Kalyanpur, Chris Welty, James Fan, David A Ferrucci, DC Gondek, Lei Zhang, and Hiroshi Kanayama. "Typing candidate answers using type coercion." In: *IBM Journal of Research and Development* 56.3.4 (2012), pp. 7–1.

[143] Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. "PATTY: A Taxonomy of Relational Patterns with Semantic Types." In: *EMNLP-CoNLL 2012, 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Proceedings of the Conference*. Stroudsburg, USA: Association for Computational Linguistics, 2012, pp. 1135–1145.

[144] Jun-Ping Ng and Min-Yen Kan. "QANUS: An Open-source Question-Answering Platform." In: *arXiv preprint arXiv:1501.00311* (2015).

[145] Axel-Cyrille Ngonga Ngomo. "A Time-Efficient Hybrid Approach to Link Discovery." In: *Proceedings of OM@ISWC*. 2011.

[146] Axel-Cyrille Ngonga Ngomo. "Link Discovery with Guaranteed Reduction Ratio in Affine Spaces with Minkowski Measures." In: *The Semantic Web–ISWC 2012*. Ed. by Philippe Cudré-Mauroux et al. Berlin Heidelberg, Germany: Springer-Verlag, 2012.

[147] Axel-Cyrille Ngonga Ngomo and Sören Auer. "LIMES—A Time-Efficient Approach for Large-Scale Link Discovery on the Web of Data." In: *IJCAI–11, Proceedings of the 24th International Joint Conference on Artificial Intelligence*. Palo Alto, California, USA: AAAI Press, 2011.

[148] Axel-Cyrille Ngonga Ngomo, Lorenz Bühmann, Christina Unger, Jens Lehmann, and Daniel Gerber. "SPARQL2NL—Verbalizing SPARQL queries." In: *Proceedings of the IW3C2 WWW 2013 Conference*. Ed. by Daniel Schwabe, Virgilio Almeida, Hartmut Glaser, Ricardo Baeza-Yates, and Sue Moon. 2013, pp. 329–332.

[149] Axel-Cyrille Ngonga Ngomo, Jens Lehmann, Sören Auer, and Konrad Höffner. "RAVEN: Active Learning of Link Specifications." In: *Proceedings of the 6th International Conference on Ontology Matching*. Vol. 814. 2011, pp. 25–36. URL: http://jens-lehmann.org/files/2011/raven.pdf.

[150] Axel-Cyrille Ngonga Ngomo, Jens Lehmann, Sören Auer, and Konrad Höffner. *RAVEN—Towards Zero-Configuration Link Discovery*. Tech. rep. University of Leipzig, 2012. URL: http://jens-lehmann.org/files/2012/raven_report.pdf.

[151] Shiyan Ou and Zhenyuan Zhu. "An Entailment-Based Question Answering System Over Semantic Web Data." In: *Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation*. doi:10.1007/978-3-642-24826-9_39. Berlin Heidelberg, Germany: Springer, 2011, pp. 311–320.

[152] Georges Paliouras and Axel-Cyrille Ngonga Ngomo, eds. *BioASQ 2013—Biomedical Semantic Indexing and Question Answering*. Vol. 1094. CEUR Workshop Proceedings. Online Working Notes. 2013, pp. 1–8.

[153] Seonyeong Park, Soonchoul Kwon, Byungsoo Kim, Sangdo Han, Hyosup Shim, and Gary Geunbae Lee. "Question Answering System using Multiple Information Source and Open Type Answer Merge." In: *Proceedings of NAACL-HLT*. doi:10.3115/v1/N15-3023. 2015, pp. 111–115.

[154] Seonyeong Park, Hyosup Shim, and Gary Geunbae Lee. "ISOFT at QALD-4: Semantic similarity-based Question Answering system over Linked Data." In: *CLEF 2014 Working Notes*. Ed. by Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij. 2014.

[155] Peter F. Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Z. Pan, Ian Horrocks, and Birte Glimm, eds. *The Semantic Web–ISWC 2010*. doi:10.1007/978-3-642-17746-0. Berlin Heidelberg, Germany: Springer, 2010.

[156] Suzanne J. Piotrowski and Gregg G. Van Ryzin. "Citizen Attitudes Toward Transparency in Local Government." In: *The American Review of Public Administration* 37.3 (2007), pp. 306–323. ISSN: 1552-3357. DOI: 10.1177/0275074006296777. URL: http://dx.doi.org/10.1177/0275074006296777.

[157] Camille Pradel, Guillaume Peyet, Ollivier Haemmerle, and Nathalie Hernandez. "SWIP at QALD-3: Results, Criticisms and Lesson Learned." In: *CLEF2013 Working Notes*. Ed. by Pamela Forner, Roberto Navigli, Dan Tufis, and Nicola Ferro. 2013.

[158] V. Presutti, C. d'Amato, F. Gandon, M. d'Aquin, S. Staab, and A Tordai, eds. *The Semantic Web: Trends and Challenges*. Vol. 8465. Lecture Notes in Computer Science. doi:10.1007/978-3-319-07443-6. Cham, Switzerland: Springer Publishing, 2014.

[159] Md-Mizanur Rahoman and Ryutaro Ichise. "An automated template selection framework for keyword query over Linked Data." In: *Semantic Technology*. doi:10.1007/978-3-642-37996-3_12. Berlin Heidelberg, Germany: Springer, 2013, pp. 175–190.

[160] Monika Rani, Maybin K Muyeba, and O.P. Vyas. "A hybrid approach using ontology similarity and fuzzy logic for semantic Question Answering." In: *Advanced Computing, Networking and Informatics-Volume 1*. doi:10.1007/978-3-319-07353-8_69. Berlin Heidelberg, Germany: Springer, 2014, pp. 601–609.

[161] Aarne Ranta. "Grammatical framework." In: *Journal of Functional Programming* 14 (2004). doi:10.1017/S0956796803004738, pp. 145–189.

[162] *Regulation (EU, Euratom) No 966/2012*. Article 35: Publication of information on recipients and other information. 2012.

[163] Percy E Salas, Fernando Maia Da Mota, Karin Breitman, Marco A Casanova, Michael Martin, and Sören Auer. "Publishing Statistical Data on the Web." In: *International Journal of Semantic Computing* 06.04 (2012), pp. 373–388. DOI: 10.1142/S1793351X12400119.

[164] Klaus U Schulz and Stoyan Mihov. "Fast string correction with Levenshtein automata." In: *International Journal on Document Analysis and Recognition* 5 (2002). doi:10.1007/s10032-002-0082-8, pp. 67–85.

[165] Saeedeh Shekarpour, Sören Auer, Axel-Cyrille Ngonga Ngomo, Daniel Gerber, Sebastian Hellmann, and Claus Stadler. "Generating SPARQL Queries Using Templates." In: *Web Intelligence and Agent Systems* 11.3 (2013).

[166] Saeedeh Shekarpour, Kemele M. Endris, Ashwini Jaya Kumar, Denis Lukovnikov, Kuldeep Singh, Harsh Thakkar, and Christoph Lange. "Question Answering on Linked Data: Challenges and Future Directions." In: *Proceedings of the 25th International Conference Companion on World Wide Web* (Montréal, Canada). WWW '16 Companion. Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2016, pp. 693–698. DOI: 10.1145/2872518.2890571.

[167] Saeedeh Shekarpour, Konrad Höffner, Jens Lehmann, and Sören Auer. "Keyword Query Expansion On Linked Data Using Linguistic and Semantic Features." In: *ICSC 2013 IEEE Seventh International Conference on Semantic Computing*. doi:10.1109/ICSC.2013.41. Los Alamitos, USA: IEEE Computer Society, 2013.

[168] Saeedeh Shekarpour, Konrad Höffner, Jens Lehmann, and Sören Auer. "Keyword Query Expansion on Linked Data Using Linguistic and Semantic Features." In: *7th IEEE International Conference on Semantic Computing, September 16-18, 2013, Irvine, California, USA*. 2013, pp. 191–197. URL: http://jens-lehmann.org/files/2013/icsc_query_expansion.pdf.

[169] Saeedeh Shekarpour, Edgard Marx, Axel-Cyrille Ngonga Ngomo, and Sören Auer. "SINA: Semantic Interpretation of User Queries for Question Answering On Interlinked Data." In: *Journal of Web Semantics* 30 (2015), pp. 39–51.

[170] Saeedeh Shekarpour, Axel-Cyrille Ngonga Ngomo, and Sören Auer. "Query Segmentation and Resource Disambiguation Leveraging Background Knowledge." In: *Proceedings of the Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference (ISWC 2012)*. Ed. by Giuseppe Rizzo, Pablo N. Mendes, Eric Charton, Sebastian Hellmann, and Aditya Kalyanpur. 2012.

[171] Saeedeh Shekarpour, Axel-Cyrille Ngonga Ngomo, and Sören Auer. "Question Answering on interlinked data." In: *Proceedings of the 22st international conference on World Wide Web*. 2013, pp. 1145–1156.

[172] Yelong Shen, Jun Yan, Shuicheng Yan, Lei Ji, Ning Liu, and Zheng Chen. "Sparse Hidden-dynamics Conditional Random Fields for User Intent Understanding." In: *Proceedings of the 20st international conference on World Wide Web* (Hyderabad, India). Stroudsburg, USA: Association for Computational Linguistics, 2011, pp. 7–16. DOI: 10.1145/1963405.1963411.

[173] Kiyoaki Shirai, Kentaro Inui, Hozumi Tanaka, and Takenobu Tokunaga. "An empirical study on statistical disambiguation of Japanese dependency structures using a lexically sensitive language model." In: *Proceedings of Natural Language Pacific-Rim Symposium*. 1997, pp. 215–220.

[174] E. Simperl, P. Cimiano, A. Polleres, O. Corcho, and V. Presutti, eds. *The Semantic Web: Research and applications*. Vol. 7295. Lecture Notes in Computer Science. doi:10.1007/978-3-642-30284-8. Berlin Heidelberg, Germany: Springer, 2012.

[175] David Smiley and David Eric Pugh. *Apache Solr 3 Enterprise Search Server*. Packt Publishing Ltd, 2011.

[176] Karen Sparck Jones. "A statistical interpretation of term specificity and its application in retrieval." In: *Journal of documentation* 28 (1972), pp. 11–21.

[177] Claus Stadler, Jens Lehmann, Konrad Höffner, and Sören Auer. "LinkedGeoData: A Core for a Web of Spatial Open Data." In: *Semantic Web Journal* 3.4 (2012), pp. 333–354. URL: http://jens-lehmann.org/files/2012/linkedgeodata2.pdf.

[178] Claus Stadler, Michael Martin, and Sören Auer. "Exploring the web of spatial data with Facete." In: *WWW '14: Proceedings of the 23rd International Conference on World Wide Web*. doi:10.1145/2567948.2577022. New York, USA: Association for Computing Machinery, 2014, pp. 175–178.

[179] Claus Stadler, Joerg Unbehauen, Patrick Westphal, Mohamed Ahmed Sherif, and Jens Lehmann. "Simplified RDB2RDF Mapping." In: *Proceedings of the 8th Workshop on Linked Data on the Web (LDOW2015), Florence, Italy*. 2015.

[180] Rob Stewart. "A Demonstration of a Natural Language Query Interface to an Event-Based Semantic Web Triplestore." In: *The Semantic Web: ESWC 2014 Satellite Events: ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers* 8798 (2014), p. 343.

[181] Huan Sun, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. "Open Domain Question Answering via Semantic Enrichment." In: *Proceedings of the 24th International Conference on World Wide Web*. doi:10.1145/2736277.2741651. 2015, pp. 1045–1055.

[182] Huan Sun, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. "Open Domain Question Answering via Semantic Enrichment." In: *Proceedings of the 24th International Conference on World Wide Web*. doi:10.1145/2736277.2741651. New York, USA: Association for Computing Machinery, 2015, pp. 1045–1055.

[183] Marcin Synak, Maciej Dabrowski, and Sebastian Ryszard Kruk. "Semantic Web and Ontologies." In: *Semantic Digital Libraries*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 41–54. ISBN: 978-3-540-85434-0. DOI: 10.1007/978-3-540-85434-0_3.

[184] Cui Tao, Harold R Solbrig, Deepak K Sharma, Wei-Qi Wei, Guergana K Savova, and Christopher G Chute. "Time-Oriented Question Answering From Clinical Narratives Using Semantic-Web Techniques." In: *The Semantic Web–ISWC 2010*. Berlin Heidelberg, Germany: Springer, 2010, pp. 241–256.

[185] George Tsatsaronis, Michael Schroeder, Georgios Paliouras, Yannis Almirantis, Ion Androutsopoulos, Eric Gaussier, Patrick Gallinari, Thierry Artieres, Michael R Alvers, Matthias Zschunke, et al. "BioASQ: A Challenge On Large-Scale Biomedical Semantic Indexing and Question Answering." In: *2012 AAAI Fall Symposium Series*. 2012.

[186] Christina Unger, Lorenz Bühmann, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, Daniel Gerber, and Philipp Cimiano. "Template-based Question Answering over RDF data." In: *Proceedings of the 21st international conference on World Wide Web*. 2012, pp. 639–648.

[187] Christina Unger and Philipp Cimiano. "Pythia: Compositional Meaning Construction for Ontology-Based Question Answering On the Semantic Web." In: *16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011*. Ed. by Rafael Munoz, Andres Montoyo, and Elisabeth Metais. doi:10.1007/978-3-642-22327-3_15. Berlin Heidelberg, Germany: Springer, 2011, pp. 153–160.

[188] Christina Unger and Philipp Cimiano. "Representing and Resolving Ambiguities in Ontology-Based Question Answering." In: *Proceedings of the TextInfer 2011 Workshop on Textual Entailment* (Edinburgh, Scotland). Ed. by Sebastian Padó and Stefan Thater. Stroudsburg, USA: Association for Computational Linguistics, 2011, pp. 40–49.

[189] Christina Unger, Philipp Cimiano, Vanessa Lopez, and Enrico Motta, eds. *Proceedings of the 1st Workshop on Question Answering Over Linked Data (QALD-1), Co-located with the 8th Extended Semantic Web Conference* (Heraklion,Greece). 2011.

[190] Christina Unger, Philipp Cimiano, Vanessa Lopez, Enrico Motta, Paul Buitelaar, and Richard Cyganiak, eds. *Interacting with Linked Data (ILD 2012)* (Heraklion, Greece). Vol. 913. CEUR Workshop Proceedings. 2012. URL: http://ceur-ws.org/Vol-913.

[191] Christina Unger, Corina Forascu, Vanessa Lopez, Axel-Cyrille Ngonga Ngomo, Elena Cabrio, Philipp Cimiano, and Sebastian Walter, eds. *Question Answering over Linked Data (QALD-4)*. 2014.

[192] Christina Unger, Axel-Cyrille Ngonga Ngomo, and Elena Cabrio. "6th Open Challenge on Question Answering over Linked Data (QALD-6)." In: *Semantic Web Evaluation Challenge*. Springer, 2016, pp. 171–177.

[193] Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Lorenz Bühmann, and Christina Unger. "HAWK—Hybrid Question Answering over Linked Data." In: *The Semantic Web. Latest Advances and New Domains*. Ed. by F. Gandon, M. Sabou, H. Sack, C. d'Amato, P. Cudré-Mauroux, and A. Zimmermann. Vol. 9088. Lecture Notes in Computer Science. New York, USA: Springer Publishing, 2015.

[194] Ricardo Usbeck, Michael Röder, Peter Haase, Artem Kozlov, Muhammad Saleem, and Axel-Cyrille Ngonga Ngomo. "Requirements to Modern Semantic Search Engines." In: *Knowledge Engineering and Semantic Web—7th International Conference*. 2016.

[195] Ricardo Usbeck, Michael Röder, Michael Hoffmann, Felix Conrads, Jonathan Huthmann, Axel-Cyrille Ngonga Ngomo, Christian Demmler, and Christina Unger. "Benchmarking Question Answering Systems." In: *Semantic Web Journal* 10.2 (2019), pp. 293–304. DOI: 10.3233/SW-180312. URL: http://www.semantic-web-journal.net/system/files/swj1578.pdf.

[196] Michalis Vafopoulos, Marios Meimaris, Ioannis Anagnostopoulos, Agis Papantoniou, Ioannis Xidias, Giorgos Alexiou, Giorgos Vafeiadis, Michalis Klonaras, and Vasilis Loumos. "Public spending as LOD: the case of Greece." In: *Semantic Web Journal* (2013).

[197] Piek Vossen. *A multilingual database with lexical semantic networks*. Berlin Heidelberg, Germany: Springer, 1998. DOI: 10.1007/978-94-017-1491-4.

[198] *WWW '11: Proceedings of the 20th International Conference on World Wide Web* (Hyderabad, India). doi:10.1145/1963192. New York, USA: Association for Computing Machinery, 2011.

[199] *WWW '12: Proceedings of the 21st International Conference on World Wide Web* (Lyon, France). doi:10.1145/2187836. New York, USA: Association for Computing Machinery, 2012.

[200] *WWW '13 Companion: Proceedings of the 22nd International Conference on World Wide Web Companion* (Rio de Janeiro, Brazil). doi:10.1145/2488388. New York, USA: Association for Computing Machinery, 2013.

[201] *WWW '14: Proceedings of the 23rd International Conference on World Wide Web* (Seoul, Korea). doi:10.1145/2566486. New York, USA: Association for Computing Machinery, 2014.

[202] *WWW '15 Companion: Proceedings of the 24th International Conference on World Wide Web Companion* (Florence, Italy). doi:10.1145/2736277. New York, USA: Association for Computing Machinery, 2015.

[203] Sebastian Walter, Christina Unger, Philipp Cimiano, and Daniel Bär. "Evaluation of a Layered Approach to Question Answering Over Linked Data." In: *The Semantic Web–ISWC 2012*. Ed. by Philippe Cudré-Mauroux et al. doi:10.1007/978-3-642-35173-0_25. Berlin Heidelberg, Germany: Springer, 2012, pp. 362–374.

[204] Chris Welty, J William Murdock, Aditya Kalyanpur, and James Fan. "A comparison of hard filters and soft evidence for answer typing in Watson." In: *The Semantic Web–ISWC 2012*. Ed. by Philippe Cudré-Mauroux et al. doi:10.1007/978-3-642-35173-0_16. Berlin Heidelberg, Germany: Springer, 2012, pp. 243–256.

[205] Stephen Wolfram. *The Mathematica Book*. 5th ed. Champaign, USA: Wolfram Media, 2004.

[206] William A Woods. "Semantics and Quantification in Natural Language Question Answering." In: *Advances in Computers*. Vol. 17. Elsevier, 1978, pp. 1–87.

[207] William A Woods and R Kaplan. "Lunar rocks in natural English: Explorations in natural language question answering." In: *Linguistic structures processing* 5 (1977), pp. 521–569.

[208] Kun Xu, Yansong Feng, and Dongyan Zhao. "Xser@ QALD-4: Answering Natural Language Questions via Phrasal Semantic Parsing." In: *CLEF 2014 Working Notes*. Ed. by Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij. 2014.

[209] Mohamed Yahya, Klaus Berberich, Shady Elbassuoni, Maya Ramanath, Volker Tresp, and Gerhard Weikum. "Deep Answers for Naturally Asked Questions On the Web of Data." In: *Proceedings of the 21st international conference on World Wide Web*. doi:10.1145/2187980.2188070. Stroudsburg, USA: Association For Computational Linguistics, 2012, pp. 445–449.

[210] Mohamed Yahya, Klaus Berberich, Shady Elbassuoni, Maya Ramanath, Volker Tresp, and Gerhard Weikum. "Natural Language Questions for the Web of Data." In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (Jeju Island, Korea). EMNLP-CoNLL '12. Stroudsburg, USA: Association For Computational Linguistics, 2012, pp. 379–390.

[211] Mohamed Yahya, Klaus Berberich, Shady Elbassuoni, and Gerhard Weikum. "Robust Question Answering over the web of Linked Data." In: *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. New York, USA: Association for Computing Machinery, 2013, pp. 1107–1116.

[212] Zi Yang, Elmer Garduno, Yan Fang, Avner Maiberg, Collin McCormack, and Eric Nyberg. "Building optimal information systems automatically: Configuration space exploration for biomedical information systems." In: *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. Ed. by Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi. doi:10.1145/2505515.2505692. New York, USA: Association for Computing Machinery, 2013, pp. 1421–1430.

[213] Eman M. G. Younis, Christopher B. Jones, Vlad Tanasescu, and Alia I. Abdelmoty. "Hybrid Geo-Spatial Query Methods On the Semantic Web with a Spatially-Enhanced Index of DBpedia." In: *Geographic Information Science*. Ed. by N. Xiao, M.-P. Kwan, M.F. Goodchild, and S. Shekhar. doi:10.1007/978-3-642-33024-7_25. Berlin Heidelberg, Germany: Springer, 2012, pp. 340–353.

[214] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. "Quality Assessment for Linked Data: A Survey." In: *Semantic Web Journal* (2015).

[215] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. "Quality Assessment for Linked Data: A Survey." In: *Semantic Web Journal* (2015). URL: http://www.semantic-web-journal.net/content/quality-assessment-linked-data-survey.

[216] Naiyu Zhang, Jean-Charles Creput, WANG Hongjian, Cyril Meurie, and Yassine Ruichek. "Query Answering using User Feedback and Context Gathering for Web of Data." In: *INFOCOMP 2013: The Third International Conference on Advanced Communications and Computation*. Rockport, USA: Curran Press, 2013, pp. 33–38.

[217] Lei Zou, Ruizhe Huang, Haixun Wang, Jeffer Xu Yu, Wenqiang He, and Dongyan Zhao. "Natural language Question Answering over RDF: A graph data driven approach." In: *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. doi:10.1145/2588555.2610525. New York, USA: Association for Computing Machinery, 2014, pp. 313–324.

A

THE CUBEQA QUESTION CORPUS

1. What was the average student grade per semester in year 2010?
2. How many diseases have a rate of >100 deaths per year?
3. What is the number of deaths and the number of clinical trials per disease per year?
4. How many of the current drugs being sold in developing countries are making profits?
5. Which type of products are costing the most to produce in a particular country?
6. What is the central government's debt percentage of GDP of countries in 2010 that have the highest loans due to IBRD?
7. What is % of GDP spent towards healthcare of top 10 countries in 2008?
8. What is average energy use per income level?
9. Which geographical region has the highest rate of population growth?
10. What is the total death rate for all diseases for countries with populations greater than 10,000,000?
11. How much money, does Leipzig and Dresden spend on child care in relation to the birth rate in comparison to the average in Saxony.
12. How much money does Leipzig get from Saxony for education compared to other major cities in Saxony.
13. How much money is spend on education in which German province and how is it comprised of shares from EU, Germany and the province.
14. How is the percentage share of research funding coming from economy compared to the government per faculty and university.
15. How much money does a province spend on education, healthcare and childcare compared to the average earnings of the members of parliament.
16. Relate the earnings of the members of the bundestag to the earnings of the population in the cities Chemnitz, Dresden and Leipzig.
17. How many percent of pupils finish with a high school degree compared to the beginners and graduates of university.
18. How much does the government spend on businesses infrastructure compared to education and healthcare?
19. How much of their income does the population in which age classes spend on transportation compared to food.

20. How much do people in which distance to big cities spend on transportation compared to the local unemployment rate. (Display it on a map.)
21. What is the average monthly income of a German citizen?
22. What was the average inflation in Germany over the last 10 years?
23. How much money did the German government spend for infrastructure projects in 2013?
24. How many kids are born in Berlin on a single day?
25. In Germany, how many hospitals are there?
26. How much money was invested to fight bicycle thefts in Leipzig?
27. Which were the top 10 funded research institutions in Europe in 2013?
28. What's the gross domestic product of the 10 richest and 10 poorest countries?
29. What does a juxtaposition of the top industry subventions and corresponding revenue look like?
30. What is the relation of investments into the health care system and the frequency of visits to the doctors?
31. How many citizens live in a <certain area>?
32. How much money spend <X> on <Y>?
33. How much did building <X> cost?
34. How much money goes into police over time?
35. For what is the money, invested in the police, used for?
36. Where goes my taxes?
37. How much money sends each EU country to the EU?
38. How much money gets each country from the EU?
39. How much earns a politician?
40. Where is the biggest per person income?
41. How much does Germany spend on research a year?
42. How many professor positions per students in a university in Germany?
43. How many spin-off companies were created from Government budget?
44. How many projects which are cooperation between universities/institutes and industrial companies?
45. How many percent of tax money of a person is spend on that person's utilities (including public means)?
46. How much money Germany contribute to European projects?

47. How much European projects German Universities get per year?
48. How many foreign researchers are working in Germany by governmental fund?
49. Top 10 taxpayer companies in Germany?
50. How much money Germany spend to support other countries?

B

THE QALD-6 TASK 3 BENCHMARK QUESTIONS

B.1 TRAINING DATA

1. How much was spent on public works and utilities by the Town of Cary in 2011?
2. Which programs were done under the class of public works and utilities in the expenditure of the Town of Cary?
3. How much did the Town of Cary earn in 2009?
4. Which program earned the most for the Town of Cary?
5. Which department of whiteacre spent the most in 2010?
6. How much did social services in Whiteacre spend for refugee resettlement?
7. Which agencies in the Maldives have proposed expenditure amounts of more than 1 billion Maldivian rufiyaa in 2015?
8. What is the total aid to the Anti Corruption Commission in the Maldives in 2015?
9. Which Ugandan district had the highest budget in 2014?
10. What was the average Uganda health budget over all districts in 2014?
11. What is the highest Newcastle city council payment for supplies and services?
12. What's are the 10 highest payments of the Newcastle city council?
13. How much did the Waltham Forest Council spend on Environment and Regeneration?
14. On the Waltham Forest Council, how much money was given to Synarbor Global Solutions Ltd?
15. How much money give Cheshire West and Chester on Adult Social Care and Health?
16. What are the top 5 narratives in Cheshire West and Chester?
17. How much did the City of Redacre expend from the insurance fund?
18. How much receives each division in the City of Redacre?

    What is the total Wandsworth spending
19. FROM all departments?
20. How much money Wandsworth spends on the criminal records bureau?
21. Top 10 IW Council Spending expense types?

22. Which IW Council service area has the highest spending?
23. Which departments of the city of Springfield had a higher budget in 2005 then in 2006?
24. What is the highest single budget amount in the city of Springfield for public works?
25. Which proportion went to Fullerton of the amount spent on Californian cities in 2010?
26. How many categories are there for californian cities?
27. What was the highest Washington DC employee salary in 2011?
28. What is the average salary of an Engineering Technician in Washington DC?

    What was the amount recieved by King George's Field
29. FROM Big Lottery Fund grants?
30. Over which programmes more than 1000000 pound but less than 10000000 pound in grants were given by the Big Lottery Fund?
31. On which service areas of Gloucestershire was spent more than 1000000 pound in total?
32. How much money did the Gloucestershire Police Authority receive?
33. How much did the department for education pay for extra education services in Scotland?

    What HTM functions are paid
34. FROM the department for education the UK?
35. What are the activity statuses of basic health care in Urozgan?
36. What is the total amount of basic nutrition aid by Cordaid in Afghanistan?
37. What was the total budget on Technical Services in City of Toronto in 2009?
38. Which divisions of the City of Toronto received more than 10000000 canadian dollar in 2010?
39. How much did the Sightsavers charity in Ireland pay in total?
40. How much money does the Special Olympics Ireland charity spend on generating funds?
41. What is the frontex budget for administrative expenditure?
42. What is the average frontex budget chapter budget?
43. What was the largest amount spent on housing and building in the Dublin City Council Expenditure Budget of 2013?
44. How much was the budget amount of the Dublin City Council in 2013?

45. How much financial crisis aid did Austria receive in Guaranetees?
46. How much financial crisis aid did Belgium receive in the year of 2011?
47. How much was charity spending was expended for charitable activities in Bangladesh?
48. What is the total cost of generating funds for public appeals events?
49. What is the amount given by the Department of Health to VWR International LLC?
50. How much was given in total to the Riverside Publishing vendor for educational purposes.
51. What was the highest amount for materials and supplies used in Cameroon in 2008?
52. How much is the total expenditure of the Tigerne Council in Cameroon?
53. When did London get Nominet Trust funding?
54. When did the Web in Society programme get Nominet Trust funding?
55. How much did Armenia spent in 2009 on general public services?
56. What are the budgetary classifications of running expenses in Armenia?
57. In Nigerias proposed budget of 2013, how much is assigned to total overhead costs?

    How much did Nigernian Ministry of Petroleum Resources receive
58. FROM ministries and department agencies?
59. How much did the New York City Council members give to the Gun Hill Basketball Association?
60. In how many years did Dickens give money to American Performing Arts Collaborative, Inc.?
61. How high were the service support costs of the Fingal County Council Expenditure Budget of 2011?
62. What was the lowest amount for the veterinary service in the Fingal County Council expenditure budget?
63. What is the total investment budget for basic education in Mezam, Cameroon?
64. Which areas in Cameroon had an investment budget of more than 5 billion CFA in 2010?
65. How much money was given to works and transport in the Ugandan budget?
66. In which years did the Uganda budget contain money for Education?
67. How many donor entities provided foreign aid for Typhoon Yolanda?

    How much foreign aid went over the Red Cross

68. FROM China?
69. Which type of sector is receiving the most Finland foreign aid in India?
    What types of foreign aid
70. FROM Finland did Belarus receive in 2011?
71. What was the proposed City of Providence budget amount for City Courts by the Muncipal Court?
72. How much was the City of Providence budget for educational materials?
73. When was upgrading and greening in ward 49 paid?
74. What are the funding sources for Cape Town Electricity?
75. Which investor type using CKAN technology received the most funding?
76. When did Pew Charitable Trust invest?
77. How much did Armenia spend in 2006 for buildings and edifices?
78. Which were the admins for personal and catering materials in Armenia in 2006?
79. How much was spent on public utilities in Armenia in 2007?
80. What was the total cost of running expenses for Armenia in 2007?
81. What was the total Scottish Government expenditure of 2013-01-09?
82. What was the smallest amount expended for Environment and Forestry by the Scottish Government in January of 2013?
83. In which year did the City of Oakland have the highest total expenditure budget?
84. What was the total City of Oakland budget for the Administrative Unit in 2012?
85. How high was the recurrent expenditure for the Sierra Leona Government budget in 2013?
86. How much was budgeted for general services for the Office of the President of Sierra Leona in 2013?
87. Which clients received lobbying contributions of more than 50000000 $?
88. Which industry received the most lobbying contributions?
89. How much did the Manchester City Council give for Learning Disabilities to SLC Paragon?
90. Which service areas do the Manchester City Council spendings contain?
91. How much was Albanias 2013 budget for education?

92. What are the identification numbers of housing and community amenities in Albanias 2013 budget?
93. What was the total Kenya County Expenditure in Kiambu by the Thika administration?
94. Which Kenya country administration had the highest expenditure budget?
95. What was the Albania budget for public order and safety in 2007?
    What was the combined Albania budget for health
96. FROM 2007 to 2010?
97. What is the external debt amount of Kwara?
98. How much external debt did rivers have in 2010?
99. How many admins were responsible for mandatory payments in the Armenian approved budget of 2010?
100. How much total running expenses under budgetary classification did the Armenian approved budget of 2010 have for personnel?

B.2 TESTING DATA

1. How much was spent on public safety by the Town of Cary in 2010?
2. How many programs were done under the class of General Government in the expenditure of the Town of Cary?
3. How much did the Town of Cary earn in 2010?
4. Which class achieved the highest revenue for the Town of Cary?
5. For which account type of whiteacre was spent the most?
6. How much interest did the debt service of the city of Whiteacre spend?
7. Which expenses had the highest total amount of proposed expenditures for the Maldives?
8. What was the highest single expenditure amount proposed by the Maldives Broadcasting Corporation?
9. Which Ugandan output had the highest budget in 2014?
10. What was the average Uganda health budget amount in Namutumba District?
11. How many suppliers did the Newcastle city council use for education?
12. How many directorates does the Newcastle city council have?
13. Which suppliers did the Waltham Forest Council utilize for recycling?
14. On the Waltham Forest Council, how much money was given to the Forest Recycling Project?
15. How many narratives are there for Cheshire West and Chester council spending in the category of Marketing?
16. What are the top 3 expenditure categories in Cheshire West and Chester council spending?
17. Which priorities does the insurance fund have for the City of Redacre?
18. How many divisions have safety priority in the City of Redacre spending?
19. What was the total Wandsworth spending in 2013 from the housing department?
20. How much money does Wandsworth spendt on general internal repairs?
21. Top 3 IW Council Spending service areas?
22. Under which directorate does the IW Council service area have the highest revenue?
23. Which departments of the city of Springfield had a higher budget in 2004 then in 2005?

24. What is the highest single budget amount in the city of Springfield for public safety?
25. Which of the Californian cities received the highest amount of money?
26. Under which caption did Livermore receive the highest amount of money in 2011?
27. What is the average Washington DC teacher salary?
28. Which position has the highest average salary in Washington DC?
29. How many big lottery found grants were given in the South West in 2012?
30. Has there been a big lottery fund grant to Stanbury Court Social Club?
31. On which expenses in Gloucestershire was spent more than 10000000 pound in total?
32. How much money did Cheltenham Borough Homes receive?
33. How much did the department for work and pensions pay for Research into Infrastructure?
34. What are the geographic regions in the UK Country Regional Analysis from the scottish executive and its departments for forests?
35. How much cost the implementation of Midwifery Education in Nangarhar?
36. How much was spent on food security by Cordaid in Afghanistan?
37. What was the total expenditure on Materials and Supplies of the City of Toronto in 2010?
38. How much did Ireland charities pay in total governance costs?
39. What was the frontex staff budget in 2005?
40. What was the smallest amount for industrial and commercial facilities in the Dublin City Council Expenditure Budget of 2013?
41. Which country received the highest financial crisis aid?
42. How much was charity spending was expended for charitable activities in Haiti?
43. What is the amount given by the Metropolitan Police Department to Cybernational?
44. What was the highest amount under the sub-account for layout and construction of buildingsin Cameroon in 2009?
45. When did Canada get Nominet Trust funding for the last time?
46. Which admin was reponsible for the most total running expenses in Armenia in 2009?

47. How much is the total amount of statuatory transfers in Nigerias proposed budget of 2013?
48. How much did the New York City Council Members give in 2015 for the Manhattan youth?
49. What was the amount of the smallest community grant in the Fingal County Council expenditure budget?
50. When was the upgrade of the Parks-Baba Park paid?

CURRICULUM VITAE

EDUCATION

2012–2020	PhD student, Research Group Agile Knowledge Engineering and Semantic Web (AKSW)
2003–2012	Diplom, Computer Science, 2.3, Leipzig University
2007–2008	Studying abroad, 2 Semesters, Montpellier 2 University and LIRMM, Montpellier, France
1997–2002	Abitur (A-level), 2.0, Wilhelm-Ostwald-Gymnasium, Leipzig

WORK

2001–2002	Call center agent, FGM Forschungsgruppe Medien GmbH
2002–2003	Civil service, Paul-Gerhardt-Church, Leipzig
2003–2003	Bicycle courier, messenger logistics GmbH
2004–2005	Typesetter in LaTeX, le-tex publishing services
2005–2005	News reporter volunteer, student radio mephisto 97.6
2004–2006	Mascot "Buddel" of the BELANTIS amusement park
2006–2007	Student assistant, Leipzig University
2008–2009	Student assistant, Leipzig University
2010–2011	Student assistant, Research Group Agile Knowledge Engineering and Semantic Web (AKSW)
2011–2011	Software developer, sedruck KG
since 2012	Graduate assistant, AKSW
since 2016	Graduate assistant, Institute for Medical Informatics, Statistics and Epidemiology (IMISE)

TEACHING

1995	Computer Science, school year of 1995/96, 7th grade, comprehensive free school Freie Schule Leipzig e.v.
2013	Semantic Web lecture at KESW School, Saint Petersburg
2013	Semantic Web lecture at Higher School of Economics (HSE), Moscow
2014	Supervisor, software engineering internship "Linked Spending", Leipzig University
2015	Supervisor, software engineering internship "Interactive Financial Calculator for the City of Leipzig", Leipzig University
2016–2020	Seminar on medical coding at Institute for Medical Informatics, Statistics and Epidemiology (IMISE), Medical Faculty, Leipzig University
2019	Supervisor, special interest subject ("Besondere Lernleistung"), Wilhelm-Ostwald-Gymnasium, Leipzig
2020	Supervisor, special interest subject, Wilhelm-Ostwald-Gymnasium, Leipzig

AWARDS

2000	shared 2nd prize and special award at the state level of youth science competition "Jugend Forscht" in Mathematics and Computer Science
2009	shared 3rd prize at LOD Triplification Challenge 2009

SCIENTIFIC REVIEWS AND PROGRAM COMMITTEE

2017	International Conference on Knowledge Engineering and Semantic Web (PC),
	Natural Language Interfaces for Web of Data Workshop (reviewer),
	Computer Science Conference for University of Bonn Students (reviewer)
2016	Semantic Web Journal (reviewer)
2015	International Conference on Knowledge Engineering and Semantic Web (PC),
	Semantic Web Journal (reviewer)
2014	European Semantic Web Conference 2014 (reviewer),
	International Conference on Knowledge Engineering and Semantic Web (PC),
	Journal of Web Semantics (reviewer)
2013	Linked Data on the Web Workshop(reviewer),
	AAAI Conference (reviewer),
	NLP & DBpedia workshop (reviewer)
2012	ACM Hypertext (reviewer)
2011	Studentenkonferenz Informatik Leipzig (PC)

INTERNSHIPS

2008	Software developer, Satin IP, Montpellier, France
2012	Software developer, brox IT-Solutions GmbH, Hannover

SELBSTÄNDIGKEITSERKLÄRUNG

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialen oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, den 23. März 2020

.....
Konrad Höffner