Scalable Quality Assessment of Linked Data

Dissertation
zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

by
Jeremy Debattista
from
Birkirkara, Malta

Bonn, 17.10.2016

Dieser Forschungsbericht wurde als Dissertation von der Mathematisch-Naturwissenschaftlichen Fakultät der Universität Bonn angenommen und ist auf dem Hochschulschriftenserver der ULB Bonn http://hss.ulb.uni-bonn.de/diss_online elektronisch publiziert.
Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

Gutachter:	Prof. Dr. Sören Auer
Gutachter:	Prof. Dr. Michel Dumontier
Tag der Promotion:	28.03.2017
Erscheinungsjahr:	2017

Abstract

In a world where the information economy is booming, poor data quality can lead to adverse consequences, including social and economical problems such as decrease in revenue. Furthermore, data-driven industries are not just relying on their own (proprietary) data silos, but are also continuously aggregating data from different sources. This aggregation could then be re-distributed back to "data lakes". However, this data (including Linked Data) is not necessarily checked for its quality prior to its use. Large volumes of data are being exchanged in a standard and interoperable format between organisations and published as Linked Data to facilitate their re-use. Some organisations, such as government institutions, take a step further and *open* their data. The Linked Open Data Cloud is a witness to this. However, similar to data in data lakes, it is challenging to determine the quality of this heterogeneous data, and subsequently to make this information explicit to data consumers.

Despite the availability of a number of tools and frameworks to assess Linked Data quality, the current solutions do not aggregate a holistic approach that enables both the assessment of datasets and also provides consumers with quality results that can then be used to find, compare and rank datasets' fitness for use. In this thesis we investigate methods to assess the quality of (possibly large) linked datasets with the intent that data consumers can then use the assessment results to find datasets that are fit for use, that is; finding the right dataset for the task at hand. Moreover, the benefits of quality assessment are two-fold: (1) data consumers do not need to blindly rely on subjective measures to choose a dataset, but base their choice on multiple factors such as the intrinsic structure of the dataset, therefore fostering trust and reputation between the publishers and consumers on more objective foundations; and (2) data publishers can be encouraged to improve their datasets so that they can be re-used more. Furthermore, our approach scales for large datasets. In this regard, we also look into improving the efficiency of quality metrics using various approximation techniques. However the trade-off is that consumers will not get the exact quality value, but a very close estimate which anyway provides the required guidance towards fitness for use.

The central point of this thesis is not on data quality improvement, nonetheless, we still need to understand what data quality means to the consumers who are searching for potential datasets. This thesis looks into the challenges faced to detect quality problems in linked datasets presenting quality results in a standardised machine-readable and interoperable format for which agents can make sense out of to help human consumers identifying the *fitness for use* dataset. Our proposed approach is more consumer-centric where it looks into (1) making the assessment of quality as easy as possible, that is, allowing stakeholders, possibly non-experts, to identify and easily define quality metrics and to initiate the assessment; and (2) making results (quality metadata and quality reports) easy for stakeholders to understand, or at least interoperable with other systems to facilitate a possible data quality pipeline. Finally, our framework is used to assess the quality of a number of heterogeneous (large) linked datasets, where each assessment returns a quality metadata graph that can be consumed by agents as Linked Data. In turn, these agents can intelligently interpret a dataset's quality with regard to multiple dimensions and observations, and thus provide further insight to consumers regarding its fitness for use.

Acknowledgements

There are a number of individuals whom I would like to thank for their support over the past three years. First, I would like to thank Prof. Dr. Sören Auer who gave me the opportunity to join the Enterprise Information Systems group at the University of Bonn, and whom with his experience helped me form my own ideas and thoughts for this Ph.D. Furthermore, I would like to express my sincere gratitude to my advisor Dr. Christoph Lange for his continuous support, long discussions, patience, and motivation. I could not have asked for a better mentor for my Ph.D. I would also like to express my gratitude to my friend Dominic, the statistics and mathematical guru who gave his advice on a number of issues in this thesis.

Special thanks goes to Simon, Fabrizio, Steffen and Nicole, with whom we shared moments of fun in Bonn. I would like to thank my siblings, my sister in law, and my little nephew, and especially my parents, who supported me throughout my life. Last but not least, I am very grateful to my better half, Judie, who apart from being supportive and amazing throughout these five years, shared this academic journey with me and was always there when needed.

Thanks and Auf Wiedersehen!

Contents

List of Figures

List of Tables

Part I

Prologue

CHAPTER 1

Introduction

Data, distributed or not, is central to the digital era. Furthermore, data is now being constantly generated, more specifically with advancements in Web applications (e.g. social networks) and hardware compliant with the Internet of Things concept. Data-driven applications are moving towards the *Data Lake* concept. James Dixon compared this theoretical concept to a *"large body of water in a more natural state"* [53], where various publishers contribute towards the "filling" of the lake for data consumers to exploit. With "natural state", Dixon indicates that this data is published without any post-processing and structuring. Due to its correlation to *Big Data*, data in the data lake *"is characterised not only by the enormous volume or the velocity of its generation but also by the heterogeneity, diversity and complexity of the data"* [104]. This data can be picked out according to certain requirements, and exploited by data consumers who can then pre-process and use it in their application.

Large volumes of data are being exchanged in a standard and interoperable format between organisations and published openly as Linked Data to facilitate their re-use. Examples include the Linked Open Data (LOD) Cloud, as well as RDFa[1] and *Microformats*[2] data, which are increasingly being embedded in ordinary Web pages as an effect of initiatives such as *schema.org*[3]. The LOD Cloud accounts for more than 70 billion facts [112]. RDFa or Microformats are, to some extent, embedded in approximately 45% of all Web pages[4]. Although a number of best practices on how to publish Linked Data exist, such as [86, 105], the heterogeneity of this data is reflected in the quality of the data itself.

Linked Data publishers are not always check-boxing best practices during the publishing stage, for a variety of reasons, and this might reduce the *fitness for use* in certain situations. In the LOD Laundromat [18], a number of RDF serialised documents and triples related to the *infrastructure* (such as Serialisation, Content-Type and HTTP Exceptions) were detected as "defective". This shows that data quality is not always the highest priority for data publishers; in this case, such infrastructure problems reduce the data's availability. However, data consumers necessitate high quality data that is suitable for their particular use. Usually, different uses of the same data needs to be viewed from different quality aspects. For example, one would expect that data sources in a question answering system are more or less free of error and trustworthy, whilst correct serialisation is a secondary quality aspect.

Fishing in the data lake for good quality data could be a complex task. Stakeholders would first need to identify the quality criteria required for the use case at hand, and then "filter" this vast data lake based on these criteria. Nevertheless, most of the time, these consumers have to rely on their trust of data

[1] http://www.w3.org/TR/rdfa-primer/. Date Accessed 14th August 2016
[2] http://microformats.org/about. Date Accessed 14th August 2016
[3] http://schema.org. Date Accessed 14th August 2016
[4] http://webdatacommons.org/structureddata/#toc3. Date Accessed 25th July 2016

publishers to satisfy these quality criteria. Therefore, improving and maintaining the data's quality should be one of the main tasks of data publishers.

The definition of quality was attempted in various literature, though all pointed towards the subjectivity of the term. Robert Pirsig defines quality as *the result of care* [132], whilst Juran defines quality as *fitness for use* [90]. Juran's views on data quality were shared by Phillip Crosby, who defined quality as *conformance to* (user) *requirements* [38].

Predominantly, poor data quality have technical and social implications to all stakeholders. From a technical point of view, a poorly selected data source can lead to incorrect results. From a social perspective, this would lead to more skepticism towards the data publisher and possibly also the service provider who is providing some interfaces over the data.

This thesis aims to open up new horizons for Linked Data publishers and consumers by providing a framework to analyse a variety of quality problems in linked datasets and enable consumers to make educated decisions when searching for datasets that are "fit for use".

1.1 Problem Specification and Challenges

Ideally, data consumers should be able to find the right dataset based on a number of quality indicators. Thus, data quality management should be part of the core processes in data publishing. The process of assessing data quality should be time efficient whilst detailed account of the data's quality should be available and accessible to data consumers, in order to facilitate the separation of the wheat from the chaff when it comes to finding the right dataset that is fit for use. However, despite the fact that efforts and initiatives (such as [2, 127, 145]) have been made to improve the quality of Linked Data, consumers are still facing challenges when it comes to the quality appraisal and filtering of these datasets, in order to understand if a particular dataset is appropriate for the task at hand.

Challenge 1: Detecting Quality Problems and Making Information about Quality Accessible.

Current automatic approaches for assessing Linked Data quality are limited to objective quality issues, i.e. deterministic algorithms that given a particular input will always return the same output. Although such algorithms usually have polynomial complexity, there are some that become intractable for large datasets and it is difficult to reach a running time sufficient for practical applications. On the other hand, quality issues that require human's interaction, are subjective to each use case and thus not easy to automate.

The quality of data can (usually) not be described using a single measure, but commonly requires a large variety of quality measures to be computed. Ballou and Pazer describe data quality as a multi-dimensional concept, with indicators divided into category and dimension clusters [15]. Doing this for large datasets poses a substantial data processing challenge. However, for large datasets, meticulously exact quality measures are usually not required. Instead users may want to obtain an approximate indication of the quality they can expect. Linked Data quality can be measured along several dimensions, including accessibility, interlinking, performance, syntactic validity or completeness [160].

Moreover, quality information of linked datasets is rarely available for data consumers, thus making exploitation and re-use more difficult. The challenge here is that publishers might be reluctant to disclose the quality of their published data as they might be afraid that their reputation suffers. From our experience in the DIACHRON - Preserving the Evolving Data Web: Making (Open) Linked Data Diachronic[5] project, data publishers were reluctant to make quality information publicly available as they dwelled on the idea

[5] http://www.diachron-fp7.eu. Date Accessed 14th August 2016

that their customers (i.e. whom they publish data for) might end up using their competitors' services instead. On the other hand, having this quality information privately available to them would be an asset as they would be able to offer better services than that of their rivals. Therefore, the biggest hurdle in publishing quality information of publicly available linked dataset is to re-assure publishers that such information is not there to point fingers at, but to encourage them to improve their linked dataset for further consumption.

Challenge 2: Identifying the Right Quality Indicators for the Task at Hand.

Identifying the right quality indicators for choosing a dataset is a challenge that data consumers are facing. The main problem stems from the fact that different domains require different quality aspects that indicate *fitness for use*. Various authors, such as [25, 57, 80] amongst others, have defined a number of quality factors pertinent to linked datasets. Zaveri et al. [160] provide a systematic review of these and other literature, defining and classifying a broad range of data quality dimensions and categories of such dimensions, as well as concrete metrics for measuring quality in these dimensions. Zaveri et al. define, for example, the category of accessibility dimensions. Within this category, there is the dimension of licensing, which comprises the metrics "[existence of a] machine-readable indication of a license" and "human-readable indication of a license" [160]. Choosing the right quality metrics is a challenge for both quality assessment and quality-based filtering.

Challenge 3: Continuously assessing Data Quality in Dynamic Datasets.

In light of data lakes, the majority of linked datasets can be considered as dynamic, meaning that datasets evolve and change over time, thus quality assessment has to be performed to identify the increase or decrease in the data's quality. Currently, there is no method that can iteratively propagate data quality values from one dataset version to another. This means that using current methods, one requires to restart the quality assessment from the beginning, every time a change is made on a previously assessed dataset.

1.2 Motivation

The Web of Data shares many characteristics with the original Web of Documents. Similar to the Web of Documents, the Web of Data is of varying quality [77]. Quality on the Web of Documents is usually measured indirectly using techniques such as the Page Rank [126] the Hyperlink-Induced Topic Search (HITS) algorithm [95]. The reason for this is that document quality is often only assessable subjectively, and thus an indirect measure such as the number of links created by others to a certain Web page is a good approximation of quality. There is a large variety of measures for Linked Data quality that can be computed automatically, so we do not have to rely on indirect indicators alone.

The Web of Data is continuously changing with large volumes of data from different sources being added. Inevitably, this causes the data to suffer from inconsistency, both at a semantic level (contradictions) and at a pragmatic level (ambiguity, inaccuracies), thus creating a lot of noise around the data. It also raises the question of how *authoritative* and *reputable* the data sources are. Taking *DBpedia*[6] as an example, data is extracted from a semi-structured source created in a crowdsourcing effort (i.e. Wikipedia). This extracted data might have quality problems because it is either mapped incorrectly or the information itself is incorrect. *Data consumers* increasingly rely on the Web of Data to accomplish tasks such as performing analytics or building applications that answer end user questions. Information overload is

[6] http://www.dbpedia.org. Date Accessed 14th August 2016

a consistent problem that these consumers face daily. Ensuring quality of the data on the Web is of paramount importance for data consumers, since it is infeasible to filter this infobesity manually.

Poor quality data may lead to adverse consequences. Moreover, poor information quality can lead to social and economical problems [14, 155] such as decrease in business revenues leading to eventual redundancies. In Linked Data, poor data quality can lead to issues such as incorrect reasoning over inconsistent data and poor interlinks between datasets. Furthermore, the quality of Linked Data mashups depends on the overall quality of the integrated external datasets.

In their editorial [77], Hitzler and Janowicz claim that Linked Data is frequently overlooked (in Big Data scenarios) due to its reputation of having poor quality. Therefore, the rewards of addressing Linked Data quality are two-fold: first inherent quality problems can be detected and subsequently fixed, leading to higher quality over time; and secondly proving high quality within linked datasets ensures better exploitation of the same datasets in potential applications and solutions.

1.3 Research Questions

Following the discussion in the previous sections we defined a core research question:

> *How can we assess (large-scale) linked datasets with regard to the fitness for a use case with the aim of enabling quality-driven selection according to various quality measures?*

The answer to this question creates a leverage to data consumers to search for datasets that are "fit for use" for the task at hand. Therefore, with this research we aim to address **Challenge 1** described in the Section 1.1. Furthermore, the framework that is derived from this research acts as a basis for enabling data consumers to manually choose the right quality indicators to identify or assess datasets (i.e. **Challenge 2**), and as a future starting point for **Challenge 3**. In particular we identify more specific research questions:

RQ1: How can stakeholders be enabled to assess linked datasets' quality based on their choice of quality indicators?

In this question we analyse a methodology for quality assessment and hence implement a framework that enables stakeholders to assess the quality of a dataset and facilitates the *fitness of use* filtering and ranking. Apart from *scalability* (as discussed in RQ3), it is necessary that the framework is:

- *extensible* - in order to allow stakeholders to implement and assess their own quality metrics in light of "fit for use";
- *interoperable* - any output could be re-used in any other semantic frameworks and tools;
- *customisable* - stakeholders can cherry-pick the most important quality indicators for dataset selection.

Furthermore, stakeholders should be able to define their own quality indicators that can be used in the framework, therefore we investigate the possibility of creating a domain specific language for defining such indicators.

RQ2: How can we model and represent information about the quality of heterogeneous linked datasets to enable quality-driven dataset selection?

In this question we investigate how Semantic Web technologies, in particular ontology engineering, can be leveraged to create quality metadata that can be attached linked datasets. It is necessary to capture information that enables data consumers to make informed decisions on what datasets to use based on their quality. Furthermore, we study ways on how quality can be semantically represented, both from a modelling point of view (i.e. representing quality indicators as described in [160]) and metadata point of view.

RQ3: What techniques can be used to scale the assessment of large linked datasets?

Here we study how large linked datasets can be assessed in a time efficient manner. From a metrics perspective, we investigate techniques related to probabilistic approximations and 2D space distance-based outliers, in order to discover whether running time decreases whilst the approximate value remains close to the actual value. Furthermore, we investigate a streaming approach towards the assessment of datasets.

RQ4: What is the quality of existing Data on the Web?

To answer this question, we take a look at the general state of the quality in the datasets that make up the Linked Open Data Cloud[7]. Using the framework, we implement a number of quality metrics relevant for linked datasets described in [160], assess and analyse the quality of a number of linked datasets. Furthermore, based on the quality results obtained during the assessment, we perform a statistical study in order to identify what quality metrics are informative and thus important to describe a dataset's quality.

1.4 Thesis Overview

To prepare the reader for the rest of the document, in this section we present an overview our main contributions and the research areas investigated by this thesis, references to scientific publications covering this work, and an overview of the thesis structure.

1.4.1 Research Map and Contributions

As shown in Figure 1.1, our contributions combine research from the areas of Linked Data and Ontology Engineering, Data Quality and Quality Metrics, and Big Data.

1.4.2 Contributions

Main Contributions

1. *A methodology and framework for assessing the quality of Linked Open Data.*
   We propose a conceptual methodology for assessing Linked Data quality together with a formalisation of this conceptualisation. This framework has been implemented keeping in mind the *scalability*, *extensibility*, *interoperability* and *customisability* principles. Furthermore, the underlying layer of this framework is based on a number of generic and specific ontologies, such as a model to represent quality problems in an assessed dataset. We also define a mechanism that

[7] http://lod-cloud.net. Date Accessed 25th August 2016

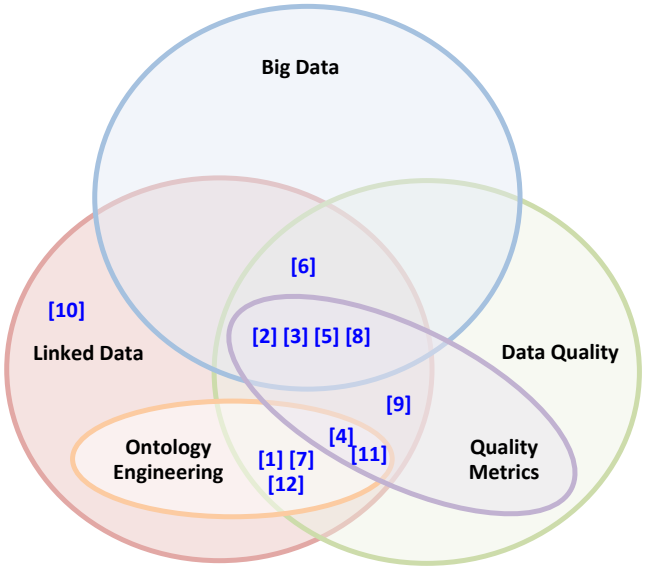

Figure 1.1: Overview of the main research areas and methodologies covered by this thesis. Numbers correspond to the paper contributions listed in Section 1.4.3

enables user-driven quality-based ranking of assess datasets. This framework has a web interface to assist users in the assessment and the subsequent user-ranking and analysis of a dataset's quality metadata in a visual manner. In light of *extensibility*, we analysed the possibility of having a domain specific language developed as part of the framework, in order to allow the definition of quality metrics in the most simplistic way possible. Furthermore, we propose a data quality life cycle, based on previous similar methodologies [16, 138], in terms of co-evolution of linked datasets. More details on this contribution are provided in Chapter 4 and Chapter 5, and our publications [43, 44, 46].

2. *A modelling solution for representing quality indicators and information about a dataset's quality.*
This solution tackles the problem of having quality indicators and information on quality, that is quality metadata, represented in a structured and standardised manner. Having such interoperable metadata, agents (e.g. data consumers) can then query, analyse or visualise a dataset's quality prior to its use in an application. In this regard, we propose an abstract meta-model based on two W3C standard vocabularies; the RDF Data Cube Vocabulary [40] and the Provenance Vocabulary [100], to enable (1) the semantic representation of quality indicators, and (2) the attachment of quality metadata to an assessed dataset. We also developed a proof-of-concept validation tool that validates schemas extending this meta-model that describe quality indicators. More details are provided in Chapter 6, and our publications [45, 47]. Furthermore, the outcome of this work is the basis of a W3C Data Quality Vocabulary recommendation [5].

3. *Approach for improving the time complexity of a number of quality metrics for Linked Data quality assessment.*
Applying techniques frequently used in Big Data scenarios to Linked Data quality assessment, we proposed a number of solutions and developed strategies how such techniques can be applied to boost the running time of quality metric computations. In this regard, we look at *sampling*, *bloom filters*, *clustering coefficient estimations*, and *2D space distance-based* outlier detection, and apply them to a Linked Data quality assessment. In addition, we evaluate a number of metrics that were

implemented using these techniques on a number of real linked datasets. More information on this contribution is provided in Chapter 7 and Chapter 8, and our publication [49].

4. *Quality metadata graphs for a number of LOD Cloud datasets.*
   As part of our large-scale evaluation using the implemented quality framework, we provide the quality metadata for a number of LOD Cloud datasets following the Linked Data publishing principles. To assess the quality of these datasets, we implemented a number of quality metrics for Linked Data for our quality assessment framework, as classified in [160]. More information about this evaluation can be found in Chapter 9.

5. *Identifying the Informative and Non-Informative Linked Data Quality Metrics.*
   Using the results obtained from the quality assessment empirical study, we use the Principal Component Analysis (PCA) test in order to identify the key quality indicators that can give us sufficient information about a dataset's quality. In other words, the PCA will help us to identify the non-informative metrics. More information about this evaluation can be found in Chapter 9, more specifically in Section 9.4.2.

Other Contributions

1. *A system that finds the most "fit for use" data sources in Question Answering.*
   The framework and ontology developed in this thesis were used to assess the quality of the data sources that could be used for answering natural language questions. The results were used in the same system to identify which data source should be used to answer a particular question in a particular domain. In Thakkar et al. [146], a number of quality metrics relevant to Question Answering were identified and two widely used LOD datasets - Wikidata and DBpedia - were assessed over these quality metrics. More specifically, in this evaluation only slices covering the specific domains of restaurants, politicians, films and soccer players in both datasets were assessed. This can be considered as an application of this thesis in the real world.

2. *Understanding literals in data consumed in the LOD Laundromat [8], an access point for crawled Linked Open Data resources on the web.*
   As a result of the collaboration with VU University Amsterdam, we developed two quality metrics related to literals, in order to assess the quality of Linked Data triples in LOD Laundromat. In particular, apart from evaluating the assessment, this collaboration was also an opportunity to verify how the quality assessment methodology and developed framework can be connected in an existing workflow, that of LOD Laundromat. Some of the literal metrics are described in Chapter 9 and in our under-review publication [17].

3. *Big Linked Data - A pipeline to improve Big Data's Value and Veracity.*
   In [48] we discuss how Linked Data can be an enabler to increasing the value and the veracity dimensions in Big Data. Describing various Linked Data tools, including the proposed quality assessment framework, we propose a Semantic Pipeline that enables value creation out of raw data, therefore improving the Big Data value dimension. We also discuss a number of quality indicators in Linked Data to help identify and improve the Big Data veracity dimension.

4. *Activities in the International Scientific Community*
   In addition to the articles published and presented at relevant conferences and workshops, I actively

[8] http://lodlaundromat.org. Date Accessed 20th August 2016

participated in the W3C Data on the Web Best Practices working group. Furthermore, I also co-organised two workshops "Managing the Evolution and Preservation of the Data Web" at the European Semantic Web Conference in 2015 and 2016, and was also active in relevant Programme Committees.

1.4.3 List of Publications

The following publications were used during the compilation of this thesis.

- *Journal Articles*:
  1. **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Methodology and Framework for Linked Data Quality Assessment.* In ACM Journal of Data Information Quality. (To Appear);
  2. **Jeremy Debattista**, Christoph Lange, Sören Auer. *Evaluating the Quality of the LOD Cloud.* To be submitted in a relevant journal;
  3. Wouter Beek, Filip Ilievski, **Jeremy Debattista**, Stefan Schlobach, Jan Wielemaker. *Literally Better: Analyzing and Improving the Quality of Literals.*. In Semantic Web Journal, 2016. Under revision (available online from 05 May 2016).
- *Conference Papers*:
  4. **Jeremy Debattista**, Christoph Lange, Sören Auer. *Representing dataset quality metadata using multi-dimensional views.* In Proceedings of the 10th International Conference on Semantic Systems (SEMANTiCS '14), 92-99, ACM;
  5. **Jeremy Debattista**, Santiago Londoño, Christoph Lange, Sören Auer. *Quality Assessment of Linked Datasets using Probabilistic Approximation.* (One of three nominees for the Best Paper Award) In 12th European Semantic Web Conference Proceedings 2015, 221-236, Springer;
  6. **Jeremy Debattista**, Christoph Lange, Simon Scerri, Sören Auer. *Linked 'Big' Data: Towards a Manifold Increase in Big Data Value and Veracity.* IEEE/ACM 2nd International Symposium on Big Data Computing (BDC) 2015, 92-98, ACM;
  7. **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Framework for Linked Data Quality Assessment.* IEEE Tenth International Conference on Semantic Computing (ICSC) 2016, 124-131, IEEE;
  8. **Jeremy Debattista**, Christoph Lange, Sören Auer. *A Preliminary Investigation Towards Improving Linked Data Quality using Distance-Based Outlier Detection.* Submitted for Review at Joint International Semantic Technology Conference 2016
  9. Harsh Thakkar, Kemele M. Endris, José M. Giménez-García, **Jeremy Debattista**, Christoph Lange, Sören Auer. *Are Linked Datasets fit for Open-domain Question Answering? A Quality Assessment.* In Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics (WIMS) 2016, ACM;
  10. Alan Tygel, Sören Auer, **Jeremy Debattista**, Fabrizio Orlandi, Maria Luiza Machado Campos. *Towards Cleaning-Up Open Data Portals: A Metadata Reconciliation Approach.* IEEE Tenth International Conference on Semantic Computing (ICSC) 2016, 71-78, IEEE;
- *Workshops and Demos*:

11. **Jeremy Debattista**, Christoph Lange, Sören Auer. *daQ, an Ontology for Dataset Quality Information.* In Linked Data on the Web (LDOW) 2014 at WWW'14, CEUR-WS Vol 1184.
12. **Jeremy Debattista**, Christoph Lange, Sören Auer. *Luzzu - A Framework for Linked Data Quality Assessment*. In Posters and Demo Tracks at ISWC15 2015, CEUR-WS Vol 1486.

1.5 Thesis Structure

This thesis is structured into five parts. We introduce this thesis by providing the context, motivation, and research questions in Part I, where we also provide the preliminaries and the related work to give the reader the relevant background before describing our work. In Part II we describe the framework developed for this thesis in order to support the scalable streaming during the assessment of Linked Data Quality. Furthermore, in this part we look into how Domain Specific Languages can be used in order to enable the definition of domain-specific Linked Data quality metrics. In the next part, Part III, we explain how quality metadata can be made available to agents in a machine-interoperable manner. Directing our efforts to the scalability of quality metrics themselves, in Part IV we look at a number of probabilistic approximation techniques with the aim of improving the running time of metrics on large linked datasets whilst ensuring that the quality values are a close accurate. Finally, in Part V, using the output from the previous chapters, we conduct a large scale experiment in order to identify the quality on the Linked Open Data Cloud. We conclude this thesis by revisiting the research questions and provide the future directions of this work. An overview of each relevant chapter is given as a synopsis for each part.

CHAPTER 2

Preliminaries

In this section we will discuss the background related to our research. We start by looking into the main topic of our research, Data Quality and then discuss some background on the Semantic Web and Linked Data. We then discuss techniques that are used in our approach to scale Linked Data quality metrics.

Some parts of this chapter are based on:

- **Jeremy Debattista**, Christoph Lange, Simon Scerri, Sören Auer. *Linked 'Big' Data: Towards a Manifold Increase in Big Data Value and Veracity.* IEEE/ACM 2nd International Symposium on Big Data Computing (BDC) 2015, 92-98, ACM;

- **Jeremy Debattista**, Santiago Londoño, Christoph Lange, Sören Auer. *Quality Assessment of Linked Datasets using Probabilistic Approximation.* (One of three nominees for the Best Paper Award) In 12th European Semantic Web Conference Proceedings 2015, 221-236, Springer;

- **Jeremy Debattista**, Christoph Lange, Sören Auer. *A Preliminary Investigation Towards Improving Linked Data Quality using Distance-Based Outlier Detection.* Submitted for Review at Joint International Semantic Technology Conference 2016

2.1 Data Quality

Various academics, from philosophers to engineers, gave their view on the term *quality*. However, the term *quality* is commonly described as *fitness for use* [90]. More specifically, Juran [90] explains that quality is a measure of how much a product matches the users' requirements. This definition is also reflected in the quality definition described in the ISO standard (9000:2015), Quality management systems – Fundamentals and vocabulary.

In context of *data* quality, fitness for use can be considered to be subjective since different studies show that *data* quality is a multi-dimensional concept [131, 154, 155], that has to be considered according to the context of the data [56]. Therefore, in terms of information systems, it is difficult to have a concise definition for data quality, because one particular dataset may be of good quality for one use case, but not for another. This increases the complexity of data quality. Furthermore, a data publisher has no influence on the quality indicators chosen by a data consumer. However, Wand and Wang [154] suggests that publishers should "provide a design-oriented definition of data quality" in a way that the published data reflects the intended use.

2.1.1 Subjective vs Objective Quality Metrics

Data quality has been the focus of research for a number of years. Literature, such as [15, 131, 155, 160], describe different dimensions and indicators for data quality. Quality metrics can be classified as either *subjective* or *objective* quality indicators [131]. Subjective indicators are those that require the expertise of the stakeholders, that is, the quality assessment is influenced by the stakeholders' decisions and choices. On the other hand, an objective quality indicator will always return the same value, regardless of the stakeholder decision. Pepino et al. [131] classify objective quality indicators as *task independent* metrics or *task dependent* metrics. The former metrics assess datasets irrelevant of the task at hand, for example, whether a dataset has a machine readable license. On the other hand, *task dependent* metrics are those that are specific to the domain of the use case, for example, in a geographic domain one might check for dataset completeness by considering that each place has a geographical latitude, longitude and altitude values.

2.1.2 Quality Indicator Classification Terminology

Various literature uses different levels of classification and terminology for quality indicators. For example in [131], the authors use two-level classification (dimension and indicators), whilst in [155, 160] the authors use a three-level hierarchical classification. In this thesis we stick to the classification provided in the Linked Data Quality survey [160], which we introduce in this section.

- A **Quality Dimension** is a characteristic of a dataset relevant to the consumer (e.g. availability of a dataset).
- A **Quality Metric** is concrete quality measure for a concrete quality *indicator* usually associated with a measuring procedure. This assessment procedure returns a *score*, which we also call the *value* of the metric. There are usually multiple metrics per dimension; e.g., availability can be measured by the accessibility of a SPARQL endpoint, or of an RDF dump. The value of a metric can be numeric (e.g., for the metric "human-readable labelling of classes, properties and entities", the percentage of entities having a human-readable text as label or comment) or boolean (e.g. whether or not a SPARQL endpoint is accessible).
- A **Category** is a group of quality dimensions in which a common type of information is used as quality indicator (e.g. accessibility, which comprises not only availability but also dimensions such as security or performance). Grouping the dimensions into categories helps to organise the space of all quality aspects, given their large number. Zaveri et al. identified 23 quality dimensions grouped into 6 categories [160].
- **Quality Measure** is a term used to refer to the Category-Dimension-Metric combination collectively.

2.1.3 The Importance of Data Quality in Big Data

The term *Big Data* is commonly associated with volume, variety and velocity, better known as the 3V's. The *veracity* dimension, introduced as a fourth V, is frequently overlooked but an equally pressing and challenging dimension. The veracity dimension deals with the uncertainty of data due to various factors such as data inconsistencies, incompleteness, and deliberate deception. To a certain extent, data quality is reliant on veracity.

In [107], Lukoianova and Rubin argue in favour of the importance of the veracity dimension in Big Data due to the lack of attention paid to the dimension so far. The authors state that irrelevant of the processes used to collect data, the input could still suffer from biases, ambiguities and lack of accuracy. Lukoianova and Rubin propose a roadmap towards defining veracity in three dimensions: (1) Objectivity/Subjectivity; (2) Truthfulness/Deception; (3) Credibility/Implausibility. In light of these definitions, the authors also propose (a) a number of quality dimensions, some of which can be mapped to their Linked Data counterparts surveyed in [160], (b) a set of NLP tools that can be used to assess these dimensions, and (c) an index, combining various measures, to assess the Big Data veracity dimension.

The authors of [141] discuss the possibilities and challenges in creating trust in Big Data. This work is similar to [107], though it puts a stronger focus on trust issues. Sanger et al. [141] raise a number of trust-related research questions in four different domains, namely; trust in data quality, measuring trust, trust in nodes (i.e. systems in a network), and trust in providers providing known research efforts. For the first two domains, the authors put forward questions on how input data can be assessed for quality measures, how trust can be measured, and how can trust be increased.

The importance of high quality data was also an issue in the Business & Information Systems Engineering editorial [35]. Buhl et al. state that data availability is one of the most important features for Big Data, together with data consistency regarding time and content, completeness, comprehensiveness, and reliability. The authors also claim that in order to have high quality Big Data, the data should have meaning, so as to enable methods to derive new knowledge. They conclude that the "challenge of managing data quality" should be part of the Big Data nucleus, whilst the creation of data value should be pivotal in business models.

If managing data quality is to be part of this Big Data nucleus, one needs to understand the term *Big Data quality*. Firmani et al. [56] discuss the concept of data quality in Big Data, emphasising on the complexity of defining *Big Data quality* in a generic way, claiming that the term as such is meaningless. Furthermore, the authors show that *Big Data quality* is very source specific, and demonstrated this by describing quality metrics for each of the different Big Data sources (i.e. Social Networks, Traditional Business systems, and Internet of Things) as described by the UNECE Big Data types classification [149].

2.1.4 Data Quality vs Information Quality

So far we have discussed data quality, however, structured data can lead to more context and thus information.

Hu and Feng [84] define data quality as *"the intrinsic quality of data (a type of information bearer) itself"*, whilst information quality as *"the degree to which the information is represented and to which the information can be perceived and accessed"*.

The main difference between the two is that the former deals with the "raw" facts of the data, for example if the date field is 30th February, then that is a data quality problem, thus focusing more on quality characteristics related to the intrinsic aspects of the data. On the other hand, information quality deals with the semantics and accesibility of the data, for example, checking if the geographical longitude and latitude values match with the place that the resource is representing. Nonetheless, information quality builds upon data quality, in a way that if the underlying raw data is incorrect and of poor quality, then this will affect the information quality negatively. However, these two terms are often interchanged in related literature, and as a results there is no general consensus [163]. Similar to [163], in this thesis we make no distinction between terms and stick to *data quality* to refer to the whole spectrum of quality issues.

2.2 The Semantic Web

The idea of *linked information systems*, has been proposed back in 1989 by Berners-Lee [21] and refined a year later in [20]. The breakthrough idea was briefly described as a *decentralised global hypertext space to share information*. In his original proposal, Berners-Lee [21] discusses how nodes describing a particular thing (person or object e.g. document, software, comment, etc . . .) and edges that link nodes together can be used more efficiently than hierarchies. Therefore, this information-sharing space would act as a publishing platform that connects different information sources, possibly from different systems (or as we know them today, servers). However, these links were very limited in scope, describing how two nodes are equivalent (e.g. depends on, uses, etc . . .), giving little consideration to content or meaning of the nodes or links.

The idea of things connected together was further explored by Berners-Lee et al. in [23], where the authors describe the Semantic Web as the "meaningful" web. The authors describe how machine agents can become "intelligent" by understanding the context of the data. Furthermore, they discussed the vision of the Semantic Web, that will give structure to the (then) currently unstructured Web of Documents, moving towards data and information, in order to represent knowledge. This allows agents to search for the right information based on the user's context. Nonetheless, the Semantic Web was not considered to be different from the existing Web but an extension, where things are meaningful to both humans and machines.

In contrast to traditional databases, the Semantic Web, or as it is known, the Web of Data[1], is largely heterogenous as the data is not conforming to one schema, yet represented in a standardised interoperable data model - the Resource Description Framework (RDF). Unlike traditional databases, the Open World Assumption (OWA - as opposed to Closed World Assumption) holds for the Semantic Web. An open world assumption is applied to systems that have incomplete information, where if a fact is not in a system, then the said system cannot say if that axiom is true or not. However, having the right tools (e.g. reasoners), an OWA system can infer axioms based on other known facts. Traditional databases are usually focused on a particular use case, having one schema underlying the data and a single organisation managing and curating the database. On the other hand, anyone can contribute to the Web of Data, which may lead to contradicting, incomplete and incorrect information.

2.2.1 The Resource Description Framework (RDF)

The Resource Description Framework (RDF) [41] is a domain-independent conceptual data model for the Semantic Web that is used to model and represent real-world or abstract concepts as information resources on the web. RDF is a W3C standard data model that is designed to be a machine-readable format whose syntax, grammar and semantics are interoperable across different architectures, described by the RDF Schema (RDFS) vocabulary [33].

RDF is a graph-based data model, in which the basic structure is a *triple* - **subject**$\times$**predicate**$\times$**object**, which can be visualised as two nodes (the subject and the object) connected by an arc (the predicate). A collection of these *triple*s make up an *RDF dataset*, or a *Knowledge Base*. A node, in an RDF Graph can be an *Internationalised Resource Identifier* (IRI)s, a *literal*, or a *blank node*. More specifically, the *subject* is can be an IRI or a blank node; the *predicate* should be an IRI; whilst the *object* can be one of IRI, literal or blank node. An IRI is a more generic URI, conforming to the URI syntax defined in RFC 3987, that allows unicode characters.It usually denotes a thing in the real-world identifying a specific resource or concept and a literal denotes a property of that thing. A literal usually denotes some data value and a datatype. For example, `"1.0324"^^<http://www.w3.org/2001/XMLSchema#double>` is

[1] Defined by W3C Semantic Web Activity WG http://www.w3.org/2001/sw/. . Date Accessed 28th August 2016

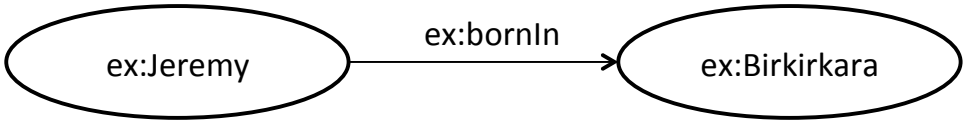

Figure 2.1: An RDF statement representing Jeremy *born in* Birkirkara.

a literal where the first part ("1.0324") is the data value with a double datatype (`http://www.w3.org/2001/XMLSchema#double`). A blank node is a non-persistent local identifier (similar to a local variable in programming) which is neither an IRI nor a literal and that is used in RDF data structures (such as in RDF collections). Blank nodes are referenced within the resource that encloses them, and are not accessible externally. A blank node specifies a (local) relationship between two resources without an explicit name, however it can still represent a thing although without a global IRI reference. For example, if we had to semantify the statement *Jane has one green apple in a bag*, we would have an instance *Jane* that has any *bag* which has an instance *green apple*. In this case, the *bag* represents the blank node that specifies a relationship between *Jane* and the *green apple*.

The RDF data model on its own is a primitive framework to describe the formal meaning of triples (statements), and relies on vocabularies and ontologies (described further in Section 2.2.2) to encapsulate the real-world meaning of the described triples. RDFS provides concepts and properties (terms) to describe the resources in the RDF data model, whilst the Web Ontology Language (OWL) is a more complex schema building upon RDFS.

Let us consider the following sentence as an example: *Jeremy is born in Birkirkara*. This sentence can be represented by an RDF statement with the following structure: *Jeremy* is a subject resource, the predicate (property) *born in*, and an object value of *Birkirkara*, which is another resource. Assuming that all three parts are valid IRIs with the namespace `http://www.example.org/` (for simplicity abbreviated with `ex:`, meaning `http://www.example.org/Jeremy` is the same as `ex:Jeremy`), this sentence can be modelled in RDF and illustrated by a graph as shown in Figure 2.1. RDF/XML[2] is one of the number of serialisation formats used to represent the RDF data model syntax. Listing 2.1 show how the triple in Figure 2.1 can be represented in such a format. Whilst RDF/XML can be easily parsed by any XML tool and library, it is cumbersome to read, especially when the number of triples grow. However, different serialistion, such as Turtle[3], RDFa[4] or JSON-LD[5], have their own pros and cons, depending on the use case.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
    xmlns:example="http://www.example.org/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

  <rdf:Description rdf:about="http://www.example.org/Jeremy">
    <example:bornIn rdf:resource="http://www.example.org/Birkirkara"/>
  </rdf:Description>
</rdf:RDF>
```

Listing 2.1: RDF/XML representation of statement in Figure 2.1.

The resources `ex:Jeremy` and `ex:Birkirkara` can be further defined, for example adding a literal value to each resource and defining the type (`ex:Person` and `ex:Town` respectively). Furthermore we

[2] `http://www.w3.org/TR/rdf-syntax-grammar/`. Date Accessed 2nd September 2016.
[3] `http://www.w3.org/TR/turtle/`. Date Accessed 2nd September 2016.
[4] `http://www.w3.org/TR/rdfa-syntax/`. Date Accessed 2nd September 2016.
[5] `http://www.w3.org/TR/json-ld/`. Date Accessed 2nd September 2016.

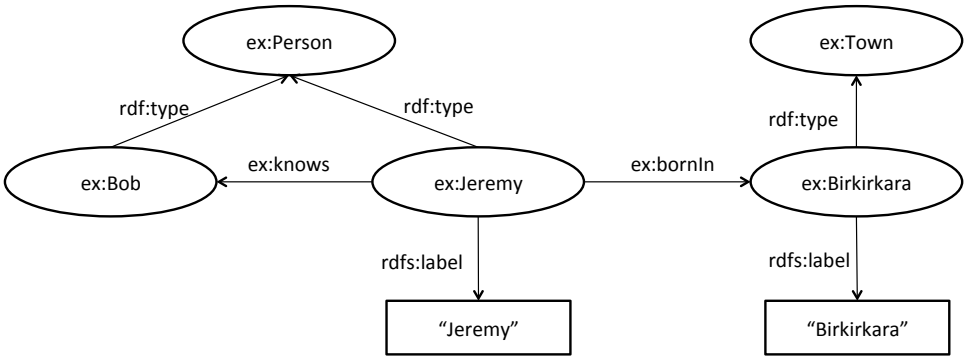

Figure 2.2: An extended RDF graph, showing a small Knowledge Base.

can link these resources with other resources, for example, *Jeremy knows Bob*. These additions to the original statement are shown in Figure 2.2 and serialised in Turtle syntax in Listing 2.2

```
@prefix ex: <http://www.example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

ex:Jeremy rdf:type ex:Person ;
  ex:bornIn ex:Birkirkara ;
  ex:knows ex:Bob ;
  rdfs:label "Jeremy" .

ex:Birkirkara rdf:type ex:City ;
  rdfs:label "Birkirkara" .

ex:Bob rdf:type ex:Person .
```

Listing 2.2: Turtle representation of statement in Figure 2.2.

Furthermore, using the RDF data model, one is not just bounded with explicit descriptions of real-world concepts such as in Listing 2.1 and 2.2. In [72], the editors demonstrate the power of the RDF data model to infer new statements in a knowledge base.

Named Graphs were introduced in [36] by Carroll et al. The idea of these named graphs is to 'group' and name a set of triples by using a URI. This set of triples can then be referred to by using the URI given to the graph. Such named graphs are used, for example, to group provenance metadata triples together. Listing 2.3 shows two graphs serialised in TriG[6]; the default graph which has no URI as a reference, and the named graph `ex:InfoGraph`, which holds a number of triples related to information about the modification of the default graph.

[6] http://www.w3.org/TR/trig/. Date Accessed 1st October 2016.

```
# prefix declaration

{
  ex:Jeremy rdf:type ex:Person ;
    ex:bornIn ex:Birkirkara ;
      ex:knows ex:Bob ;
     rdfs:label "Jeremy" .

  ex:Birkirkara rdf:type ex:City ;
    rdfs:label "Birkirkara" .

  ex:Bob rdf:type ex:Person .
}

ex:InfoGraph {
  ex:Modification ex:dateModified "2016-09-23"^^xsd:date ;
    ex:modifiedBy ex:Jeremy .
}
```

Listing 2.3: TriG representation of two graphs.

2.2.2 Vocabularies and Ontologies - RDFS and OWL

The RDF data model relies on domain dependent and independent schemas - vocabularies and ontologies. These schemas enrich the described resources such that machines can make sense out of these descriptions. The term *ontology* has its origin in philosophy, however, its technical computer science term was re-defined by Gruber in [60]. An ontology can be defined as a domain-specification of related concepts and axioms that captures a view of the real-world that needs to be represented. An ontology tends to be more formal. On the other hand, a *vocabulary* is usually more lightweight and may be a lexicon of terms whose meaning is understood by the agent(s) using it.

The RDF Schema [33] is the schema that encapsulate the RDF data model. It extends the basic RDF vocabulary by adding concepts such as `rdfs:Class`, `rdfs:subClassOf`, `rdfs:subPropertyOf`, `rdfs:domain`, and `rdfs:range` amongst others. The `rdfs:Class` concept enables the definition of classes in new schemas whilst the `rdfs:subClassOf` property allows for a hierarchical organisation. A schema can have a number of properties (`rdf:Property`), each of these can have a domain (identifying the type of the *subject* resource of a triple) and a range (identifying the type of the *object* value of a triple). The domain of a property is usually defined using the `rdfs:domain` property, whilst the range is defined using the `rdfs:range`. Similar to classes, a hierarchical organisation of properties is possible using the `rdfs:subPropertyOf`. RDFS introduced properties targeted for human-readable annotations such as `rdfs:comment` and `rdfs:label`. The RDF Schema is primarily used to build other domain-specific vocabularies that can be used in the Semantic Web to describe different *things*.

The Web Ontology Language (OWL) [144] extends RDFS to handle more complex ontologies. More specifically, OWL introduce new axioms to

1. define complex classes, such as disjointness between classes (`owl:disjointWith`) and enumerated classes (`owl:oneOf`);
2. differentiate between properties that expect a resource-valued type (`owl:ObjectProperty`) or a literal-valued type (`owl:DatatypeProperty`);
3. define complex properties such as `owl:InverseFunctionalProperty`;
4. define cardinality and other constraint restrictions on properties.

2.2.3 The SPARQL Protocol and RDF Query Language (SPARQL)

In order to query RDF knowledge bases, the SPARQL protocol has to be used. The SPARQL query language can be used to query diverse data source using different graph patterns whilst supporting aggregation, subqueries, negation, and constraints [66]. SPARQL has a similar role as of SQL in relational databases.

SPARQL uses a pattern matching approach, matching the triple patterns (basic graph pattern) with variables, against the queried knowledge base. When the basic graph patterns in a query matches an RDF subgraph, the variables are substituted with the RDF values in the resulting graph. Furthermore, SPARQL allows users to add filters, aggregate results, and perform path expressions (similar to XML), perform basic federated queries (i.e. querying multiple data sources), and expression testing amongst other functions that can be found in [66]. The SPARQL query language allows four different query forms: `SELECT`, `ASK`, and `CONSTRUCT`. `SELECT` returns the variables projected and defined in the basic graph patterns and their bindings (i.e. their matched RDF subgraph terms). `ASK` returns a "yes/no" answer, depending on whether or not a query pattern has a solution in the queried knowledge base. `CONSTRUCT` queries create new RDF graphs from the query result using the specified graph template.

In Listing 2.4 we show a simple SPARQL `SELECT` query that returns *towns in Malta with a population of more than 8000, ordered by population in descending numbers.* The variables in the basic graph pattern are indicated with the "?" prefix, whilst bindings for ?town and ?population will be returned since they are projected in the SELECT query. The *FILTER* clause restricts the population to those resources (towns) whose population is larger than 8000, and the final results are ordered in descending order according to the population size. In this example we used the DBpedia SPARQL endpoint [7] and DBpedia knowledge base [8].

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>

SELECT ?town ?population WHERE {
  ?town dbo:country dbr:Malta ;
    dbo:populationTotal ?population .
  FILTER(?population > 8000)
} ORDER BY DESC(?population)
```

Listing 2.4: A SPARQL query example.

2.2.4 Linked Data

Berners-Lee claims that Linked Data is "Semantic Web done right"[9]. In the Semantic Web, resources can be linked together using their unique resource identifier (URI). Thanks to links between resources, one can jump from one source to another in order to retrieve more complete information and answers.

Linked Data refers to the methods used for resource interlinking on the web of data. In line with re-use and interlinking in the Semantic Web, Berners-Lee defined four principles [22]:

1. use URIs as names for things;
2. use HTTP URIs so that people can look up those names;
3. when someone looks up a URI, provide useful information, using the standards (RDF, SPARQL);

[7] http://dbpedia.org/sparql/. Date Accessed 2nd September 2016
[8] http://dbpedia.org/. Date Accessed 2nd September 2016
[9] Taken from the slides Linked Open Data by Sir Tim Berners-Lee http://www.w3.org/2008/Talks/0617-lod-tbl/#(3). Accessed August 2016

4. include links to other URIs so that they can discover more things.

Based on these four principles, the Linked Open Data (LOD) initiative published a number of interconnected datasets in RDF on the web. In order to show the growth of Linked Data, the *LOD Cloud* [112] has been a point of reference to the Linked Data community, comprising a number of linked datasets crawled on the Web of Data or added to the *LODCloud* group in *datahub.io* portal[10]. Datasets in this cloud vary from biomedical to geographical datasets, however they still share the same data model.

Throughout the years, a number of Web publishers adapted their publishing methods to include Linked Data as part of their dissemination. Some of the well known datasets include Wikidata[11], a structured database of the Wikimedia projects, DBpedia, a dataset extracted periodically from Wikipedia sources, and Linked Geo Data[12], a semantic geographic dataset extracted from OpenStreetMap[13]. Furthermore, corporations such as BBC[14], New York Times[15] , Thomson Reuters[16], and governmental entities such as in the United Kingdom[17], Singapore[18], and the United States[19], gave the Linked Data principles a chance.

The success of the Linked Data initiatives is also dependent on the de-facto schemas that are shared and used amongst the datasets. Schemas such as Friend of a Friend (FOAF), Dublin Core (DC), Semantically-Interlinked Online Communities (SOIC) and Simple Knowledge Organisation System (SKOS) are used in datasets in different domains. For example, the FOAF vocabulary is used in around 70% of the datasets in the latest state of the LOD Cloud, whilst DC is used in around 56%[20].

2.3 Approximation Techniques for Improving Quality Metric Scalability

The LOD Cloud comprises datasets having less than 10K triples, and others having more than 1 billion triples. Deterministically computing quality metrics on these datasets might take from some seconds to days. This section introduces three probabilistic techniques commonly used in Big Data applications; they combine with a high probability near-to-accurate results with a low running time. We will refer to these three techniques in Chapter 7 when we discuss how the running time of Linked Data quality metrics can be improved using such techniques.

2.3.1 Sampling

Reservoir sampling is a statistics-based technique that facilitates the sampling of evenly distributed items. The sampling process randomly selects k elements ($\leq n$) from a source list, possibly of an unknown size n, such that each element in the source list has a k/n probability of being chosen [151]. The reservoir

[10] https://datahub.io/group/lodcloud. Date Accessed 12th June 2016
[11] Wikidata is not yet included in the LOD Cloud snapshot https://www.wikidata.org/. Date Accessed 12th June 2016
[12] http://linkedgeodata.org/. Date Accessed 12th June 2016
[13] https://www.openstreetmap.org/. Date Accessed 12th June 2016
[14] http://www.w3.org/2001/sw/sweo/public/UseCases/BBC/. Date Accessed 12th June 2016
[15] http://open.blogs.nytimes.com/2010/03/30/build-your-own-nyt-linked-data-application/. Date Accessed 12th June 2016
[16] http://www.opencalais.com/about-open-calais/. Date Accessed 12th June 2016
[17] http://data.gov.uk. Date Accessed 12th June 2016
[18] http://data.gov.sg. Date Accessed 12th June 2016
[19] http://data.gov. Date Accessed 12th June 2016
[20] These stats are taken from http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state/#toc6. Date Accessed 12th June 2016

sampling technique is part of the *randomised algorithms* family. Randomised algorithms offer simple and fast solutions for time-consuming counterparts by implementing a degree of randomness. Vitter [151] introduces an algorithm for selecting a random sample of k elements from a bigger list of n elements, in one pass. The author discusses that by using a *rejection-acceptance technique* the running time for the sampling algorithm improves. The main parameter that affects the tradeoff between fast computation and an accurate result is the reservoir size (k). The sample should be *large enough* such that the law of large numbers[21] can be applied.

Stratified sampling is another sampling technique that can be used when the data can be partitioned into a number of disjoint subgroups [71]. The idea is that the sample is chosen per-proportion of these subgroups, therefore improving the representative sample. For example, if we need a sample of 60 and have 3 groups of 100, 70, and 30, then a random sample of 30, 21, and 9 is chosen from the 3 groups respectively. Stratification may lead to a lower estimation error when subgroups have homogeneous data than a sampling technique such as the Reservoir Sampling [151], a technique that chooses a sample over the whole data.

2.3.2 Bloom Filters

A Bloom Filter [31] is a fast and space efficient bit vector data structure commonly used to query for elements in a set ("is element A in the set?"). The size of the bit vector plays an important role with regard to the precision of the result. A set of hash functions is used to map each item added to be compared, to a corresponding set of bits in the array filter. The main drawback of a Bloom Filter is that they can produce *false positives*, therefore it is possible that an item is identified as existing in the filter when it is not, but this happens with a very low probability. The trade-off of having a fast computation yet a very close estimate of the result depends on the size of the bit vector. With some modifications, Bloom Filters are useful for detecting duplicates in data streams [19].

2.3.3 Clustering Coefficient Estimation

The clustering coefficient algorithm measures the neighbourhood's density of a node. The clustering coefficient is measured by dividing the number of edges of a node and the number of possible connections the neighbouring nodes can have. The time complexity for this algorithm is $O(n^3)$, where n is the number of nodes in the network. Hardiman and Katzir [62] present an algorithm that estimates the clustering coefficient of a node in a network using *random walks*. A *random walk* is the process where a "walker" jumps from one connected node to another with some probability of ending in a particular node. A random walker stops when the *mixing time* is reached. In a Markov model, mixing time refers to the time until the chain is close to its steady state distribution, i.e. the total number of steps the random walker should take until it retires. Given the right *mixing time*, the value is proved to be a close approximate of the actual value. The authors' suggested measure computes in $O(r) + O(rd_{\max})$ time, where r is the total number of steps in the random walk and $d_{\max}$ is the degree of the node with the highest number of in-links and out-links.

2.4 Identifying Outliers in Knowledge Bases

In this section we introduce the techniques used in our approach to detect potentially incorrect RDF statements in Linked Data. Our preliminary idea that we discuss in Chapter 8 is to make use of distance-based outlier detection techniques, more specifically to apply unsupervised clustering to RDF statements.

[21] http://mathworld.wolfram.com/LawofLargeNumbers.html. Date Accessed 12th June 2016

We apply the technique defined by Knorr et al. in [97], with some modifications to improve computation time (using indexes) and automation (using reservoir sampling to find initial values).

2.4.1 Distance-Based Outlier Detection

Outlier detection is utilised in numerous applications such as fraud detection and text originality detection. In [78], Hodge and Austin describe three different types of approaches for outlier detection. The *Type 1* approach determines outliers "with no prior knowledge of the data" [78]. Assuming a normal distribution, the type 1 approach separates data that looks normal from outliers. This approach is usually suited for statistical approaches with *uni*variate data following a known distribution (e.g. normal or gamma distribution). However, in our case, the data objects (i.e. RDF statements) are *two*-dimensional in relation to the statement's property, as every statement has a subject and an object.

Knorr et al. [97] propose a distance-based technique to overcome this barrier using a k-nearest neighbour (k-NN) style clustering. The authors state that given a maximum number M of required objects (for a cell to be marked as a non-outlier cell) and a distance D, outliers can be detected by searching for objects within the radius D of an object O. The rationale is that the data is partitioned into cells in a two-dimensional space, according to the following axioms:

1. Two data objects are in the same cell **iff** their distance is at most $D/2$;
2. Two data objects are in two different but encircling or buffering cells **iff** their distance is at most D. *Encircling* cells (described as $Layer_1$ in [97]) are those cells surrounding the host cell with coordinates $x \pm 1$ and $y \pm 1$. *Buffer* cells (described as $Layer_2$ in [97]) are those cells surrounding the host cell with coordinates $x \pm 3$ and $y \pm 3$;
3. Two data objects are at least three cells apart from each other **iff** their distance is at least D.

Figure 2.3 depicts a two-dimensional space with various data objects (marked as black dots) in cells. A cell which has $M + 1$ data objects is coloured *red*, whilst its encircling cells are coloured *pink*. This is because the objects in the *red* cell and the objects in the *pink* cells are within distance D of each other. If two or more adjacent cells have more than $M + 1$ data objects together, but less then $M + 1$ individually, are coloured *pink* and thus not marked as outliers.

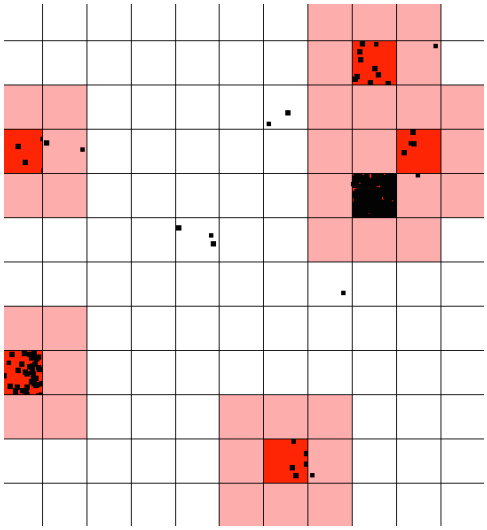

Figure 2.3: A graphical representation of distance-based outlier detection according to Knorr et al. [97] using a fictitious sample set of RDF statements, and cell colouring.

Hodge and Austin [78] state that such techniques suffer from exponential complexity, though in the specific case of 2D spaces, Knorr et al. [97] suggest a linear running time of $O(m + N)$, where m is the number of cells and N is the number of data objects in a dataset. In our approach, we will consider a 2D space and aim to improve the running time by using indexes.

2.4.2 Semantic Similarity Measures

Semantic similarity measures are metrics that are used to determine the semantic relationship between the characteristics of two elements. These measures consider the underlying schema of the instances, more specifically the '*is a*' relations, in order to determine a similarity value between two concepts. More specifically, Harispe et al. [64] defined the notion of semantic measure as a "theoretical tool or function which enables the comparison of elements according to semantic evidences".

In their survey, Harispe et al. [64] describe two different types of semantic evidence: *extensional* and *intentional*. The former is based on the analysis of the topology and usage of classes (i.e. the number of instances associated to that class) and thus the measure is biased towards the usage. This is also known as extrinsic information content. On the other hand, the intentional evidence (or intrinsic information content) considers only the topology of the classes. Although these kind of measures preclude the usage bias, it is assumed that the schema is manifested in depth such that enough knowledge is available to calculate the semantic measure of two concepts. Our approach is not dependent on a particular semantic similarity measure, thus making it adaptable to any use case that requires a particular similarity measure. The semantic similarity measures are out of the scope of this thesis, though apart from the work by [64], readers can refer to `http://www.similarity-blog.de/`, which contains a collection of scientific works related to semantic similarity measures in more depth.

CHAPTER 3

Related Work

In this chapter we discuss previous work related to our research in the context of our main research questions. We first discuss and compare the state-of-the-art Linked Data quality assessment frameworks. Following the framework comparison, we discuss domain specific languages and how these are used in the Semantic Web. We then discuss how previous literature tackled the problem of describe quality metadata. In this chapter we also look at how the probabilistic techniques described in Section 2.3 are used in different domains to achieve different goals in acceptable time, including in quality assessment. This is followed by discussing different outlier techniques used on RDF knowledge bases, mainly statistical distribution, schema enrichment and crowdsourcing. We end this chapter by discussing previous empirical studies that analysed and assessed various aspects of the quality of Data on the Web.

This chapter is based on the related work sections from:

- **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Methodology and Framework for Linked Data Quality Assessment.* In ACM Journal of Data Information Quality. (To Appear);
- **Jeremy Debattista**, Christoph Lange, Sören Auer. *Representing dataset quality metadata using multi-dimensional views.* In Proceedings of the 10th International Conference on Semantic Systems (SEMANTiCS '14), 92-99, ACM;
- **Jeremy Debattista**, Santiago Londoño, Christoph Lange, Sören Auer. *Quality Assessment of Linked Datasets using Probabilistic Approximation.* (One of three nominees for the Best Paper Award) In 12th European Semantic Web Conference Proceedings 2015, 221-236, Springer;
- **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Framework for Linked Data Quality Assessment.* IEEE Tenth International Conference on Semantic Computing (ICSC) 2016, 124-131, IEEE;
- **Jeremy Debattista**, Christoph Lange, Sören Auer. *A Preliminary Investigation Towards Improving Linked Data Quality using Distance-Based Outlier Detection.* Submitted for Review at Joint International Semantic Technology Conference 2016;

3.1 Linked Data Quality Assessment Frameworks

To prepare the reader for the following discussion, we first examine the difference between *data profiling* and *data quality assessment*, two terms that are easily interchangeable. In the Encyclopedia of Database Systems, *data profiling* refers to the systematic way of collecting summaries of the data [103], in

	Flemming	*LinkQA*	*Sieve*	*RDF Unit*	*Triple Check Mate*	*LiQuate*	*TRELLIS*	*tRDF/tSPARQL*	*WIQA*	*Luzzu*
Scalability	✗	✓	✓	✓	Crowdsourcing	N/A	N/A	✓	N/A	✓
Extensibility	✗	Java	XML	SPARQL	✗	Bayesian rules	✗	tSPARQL Rules	WIQA PL	Java, LQML
Quality Metadata	✗	✗	✓ (Optional)	✓	✗	✗	✗	✗	✗	✓
Quality Report	HTML	HTML	✗	HTML, RDF	✗	✗	✗	✗	✗	RDF
Collaboration	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗
Cleaning support	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Last release	2010	2011	2014	2016	2013	2014	2005	2014	2009	2016

Table 3.1: Functional comparison of Linked Data quality tools.

consequence collecting information about the data. On the other hand, *data quality assessment* refers to the process whereby the initiator tests the data against a set of quality indicators, which in turn results into a fixed value that can be used to check whether the data is fit for use. In other words, *data profiling* can be seen as complementary to *data quality assessment*, in a way that profiling can be the first step of the assessment. Therefore, taking into consideration these definitions, we compare Luzzu (introduced in Chapter 4) against 9 other Linked Data quality assessment frameworks which were available to use or view as a demo. Table 3.1 gives an overview of these discussed tools.

Flemming [57] provides a simple web user interface and a walk-through guide that helps a user to assess the data quality of a resource using a set of defined metrics. Metric weights can be customised either by user assignment, though no new metric can be added to the system. Luzzu enables users to create custom metrics, including complex quality metrics that can be customised further by configuring them, such as providing a list of trustworthy data providers. In contrast to Luzzu, Flemming's tool outputs the result and problems as unstructured text.

Similarly to Flemming's tool, the *LiQuate* [137] tool provides a guided walkthrough to view pre-computed datasets. LiQuate is a quality assessment tool based on Bayesian Networks, which analyse the quality of datasets in the LOD Cloud whilst identifying potential quality problems and ambiguities. This probabilistic model is used in LiQuate to explore the assessed datasets for completeness, redundancies and inconsistencies. Data experts are required to identify rules for the Bayesian Network.

Trellis [59] provides a collaborative environment that enables users to annotate facts, based on a trust vocabulary, according to their possible subjective observations, opinions and conclusions. The annotations in turn helps other users in decision making regarding the degree of trust of a dataset. Similar to Trellis, *Triple check mate* [161] is mainly a crowdsourcing tool for quality assessment, supported with a semi-automatic verification of quality metrics. With the crowdsourcing approach, certain quality problems (such as semantic errors) might be more easily detected by human evaluators rather than by computational algorithms. On the other hand, the proposed semi-automated approach makes use of reasoners and machine learning to learn characteristics of a knowledge base.

WIQA [26], is another quality assessment framework that enables users to create and apply a number of policies based on indicators such as provenance and background context related to the data providers. This framework bases its policy filtering on named graphs, which is assumed to have quality related and other meta information. Therefore, unlike in Luzzu, the filtering assessment is not done on the actual data. On the other hand, we can draw parallels between the ranking feature described in Section 4.3.8 and WIQA, where in the former quality metadata is also used to filter datasets based on what the user deems fit for purpose. Whilst the filtering in Luzzu is limited to assigning weights at different granularity levels of quality assessment (i.e. category, dimension, metric), the WIQA framework provides a SPARQL like language (WIQA-PL) that enables a user to apply any assessment metric over the defined quality metric indicator in the named graph. WIQA does not provide any quality metadata or quality problem reports, but provides an assessment result that includes the set of matching triples (against the chosen policies) with an explanation why each triple fulfils the policy(ies).

The *tRDF* framework [69] together with the *tSPARQL* specification [68] provides a number of trust

assessment metrics in order to determine the trustworthiness of RDF statements. The tRDF framework has an underlying semantic model that enables the introduction of a quantifiable trust value in RDF statements, which in turn are used in three different assessment strategies. On the other hand, tSPARQL is an extension to the mentioned framework that enables users to describe trust requirements in SPARQL queries. Similar to Luzzu and other related frameworks, the tRDF framework claims to be scalable, using a number of caching strategies to process results.

LinkQA [61] is an assessment tool to measure the quality (and changes in quality) of a dataset using network analysis measures. The authors provide five network measures, namely degree, clustering coefficient, centrality, OWL sameAs chains, and descriptive richness through OWL sameAs. Similar to Luzzu, new metrics can be integrated into LinkQA, but these metrics are related to topological measures. LinkQA reports the assessment findings in HTML. Although structured, such reports cannot be attached to linked datasets for further machine re-use, such as for running queries over the results of an earlier assessment.

Sieve [114] uses metadata about named graphs to assess data quality, where assessment metrics are declaratively defined by users through an XML configuration. In such configurations, scoring functions (that can also be extended or customised) can be applied on either one metric or an aggregate of metrics. Whereas Luzzu provides a quality problem report that can potentially be imported in third party cleaning tool, Sieve provides its own data cleaning process, where data is cleaned based on a user configuration. Data cleaning is the process of fixing (correcting or removing) incorrect data from a dataset. The quality assessment tool is part of the Linked Data Integration Framework (LDIF), which supports Hadoop[1], a framework for scalable distributed processing.

RDFUnit [99] provides test-driven quality assessment for Linked Data. In RDFUnit, users define quality test patterns based on a SPARQL query template. Similar to Luzzu and Sieve, this gives the user the opportunity to adapt the quality framework to their needs. The focus of RDFUnit is to check for integrity constraints expressed as SPARQL patterns. Thus, users have to understand different SPARQL patterns that represent these constraints. Quality is assessed by executing the custom SPARQL queries against dataset endpoints. RDFUnit is favourable for smaller datasets, where streaming is not required, as SPARQL queries executed directly on an endpoint are transaction-safe. In contrast, Luzzu does not rely on SPARQL querying to assess a dataset, and therefore can compute more complex processes (such as checking for dereferenceability of resources) on dataset triples themselves. Test case results, both quality values and quality problems, from an RDFUnit execution, are stored and represented as Linked Data and visualised as HTML. However, Luzzu uses the Dataset Quality Vocabulary (daQ, introduced in Chapter 6) that enables a more fine-grained and detailed quality metric representation. For example, representing quality metadata with daQ enables the representation of a metric's value change over time.

3.2 Domain Specific Languages

A Domain Specific Language (DSL) is a small declarative programming language focusing on a particular domain, offering appropriate notations and abstractions, in a way that is easy to use for non-programmers [52, 85, 115]. The authors of [52, 85] describe a number of benefits of DSLs, including:

- the enhancement of productivity;
- the incorporation of domain knowledge;
- the possibility of portability;
- the understandability of declarative programs by domain experts themselves;

[1] https://hadoop.apache.org. Date Accessed 23rd May 2016

- easily maintainable code.

A DSL development methodology starts with the *decision* stage, where stakeholders decide if the effort in investing in a new DSL pays off in the future. If the stakeholders decide to go ahead, then they proceed to the *analysis* stage, where the problem domain is identified and knowledge about that domain is gathered. Following that, the DSL is then *design*ed where the knowledge is concisely described as semantic notations and graphically by using tools such as a feature model. Finally, the DSL is *implement*ed. In the article "When and How to Develop Domain-Specific Languages", Mernik et al. [115] provide the reader with a comprehensive insight on DSL development methodologies, by identifying patterns for the four stages of the development methodology. In this section we highlight a few examples from a growing list of domain specific languages used within different areas in computer science and information technology. Each DSL builds on a data model to encapsulate domain knowledge into an abstract notation.

Domain specific languages are popular within various applications. LaTeX is a document preparation typesetting system usually used for technical and scientific publications. HyperText Markup Language (HTML) is a markup language used to generate websites, usually using Cascading Style Sheets (CSS), a DSL used to specify styles to an HTML page. XPath[2] is an expression language enabling the processing of values in an XML data model. XPath uses path expressions to navigate through XML. RuleML[3] is an XML markup language that allows rules to be defined using a formal notation. The Structured Query Language (SQL) is used in relational database management systems in order to allow user to query, navigate and manage data. Yet Another Compiler Compiler (YACC), is a DSL that allows users to create a parser generator using a BNF-like (Backus-Naur Form) notation. YACC was used in order to implement the Luzzu Quality Metric Language described in Chapter 5.

In the Semantic Web technology stack we have a number of DSLs for different purposes. SPARQL[4] is a domain specific query language for querying the RDF data model. SPARQL is described further in Section 2.2.3. The Shapes Constraint Language (SHACL) [133], also known as RDF data shapes, are used to describe and constrain the contents of a given RDF graph.

Going further away from the data model, the Rule Interchange Format (RIF) [94] is a web standard defining an interchange language for rules within different systems to achieve interoperability. It focuses on the definition of various dialects, which enables the exchange of rules within different systems. Similar to RIF, The Semantic Web Rule Language (SWRL) [83] combines the OWL DL and OWL Lite sublanguages (cf. Section 2.2.2) together with Rule Markup Language[5] (RuleML) in order to extend the OWL axioms with Horn-like rules. A *Horn clause* is a formula expressed in first-order logic (FOL), that is used in programming to define clauses. For example, the following FOL:

$$child(?x, ?y) \wedge sister(?y, ?z) \implies relative(?x, ?z)$$

can be phrased as if $?x$ is the child of $?y$, and the sister of $?z$ is $?y$, then we can imply that $?z$ is a relative (in this case uncle or aunt) of $?x$. The purpose of SWRL to to combine such FOL rules with OWL knowledge bases.

The above mentioned DSLs are just a few examples tackling different aspects of the RDF data model. LODStats [55] is a stream-based framework that profiles linked datasets. Similar to Luzzu, the LODStats framework can be extended with new statistical rules, for profiling, defined in a declarative manner. Similar to these, the proposed Luzzu Quality Metric Language is also based on this semantic data model.

[2] http://www.w3.org/TR/xpath-30/. Date Accessed 23rd May 2016
[3] http://ruleml.org. Date Accessed 23rd May 2016
[4] http://www.w3.org/TR/rdf-sparql-query/. Date Accessed 12th September 2016
[5] http://wiki.ruleml.org. Date Accessed 12th September 2016

In Section 3.1 we described a number of Linked Data quality assessment frameworks also mentioning how such frameworks can be be extended or customised.

3.3 Describing Quality Metadata

Fürber et al. propose an OWL ontology that primarily represents data requirements. Such rules can be defined by the users themselves, and the authors present SPARQL queries that "execute" the definitions of the requirements to compute metric values. Unlike our Dataset Quality Vocabulary (daQ) described in Chapter 6, the Data Quality Management vocabulary (DQM) defines a number of classes that can be used to represent a data quality rule. Similarly, properties for defining rules and other generic properties such as the rule creator are specified. The daQ model allows for integrating such DQM rule definitions using the *daq:requires* abstract property, but we consider the definition of rules out of daQ's own scope.

Missier et al. [118] proposed a framework, called quality views, whereby users can identify their quality requirements and eventually attach them to the data processing environment. Keeping in mind re-usability of quality components, the authors propose an OWL model, the IQ model, that captures the quality concepts and the "in-between" relationships, including metrics and dimensions, represented as quality assertions and quality properties respectively. We can draw parallels with daQ, since this model captures the semantics of user-defined metrics and dimensions. On the other hand, the daQ model enables the description of quality observations which could then be used in a number of scenarios.

The SemQuaRE ontology [134] is based on ISO standards and terminology to describe quality measures and units of measurements for semantic-based tools. Similar to the IQ model, SemQuaRE is aimed at describing quality measures rather than the assessed data/tool quality. In order to represent assessment results, the authors from SemQuaRE make use of an external ontology, Evaluation Result Ontology[6]. The Evaluation Result Ontology is used to describe knowledge related to the results obtained during some evaluation process, such as a quality evaluation process. Whilst both the described ontology and daQ represent evaluation values in a meaningful manner, the daQ vocabulary represents quality values as observations, thus recording the improvement or the deterioration of quality value over time. Furthermore, the daQ ontology expects that each quality value result has provenance information attached to it.

Ermilov et al. [55] present a framework calculating comprehensive statistics on linked datasets. This statistical metadata has an underlying ontology based on VoID and Data Cube. The authors argue that since the VoID ontology was not sufficient to cover the required statistical concepts, the Data Cube extension was required. This extension also presented the authors with an opportunity to represent such statistical data using arbitrary attribute dimensions. Motivated by this work, we model resources and their calculated metrics as Data Cube observations, allowing us to represent quality metadata in a multi-dimensional manner. Similar to [55], Mäkelä presents Aether [110], an extension to VoID to enable more fine grained statistical descriptions of datasets.

The W3C recommends VoID and the Data Catalog Vocabulary (DCAT [108]) for metadata describing datasets. The Vocabulary of Interlinked Datasets (VoID) ontology allows the high-level description of a dataset and its links [6, 7]. On the other hand, DCAT describes datasets in data catalogues, which aid discovery, allow easy interoperability between data catalogues and enable digital preservation. With the daQ vocabulary, we aim to add on what these two ontologies have managed to achieve for datasets in general to the specific aspect of data *quality*: enabling the discovery of a good quality (fit for use) datasets by providing the facility to *stamp* a dataset with quality metadata.

[6] http://purl.org/net/EvaluationResult. Date Accessed 15th September 2016

3.4 Probabilistic Techniques in Action

This section overviews the state-of-the-art in relation to the probabilistic approximation techniques that can be applied to assess data quality in Linked Open Datasets. To our knowledge, there is currently no concrete use of such techniques to assess linked dataset quality.

Since their inception, *Bloom Filters* have been used in different scenarios, including dictionaries and spell-checkers, databases (for faster join operations and keeping track of changes), caching, and other network related scenarios [34]. Recently, this technique was also used to tackle the detection of duplicate data in streams in a variety of scenarios [19, 51, 88, 116]. Such applications included the detection of duplicate clicks on pay-per-click adverts, fraud detection, URI crawling, and identification of distinct users on platforms. Metwally et al. [116] designed a Bloom Filter that applies the "window" principle: sliding windows (finding duplicates related to the last observed part of the stream), landmark windows (maintaining specific parts of the stream for de-duplication), and jumping windows (a trade-off between the latter two window types). Deng and Rafiei [51] go a step further than [116] and propose the Stable Bloom Filter, guaranteeing good and constant performance of filters over large streams, independent of the streams' size. Bera et al. [19] present a novel algorithm modifying Bloom Filters using *reservoir sampling* techniques, claiming that their approach not only provides a lower *false negative rate* but is also more stable than the method suggested in [51].

Random sampling, in different forms, is often used as an alternative to complex algorithms to provide a quick yet good approximation of results [151]. Sample-based approaches such as the latter were used to assess the quality of Geographic Information System data [139, 158]. Xie et al. [158] describe different sampling methods for assessing geographical data. In their approach, Saberi and Ghadiri [139] sampled the original base geographical data periodically. The authors in [93] propose how data quality metrics can be designed to enable (1) the assessment of data quality and (2) analyse the economic consequences after executing data quality metrics. They suggest sampling the dataset attributes to get an estimate measure for the quality of the real-world data.

Lately, various efforts have been made to estimate values within big networks, such as *estimating the clustering coefficient* [62] or calculating the average degree of a network [42]. Hardiman et al. [62] provide an estimator to measure the network size and two clustering coefficient estimators: the network average (local) clustering coefficient and the global clustering coefficient. These measures were applied on public datasets such as DBLP, LiveJournal, Flickr and Orkut. Similarly, Dasgupta et al. [42] calculate the average degree of a network using similar public domain datasets. As Guéret et al. pointed out in [61], network measures can be exploited to assess Linked Data with regard to quality, as Linked Data uses the graph-based RDF data model.

3.5 Previous Efforts in Detecting Outliers in RDF Knowledge Bases

Various research efforts have tackled the problem of detecting incorrect RDF statements using different techniques. These include *statistical distribution* [127], *schema enrichment* [148, 161] and *crowd-sourcing* [2, 153]. Outlier detection techniques such as [156] are used to validate the correctness of data literals in RDF statements, which is out of the scope of this research as our approach considers only statements where the subject and object are resources.

Statistical Distribution Paulheim et al. [127] provide an algorithm based on the statistical distribution of types over properties in order to identify possibly faulty statements, whose subject and object are

resources and not literals. Statistical distributions were used in order to predict the probability of the types used on a particular property, thus determining a confidence value to verify the correctness of a triple statement. Their three step approach first computes the frequency of the predicate and object combination in order to identify those statements that have a low value. Cosine similarity is then used to calculate a confidence score based on the statement's *subject type* probability (i.e. the probability that a particular resource be of type, for example, `foaf:Person`) and the *object type* probability. Finally, a threshold value is applied to mark those statements that are potentially incorrect. The authors report a precision of above 90% (0.9), which is comparable to the precision of our approach when using the *type-selective* reservoir sampler. Whilst their approach is said to identify and remove around 13,000 incorrect triples from DBpedia, we cannot compare how their approach fares with regard to recall per property, due to the lack of evidence, as we describe in Section 8.2.5. This means that we cannot quantify the number of false negatives produced by this approach. Our approach uses semantic similarity to identify whether a statement could be a possibly incorrect statement or not, instead of statistical distribution probabilities. Therefore, our similarity approach takes into consideration the semantic topology of types and not just their statistical usage.

Schema enrichment Schema enrichment is also a popular technique to detect incorrect statements. Töpper et al. [148] first automatically enrich a knowledge base schema (their use case being DBpedia) with additional axioms (including domain-range restrictions and class disjointness) before detecting incorrect RDF statements in the knowledge base itself. Such an approach requires external knowledge (in this case Wikipedia) in order to enrich the ontology. Similarly, Zaveri et al. [161] apply a semi-automated schema enrichment technique before detecting incorrect triples.

Crowdsourcing *WhoKnows?* [153] is a crowdsourcing game where users, possibly unknowingly, contribute towards identifying inconsistent, incorrect and doubtful facts in DBpedia. Such crowdsourcing efforts ensure that the quality of a dataset can be improved with more accuracy, as a human assessor can identify such problems even from a subjective point of view. During the evaluation, the users identified 342 triples that were potentially inconsistent from a set of overall 4,051 triples. These were distributed as follows: 77 identified incorrect triples were generated due to bugs in the code, 144 were wrongly identified as incorrect triples (false positives) and 121 were verified as true wrong triples (true positives). Based on these values, the precision (without the 77 incorrect triples) is around 0.46. A similar crowdsourcing effort was undertaken by Acosta et al. in [2]. They used pay-per-hit micro tasks as a means of improving the outcome of crowdsourcing efforts. Their evaluation focuses on checking the correctness of the object values and their data types, and the correctness of interlinking with related external sources, thus making it incomparable to our approach. In contrast to crowdsourcing, our preliminary approach gives a good precision in identifying outliers without the need of any human intervention, while at the same being time efficient (e.g. both property dumps used in the evaluation had over 10k triples and it takes around ± 3 minutes to compute outliers). Nonetheless, at some point, human expert intervention would still be required (in our approach) to validate the correctness of the detected outliers, but with any (semi-)automatic learning approaches, human intervention is reduced.

3.6 Studying the Quality of the Data on the Web

Empirical studies encourage stakeholders to engage in further discussions on how to improve the state, in this case of linked datasets, in order to improve, for example the overall quality. In this section, we review

literature that analyses the quality of various aspects of Linked Data, as a prequel for the large-scale analysis described in Chapter 9.

In [82], Hogan et al. crawled and assessed the quality of around 12 million RDF statements. The main aim was to discuss common problems found in RDF datasets, and possible solutions. More specifically, this work aimed at uncovering errors related to accessibility, reasoning, syntactical and non-authoritative contributions. The authors also provided suggestions on how publishers can improve their data, so that the consumers can find "higher quality" datasets.

In a follow up article [80], Hogan et al. conduct a larger empirical study on Linked Data conformance, with around 1 billion quads (i.e. triples + graph identifier) assessed. The aim of this study was primarily to define a number of quality metrics from various best practices and guidelines, and to assess the level of conformance of the assessed datasets against these metrics. Our work overlaps with seven quality metrics defined in [80]: (i) avoiding blank nodes; (ii) keeping URIs short; (iii) avoiding prolix features; (iv) re-using existing terms; (v) dereferenceability of resources; (vi) usage of external URIs; and (vii) human-readable metadata. The metrics in our assessment are similar to those in [80], with some modifications as explained in Section 9.3. Nevertheless, the conclusions from [80] are more or less the same, four years later, that publishers might forgo certain quality guidelines as they might be impractical. This can be seen from the distribution of quality metric values amongst the datasets, in both studies.

Buil-Aranda et al. [8] conducted a number of long-term experiments, mostly related to availability quality metrics (as classified in [160]) on around 480 SPARQL endpoints. The authors report that only one third of the endpoints have descriptive metadata such as VoID and service descriptions[7], whilst the query response performance varies widely from one endpoint to another. Our experiments confirm the performance variation and show that no single solution is available for streaming all triples directly from the endpoint (cf. Section 9.3.6). The authors also propose SPARQLES[8], a tool for monitoring the availability of public SPARQL endpoints (among other tests). With SPARQLES, consumers can make informed decisions more easily on whether a certain SPARQL endpoint is reliable and suitable for the task at hand.

In a recent study Assaf et al. [10] shed light on the metadata availability in the Linked Open Data Cloud. This metadata was used in our dataset acquisition process. In [10], the metadata is checked for general, access, ownership and provenance information. The authors conclude, that metadata quality is in a bad condition. More specifically, licensing and accessibility metadata contains noisy data, thus resulting in incorrect information. We discuss the quality of LOD Cloud metadata in more detail in Section 9.1.

Suominen and Mader, in [145], define a number of quality metrics in order to assess SKOS vocabularies with the aim of identifying their re-use in applications. The assessment is based on three categories: (1) labeling and documentation; (ii) structural issues (e.g. class disjoint issues); and (iii) Linked Data issues (e.g. invalid URIs). The authors reported that most of their representative SKOS vocabularies contained structural errors, and presented a set of correction algorithms to address such issues.

[7] http://www.w3.org/TR/sparql11-service-description/. Date Accessed 12th June 2016
[8] http://sparqles.ai.wu.ac.at/. Date Accessed 12th June 2016

Part II

A Scalable Streaming Approach for Assessing Linked Data Quality

With the aim of supporting stakeholders to assess the quality of linked datasets, this part describes the methodology and a scalable infrastructure for enabling Linked Data quality assessment. In Chapter 4 we present Luzzu, a generic open-source framework that streams Linked Data triples from different sources (data dumps, SPARQL endpoints, or resilient distributed datasets), detects different quality problems (according to the metrics defined by the stakeholder), and presents interoperable results in form of quality problems (for which the stakeholder can react upon) and quality metadata (that can be attached to the dataset for consumption). In Chapter 5 we present the Luzzu Quality Metric Language (LQML), a domain specific language for defining quality metrics that can be used in Luzzu. LQML allows quality assessors to define quality metrics in a declarative fashion using a small set of defined constructs, without having the need to learn a third-generation language or how to create SPARQL queries. Luzzu and LQML's simplicity are aimed to encourage both Linked Data and non-Linked Data users to assess the quality of linked datasets, thus attempting to widen the Linked Data audience in other communities. The work in these chapters contribute to research question 1 (RQ1) defined in Section 1.3. The research in this part is crucial to this thesis, as it forms the basis for answering the main research question.

Chapter 4 is based on the following publications:

- **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Methodology and Framework for Linked Data Quality Assessment.* In ACM Journal of Data Information Quality. (To Appear);
- **Jeremy Debattista**, Sören Auer, Christoph Lange. *Luzzu - A Framework for Linked Data Quality Assessment.* IEEE Tenth International Conference on Semantic Computing (ICSC) 2016, 124-131, IEEE;
- **Jeremy Debattista**, Christoph Lange, Sören Auer. *Luzzu - A Framework for Linked Data Quality Assessment*. ISWC 2015 Posters and Demonstrations Track, CEUR-WS.org vol. 1468, 2015.

CHAPTER 4

Luzzu - A Methodology and Framework for Linked Data Quality Assessment

In Section 1.1 we described a challenge related to the detection of quality problems. Linked Data has different quality aspects (refer to [160] for a comprehensive list of Linked Data quality metrics), but not all aspects are useful for a consumer to find a dataset that is *fit for use*. For example, if a consumer needs to build a navigation app for bike routes and requires spatially-located things to outsource maps and points of interest from other datasets, then this consumer would need geographic datasets that are complete with regard to geographic latitude, longitude, and altitude. On the other hand, this consumer might not consider quality aspects such as interlinking between resources to be as important. The challenge tackled in this chapter is related to the detection of various quality problems in a scalable fashion.

In this chapter, we present a framework for assessing Linked Data quality, with the goal of being *scalable*, *extensible*, *interoperable*, and *customisable*. Regarding *scalability*, we follow a stream processing approach, whilst for *extensibility* we provide a simple, declarative domain specific language for the integration of various quality measures. The latter is discussed in more detail in Chapter 5. With regard to *interoperability*, Luzzu is accompanied by a set of ontologies for capturing quality related information for re-use, including quality measures, issues and reports, that can be re-used in other semantic frameworks and tools. Even with the possibility to automatically compute quality measures, the large number of quality dimensions and measures complicates the user's task of judging whether a dataset is fit for use. We address this problem by developing an approach for a user-driven quality-based weighted ranking of datasets, allowing users to select and give importance to some *custom* quality measures over others. In Section 4.5 we evaluate the performance of the stream processors. Section 3.1 gives an overview of related quality assessment approaches for Linked Data. Furthermore, in Chapter 9, we demonstrate the framework's applicability, by assessing the quality of 130 linked datasets, portrayed in the Linked Open Data Cloud. The realisation of this framework is the foundation of this thesis, with other components (described in the next chapters) building upon it.

The main contributions of this chapter are:

1. the re-definition of a data quality life cycle (cf. Section 4.1) in terms of co-evolution of linked datasets;
2. a conceptual methodology for assessing Linked Data quality (cf. Section 4.2) together with a formalisation of this conceptualisation (cf. Section 4.3);
3. Luzzu, a quality assessment framework for Linked Data (cf. Section 4.4), which includes a comprehensive library of implemented quality metrics, a declarative language for creating addi-

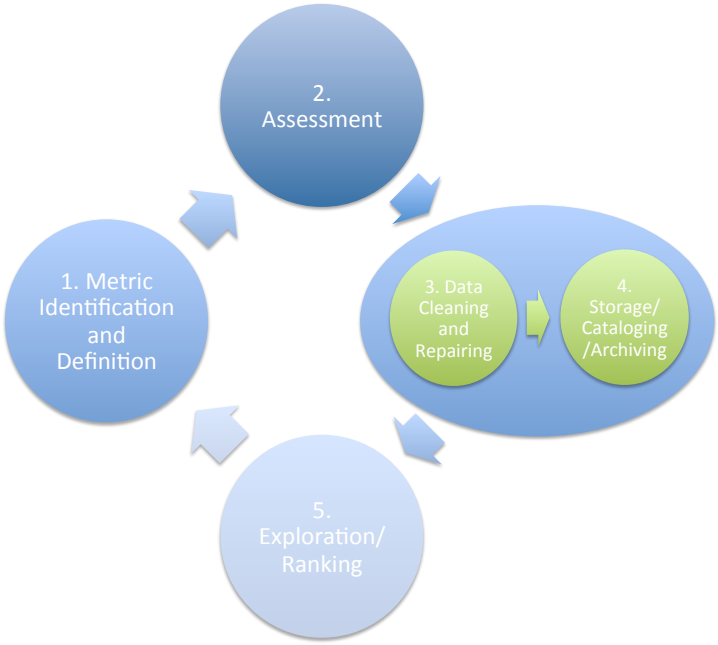

Figure 4.1: The stages of the Data Quality life cycle.

tional domain specific quality metrics, and a set of comprehensive ontologies for capturing and exchanging data quality related information.

4.1 The Data Quality Life Cycle

We deem that data quality can not be tackled in isolation, but should be considered holistically involving other stages of the data management life cycle. Quality assessment on its own cannot improve the quality of a dataset. In particular, data curation plays an important role in maintaining datasets ensuring that data can be preserved and re-used in the future. It is also a data renewing process that warrants "knowledge workers to have access to accurate, high-quality and trusted [data]" [39].

Quality assessment provides the user with the current quality status of a dataset and (in some frameworks, such as our Luzzu framework presented in Section 4.4) with a quality problem report based on the assessed metrics. Prior works have defined a life cycle for different aspects of (linked) data (cf., e.g., [11, 12]). In this section, we examine the *quality* aspects of this life cycle in further depth and thus propose a data quality life cycle (Figure 4.1), which specifically covers the phases from the assessment of data, to cleaning and storing. This life cycle builds upon the Linked Data quality assessment methodology presented in [138], ensuring that the life cycle does not stop short at the quality assessment and subsequent cleaning. In contrast to the related methodology, the proposed life cycle shows that quality assessment and improvement of datasets is a continuous process. This is similar to how Batini et al. in [16] make provisions for monitoring data quality dimensions periodically.

The proposed data quality life cycle includes:

1. Metric Identification and Definition - Quality assessment is different for every domain and sometimes even between different use cases in a single domain. In this process, domain experts (who can also be knowledge engineers) identify a set of both domain-independent metrics and domain-specific quality metrics that they deem suitable for the task at hand. For example, datasets in the geographical domain

require that each resource that is locatable on a map has the properties *geo:long* and *geo:lat* defined. After the identification process, the metrics are defined so that the concerned dataset could be assessed upon them. In Luzzu, domain experts can define metrics either in a simple declarative fashion using the Luzzu Quality Metric Language (LQML) or in a more complex imperative way (cf. Section 4.4.1).

2. Assessment - During this stage, a dataset is assessed against the quality metrics identified and defined in the previous life cycle stage. Figure 4.2 depicts a typical quality assessment workflow. The Luzzu framework (cf. Section 4.4) provides the necessary tools to assess a dataset and provide both quality metadata and a quality report, which will be used in the following stage of the life cycle. The Quality Assessment process is the main focus of this work.

3. Data Repairing and Cleaning - Ensuring a higher quality dataset and an ongoing evolution of data requires data repairing and cleaning to be performed. Data repairing deals with problems concerning violations of logical constraints in a dataset, while data cleaning aims at rectifying errors in a dataset which render the dataset incorrect with regard to syntax or semantic aspects not covered by the RDFS or OWL logics [37]. Data cleaning can be performed in various ways, such as using crowd-sourcing techniques [98]. The quality problem report (cf. Section 4.4.3) generated by the Luzzu framework can be fed into a data cleaning application (assuming that this application understands the semantics of the report's vocabulary) that handles this semi-automatic task. Currently, in the Luzzu quality assessment framework, we do not employ any (semi-)automatic data cleaning or repairing techniques, but we provide generic problem reports (following an assessment) that can be used within such cleaning frameworks.

4. Storage, Cataloging and Archiving - At this stage, datasets, possibly cleaned, are stored and archived together with their quality metadata. Data portals make these datasets available to the data consumers. Having quality metadata available with the datasets themselves, these portals can easily catalogue the datasets based on different quality criteria. This stage of the life cycle is beyond the scope of the Luzzu quality assessment framework, but the quality reports generated by Luzzu can be archived together with the datasets to enable tracking of the quality evolution. In particular, by analysing the quality evolution over time, various strategies for data curation (e.g. based on crowd-sourcing, domain expert contributions, or application of automated techniques) can be more effectively evaluated and compared.

5. Exploration and Ranking - Finding a dataset suitable for a use case is sometimes a daunting task. Currently, portal engines such as *CKAN*[1] provide faceted browsing and ranking features to search within large collections of datasets using particular tags. Having quality metadata attached to datasets, such human-friendly data portals will provide the possibility to filter and rank datasets based on their quality, thus facilitating the selection of the most suitable dataset for a certain use case. The flexibility of the Dataset Quality Vocabulary (daQ) enables data consumers to track and follow quality improvements over time (possibly over iterations of this data quality life cycle) using visualisation techniques such as those provided in *CubeViz* [109], and *Payola* [76]. The Luzzu Quality Assessment framework provides an interface for ranking (cf. Section 4.4.4) datasets based on their quality metadata, while it also generates (cf. Section 4.4.3) multi-dimensional quality metadata.

4.2 A Conceptual Methodology for Assessing Linked Data Quality

Based on the the data quality life cycle overview described in Section 4.1, we define a conceptual methodology for assessing Linked Data quality. Given an application scenario and candidate datasets, a number of quality metrics can be chosen to characterise the datasets' *fitness for use*, i.e. which datasets are useful for the task at hand, and to what extent.

In general, our methodology can be described as follows:

[1] http://www.ckan.org. Date Accessed 3rd October 2016

1. **Identify the quality measures** that are important for the task at hand. For example, a data mining task in bioinformatics requires different quality metrics than for the choice of appropriate datasets for open domain question answering.

2. Agents (a machine or a human) can either **re-use** existing quality metrics, **or define** new quality ones (following the quality metric pattern (*QMP*) defined in Section 4.3.5). A quality metric can be defined formally in a bottom-up way by first defining the value of a metric on the atomic level of single triples, and then defining the aggregation of such values over all triples of a dataset. Agents can promote re-use of defined *QMP*s by sharing them in a pool of metrics, grouped by dimension and category and enriched with further semantic descriptions.

3. **Preparing the quality assessment**. Metrics are computed by applying metric processors to each dataset under assessment. Metrics may be computed exclusively by examining a dataset, or require external resources such as gold standards, with which the respective metric processors have to be initialised. Secondly, at this point we need to prepare for processing each dataset, depending on whether the dataset is accessible as a data dump, through a SPARQL endpoint, or at some intermediate level of Linked Data Fragments [150] between these two extremes. At this state of initialisation, we are ready to apply the metric processors to any candidate dataset, for example the most recent versions of datasets D_1 and D_2, and to re-apply them to any other dataset, for example to an additional dataset D_3, or to subsequent versions of D_1 and D_2.

4. **Running the quality assessment**. Based on the bottom-up definition of quality metrics (from triples to datasets), the quality of a dataset with regard to the metrics chosen can be assessed by streaming every individual triple of the dataset under assessment to the metric processors initialised previously. One such run of a quality assessment will result in a record of quality metadata for each dataset assessed, together with a problem report, which can be as detailed as pinpointing specific problems at the level of individual triples.

5. **Assessment representation**. The quality metadata and problem reports can be used in the following ways:
   - *Immediate use*:
     - Quality metadata can be used for ranking different datasets by their (weighted) values with regard to the relevant metrics (choice of weights depends on use case);
     - Problem reports identify problematic triples that support the eventual cleaning of a dataset.
   - *Mid- to long term use*:
     - Quality metadata is stored together with the dataset that helps cataloging and preservation of the dataset for future retrieval tasks.
     - Representing external resources in the quality metadata ensures replicability of quality results.

4.3 Formalisation of the Conceptual Methodology

In this section we define the basic concepts required in a framework for assessing the quality of linked datasets. Our methodology is based on the concepts of atomic quality metrics and quality metric patterns. A dataset-level quality metric is defined as an aggregated value of the per-triple values, of either atomic quality metrics or state-aware quality metrics.

4.3.1 Datasets

A linked dataset D is a set of RDF triples belonging to one or more graphs; in the latter case the triples are extended to quads by a fourth component that provides the graph URI or other contextual information. In the simple case that all triples are in the same graph, a dataset is formally defined as follows:

$$\begin{aligned} D &\subseteq IR \setminus \text{literals} \times IP \times IR \\ D &= \{t_1, \ldots, t_n\} \\ t_i &= (s_i, p_i, o_i), 1 \leq i \leq n \end{aligned} \tag{4.1}$$

where $n \in \mathbb{N}_0$ is the total number of triples in D. IR refers to the set of all possible resources, including literals, and IP refers to the set of all properties, as specified in the semantics of RDF [72].

To facilitate the application of functions to triples in a dataset, as well as algorithmic implementations of quality metrics, we may equivalently treat a dataset as a *vector* $(t_1, \ldots, t_n)$, which is assumed to be free of duplicates and to have an arbitrary order of components.

Technically, a dataset D is provided as a URI that resolves to a serialised RDF data dump that can be retrieved following an HTTP content negotiation, or as a SPARQL endpoint.

4.3.2 Atomic Quality Metric (AQM)

An Atomic Quality Metric (AQM) is a function that takes an RDF triple $t_i = (s_i, p_i, o_i)$ and returns a single value of a simple datatype.

$$q : IR \setminus \text{literals} \times IP \times IR \rightarrow \{\mathbb{R} \cup \mathbb{Z} \cup \mathbb{B} \cup \ldots\} \tag{4.2}$$

Simple datatypes include real numbers, integers, and booleans.

4.3.3 Global Aggregation of Atomic Quality Metrics

Aggregate functions extend the definition of quality metrics from triples to datasets. A global aggregate function $\bar{\cdot}$ over an AQM, where $\cdot$ is a placeholder for a concrete AQM function, is defined like in database query languages; it takes n arguments of the same simple datatype and returns a single simply-typed aggregate value. The argument vector of a global aggregate function $\bar{\cdot}$ must have been computed by applying the same AQM function, e.g., the AQM function q, to each triple $t_i, i = 1, \ldots, n$ of a dataset vector, and its return value typically has the same type T as each of its arguments, i.e.

$$\bar{q} : T^n \rightarrow T, T \in \{\mathbb{R} \cup \mathbb{Z} \cup \mathbb{B} \cup \ldots\} \tag{4.3}$$

We require the value of a global aggregate function to be invariant over permutations of the components of the triple vector. Concrete atomic global aggregate functions include *count*, *median*, *maximum*, *sum*. Contrarily to the other functions, *count* always returns an integer value. Compound global aggregate functions are typically formed by counting the frequency of specific values in the argument vector, such as the number of booleans of value `true`, representing the number of triples that match a certain pattern, or by normalising an aggregate value over the count of all triples in the dataset. For example, the arithmetic mean is defined as a sum of values normalised over the count of values. Similarly, it is also common to compute the *ratio* of triples that match a pattern.

4.3.4 State-aware Inductive Aggregation

More complex dataset-level quality metrics may require an iterative computation, where for each triple not only the value of an AQM is taken into account, but also a *state* σ, which, in the most general case, can be a complex data structure accumulated over all triples that have been processed previously in the computation of the same metric. Here, aggregation is defined inductively, starting with an initial state σ_0 and terminating in the final aggregate value $\bar{q}_n$:

$$\begin{aligned}(\sigma_1, \bar{q}_0) &:= \sigma_0 \\ (\sigma_i, \bar{q}_i) &:= \overline{\sigma_{i-1}, q(t_i)}, i = 1, \ldots, n\end{aligned} \tag{4.4}$$

Global aggregation of AQMs can be defined as a special case of state-aware inductive aggregation; the following definition gives an example for the ratio of all triples that match a certain pattern:

$$\begin{aligned}\sigma_{r0} &:= 0 \\ \overline{\sigma_{ri-1}, q(t_i)_r} &:= \sigma_{ri-1} + \begin{cases}\frac{1}{n} & \text{if } q(t_i) = \texttt{true} \\ 0 & \text{if } q(t_i) = \texttt{false}\end{cases}\end{aligned} \tag{4.5}$$

4.3.5 Quality Metric Pattern (QMP)

Quality Metric Patterns (QMPs) help to define quality metrics beyond simple global aggregation of AQMs in more practical terms than state-aware inductive aggregation. A QMP is a triple[2], $\mu = (\alpha, \lambda, \omega)$, where α is a precursor function, λ is an assessment rule, and ω is a successor function. A *basic* QMP has null values for α and ω, whilst in a *complex* QMP at least one of the non-assessment functions is not null.

Precursor Function The precursor function α is executed to initiate the quality assessment of a dataset and yields an initial state σ_0. Such functions prepare the metric's environment, i.e. the state of the metric processor, with objects that are required for the assessment, such as loading gold standards or retrieving common vocabularies from a web API[3].

Assessment Rule An assessment rule λ is executed in every iteration of a metric's computation; it is defined as a partial map from a set of conditions C to a set of actions A:

$$\lambda : C \to A, c \mapsto a \tag{4.6}$$

A condition takes the current RDF triple $t_i = (s_i, p_i, o_i)$ and returns a truth value. A condition can either be a simple equality constraint, for example $o_i == \texttt{rdfs:Literal}$, or an elaborated constraint, for example $\texttt{isDereferenceable}(s_i)$. Whilst the former example checks if the triple's object is an RDF literal, the latter refers to an extension function whose semantics is understood by the framework. Finally, a condition can be any combination of simpler conditions using logical connectives.

A mapped action is triggered if its condition is satisfied. An action may return a simple AQM value for the triple being assessed, or it may return an aggregated value over all triples seen so far (by accessing the current state), or it may influence the next state by, e.g., incrementing a counter.

[2] Not to be confused with an RDF Dataset triple

[3] For example from Linked Open Vocabularies; http://lov.okfn.org/. Date Accessed 3rd October 2016

Successor Function The successor function ω is executed after the final iteration n of assessing a dataset. It may comprise complex computations on the overall state accumulated, such as statistical tests over all numerical literals collected from the dataset, and it may implement a special final aggregate function.

QMPs as State-aware Inductive Aggregation A QMP $\mu = (\alpha, \lambda, \omega)$ can be interpreted as a state-aware inductive aggregation as follows:

$$\begin{aligned} \sigma_0 &:= \alpha \\ (\sigma_i, \bar{q}_i) &:= \begin{cases} a(\sigma_{i-1}, t_i) & \text{if } c(t_i) = \texttt{true} \text{ for some } c \mapsto a \\ (\sigma_{i-1}, \bar{q}_{i-1}) & \text{otherwise} \end{cases}, i = 1, \ldots, n \\ \bar{q}_{n+1} &:= \omega(\sigma_n, \bar{q}_n) \end{aligned} \tag{4.7}$$

4.3.6 Dataset Quality Assessment (DQA)

A Dataset Quality Assessment (DQA) is a tuple (D, M), where D is a linked dataset that will be assessed for quality using a set of metrics M.

M is a set of defined metrics on which the dataset D is assessed upon. Each metric $M_i \in M$ is given as a specific state-aware inductive aggregation (4.4). In practice, this may often be a specific global aggregation of a specific AQM (4.3), or a QMP with specific values for precursor, assessment rule and successor.

4.3.7 Metric Instantiation and Triple Streaming

The instantiation function I maps the set of metrics M to a pool of threads Ξ such that each metric is computed in a thread of its own:

$$\begin{aligned} &I : M \to \Xi,\ M = \{M_1, \ldots, M_p\},\ \Xi = \{\xi_1, \ldots, \xi_p\},\ p \in \mathbb{N} \\ &M_j \mapsto \xi_j, 1 \leq j \leq p \end{aligned} \tag{4.8}$$

The DQA binds D to the thread pool Ξ such that each triple in the dataset is streamed to all threads and thus all instantiated metrics; therefore, for each metric a quality assessment value, i.e. a final aggregate value over all triples, can be computed:

$$\begin{aligned} \forall t_i \in D. \forall \xi_j \in \Xi. \exists \tau. \xi_j \text{ is processing } t_i \text{ at time } \tau \\ \therefore \forall M_j \in M. M_j(D) = \bar{q}_n(t_1, \ldots, t_n) \end{aligned} \tag{4.9}$$

4.3.8 User-Driven Ranking

In the spirit of *fitness for use*, we propose a *user-driven ranking algorithm* that enable users to assign weights to their preferred quality categories, dimensions or metrics, as deemed suitable for the task at hand. User-driven ranking is formalised as follows:

Let $v : F_m \to \{\mathbb{R} \cup \mathbb{N} \cup \mathbb{B} \cup \ldots\}$ be the function that yields the value of a metric – most commonly a real number, but it could also be an integer, a boolean, or any other simple type. These simple types can be converted into a numeric format, thus different metrics can be combined for the fitness for use ranking. For example, boolean type can be converted to 1 (if true) or 0. On the other hand, a metric with

a *datetime* type value, can be converted to a number by calculating the difference between *now* and the assessed value.

Ranking by Metric Ranking datasets by individual metrics requires computing a weighted sum of the values of all metrics chosen by the user. Let m_i be a metric, $v(m_i)$ its value, and θ_i its weight ($i = 1, \ldots, n$), then the weighted value $v(m_i, \theta_i)$ is given by:

Definition 1 (Weighted metric value):

$$v(m_i, \theta_i) := \theta_i \cdot v(m_i)$$

The *sum* $\sum_{i=1}^{n} v(m_i, \theta_i)$ of these weighted values, with the same weights applied to all datasets under assessment, determines the ranking of the datasets.

Ranking by Dimension When users want to rank datasets in a less fine-grained manner, they assign weights to quality dimensions. The weighted value of the dimension D is computed by evenly applying the weight θ to each metric m in the dimension.

Definition 2 (Weighted dimension value):

$$v(D, \theta) := \frac{\sum_{m \in D} v(m, \theta)}{\#D} = \theta \frac{\sum_{m \in D} v(m)}{\#D}$$

Ranking by Category A category C is defined to comprise one or more dimensions D, thus, similarly to the previous case, ranking on the level of categories requires distributing the weight chosen for a category over the dimensions in this category and then applying Definition 2.

Definition 3 (Weighted category value):

$$v(C, \theta) := \frac{\sum_{D \in C} v(D, \theta)}{\#C}$$

4.4 Luzzu Quality Assessment Framework

Luzzu[4] is a quality assessment framework for Linked Data. Luzzu implements the formal concepts introduced in Section 4.3. The rationale of Luzzu is to provide an integrated platform that: (1) assesses Linked Data quality using a library of generic and user-provided domain specific quality metrics in a scalable manner; (2) provides queryable quality metadata on the assessed datasets; (3) assembles detailed quality reports on assessed datasets. Furthermore, Luzzu is an infrastructure that:

- can easily be extended by users by defining custom, domain-specific metrics;
- implements quality-driven dataset ranking algorithms facilitating use case driven discovery and retrieval.

Figure 4.2 illustrates the workflow, based on the conceptual methodology defined in Section 4.2, of Luzzu. When an agent (machine or human) initiates quality assessment for a Linked Dataset or a SPARQL endpoint, it selects a number of quality metrics. Chosen metrics, together with external resources such as

[4] Sources: https://github.com/EIS-Bonn/Luzzu; Website: http://eis-bonn.github.io/Luzzu/

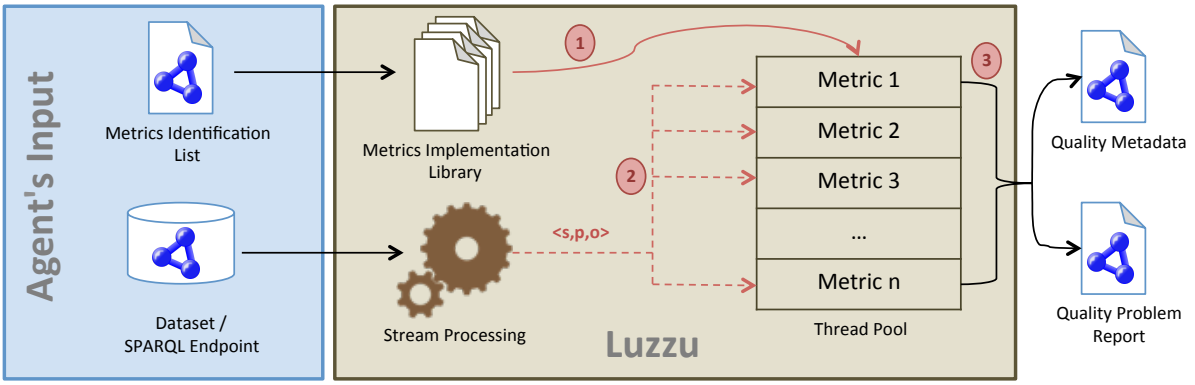

Figure 4.2: Quality Assessment Workflow – The left side displays the input sources from the agent, whilst the right side shows the process (1) the initialisation (including the execution of any precursor functions) of the metric implementation in individual threads; (2) the streaming of triples; and (3) the semantic enrichment of quality values as metadata and quality problem report, after executing any successor function.

gold standards, are initialised in individual threads. A dataset processor streams triples, either from the linked dataset data dump (given in any RDF serialisation, or, in the near future, in the compressed HDT format [9]) or the SPARQL endpoint, to each initialised metric (cf. Section 4.4.2). Once all triples are processed, a quality assessment value for each configured metric is calculated. These values are then stored as quality metadata for the assessed dataset, whilst problematic triples are reported back to the agent (cf. Section 4.4.3).

4.4.1 Defining Quality Metrics

In this section, we focus on the assessment rule λ as defined in Section 4.3.5. Given an RDF triple, the metric checks if one or more constraints are satisfied and triggers one or more actions. In Luzzu, quality metrics can be created in two ways: 1) by implementing a Java class adhering to the specified quality metric interface, or 2) by using the declarative *Luzzu Quality Metric Language* (LQML - cf. Chapter 5).

Although Luzzu provides a SPARQL endpoint processor, no metric is implemented as a SPARQL query. The reason for opting against SPARQL metrics (e.g. as used in RDFUnit [99]) is that constraints checked by metrics implemented in SPARQL are limited to the data and its schema, thus ignoring other important Linked Data quality measures, such as the identification of outliers or detecting resource performance issues. Nevertheless, any metric that can be expressed in a SPARQL query can be defined as a metric for Luzzu, as we can draw similarities between the atomic quality metric and the SPARQL basic pattern triple.

Luzzu Quality Metric Language

LQML is a domain specific language (DSL) that enables knowledge engineers to declaratively define quality metrics whose definitions can be understood more easily than an imperative defined metric. It offers shorthand notations, abstractions and expressive power. LQML is designed in a way that metrics can be written by non-programmers who are experts in the domain [52, 85]. Hudak [85] suggests that DSLs have the potential to improve productivity in the long run and with LQML we aim to simplify Linked Data quality assessment.

Our proposed domain specific language is based on the *language invention* design pattern [115], where we fuse a number of common specific terms, which were identified when we analysed quality metrics

across different dimensions, together with variable binding expressions used in the syntax of SPARQL (i.e `?s ?p ?o`) to refer to specific elements in a triple. The Luzzu Quality Metric Language is discussed in more detail in Chapter 5.

In order not to limit LQML expressiveness, programmers can extend it by custom functions for complex tasks, some of which we are providing already. Examples include network operations (e.g. `IsDereferenceable`) or logical consistency checks (e.g. `hasValidInverseFunctionalPropertyUsage` – a function that checks if a property typed as an `owl:InverseFunctionalProperty` is violated by a resource), which can then be used in a metric definition's conditions and actions. This follows the best-practice of other domain specific languages, such as SPARQL or XPath, which support user-defined functions[5]. Listing 4.1 shows an LQML defined metric that checks if the subject and the object of a triple are dereferenceable (i.e. has a `303 See Other` HTTP code, or is a hash URI).

Examples of Quality Metrics

Quality Metrics in Luzzu follow the definitions described in Section 4.3. With the help of some examples, we show the link between the definitions and implemented metrics.

A Simple Metric Listing 4.1 shows the LQML definition of the Dereferenceability metric (cf. Section **??**). The elements of this definition can be explicitly linked to the concepts defined in Section 4.3.2 and Section 4.3.3. The statements `var.x = match{(isURI(?s) && custom.IsDereferenceable(?s))} => action{countUnique(?s)};` and `var.y = match{(isURI(?o) && custom.IsDereferenceable(?o))} => action{countUnique(?o)};` correspond to an atomic quality metric (AQM) that, given a triple, returns 0 if no match condition is satisfied, 1 if one of them is satisfied, or 2 if both are satisfied. The metric's `finally` statement corresponds to a compound global aggregate function over AQMs (Section 4.3.3). The LQML keywords `totaltriples` and `add` correspond to the *count* and *sum* atomic global aggregate functions; the `ratio` keyword performs an arithmetic operation over these atomic values.

```
def{DereferenceabilityMetric}:
  metric{<http://purl.org/eis/vocab/dqm#Dereferenceability>};
  label{"Dereferenceability of Resources"};
  description{"Measures the number of valid dereferenceable resources in a dataset
    "};
  rule{
    var.x = match{isURI(?s) & custom.IsDereferenceable(?s)} => action{countUnique(?
    s)};
    var.y = match{isURI(?o) & custom.IsDereferenceable(?o)} => action{countUnique(?
    o)};

  } ;
  finally{ratio(add(var.x,var.y), totalTriples)}.
```

Listing 4.1: Using custom functions in LQML.

Complex Metric Computation Involving State Whilst AQMs work at the granularity of triples, some metrics have to be assessed on a higher granularity level, e.g., in terms of *resources*, i.e., on sets of triples having the same subject. State-aware inductive aggregation (cf. Section 4.3.4) is specifically

[5] See `http://www.w3.org/TR/sparql11-query/#extensionFunctions` and `http://www.w3.org/TR/xpath-30/`. Date Accessed 3rd October 2016

defined for such cases. The values of such metrics are computed incrementally, using a data structure that accumulates intermediate values.

An example of such a metric is the Extensional Conciseness metric (cf. Section 7.1.3), which counts the number of unique instances found in the dataset. An instance is unique to a dataset if no other instance in the same dataset exists with the same set of properties and corresponding values. Therefore, this metric requires the knowledge of all triples of a resource, which therefore must be memorised in a data structure, before being able to check whether it already exists or not.

A Complex QMP A quality metric might require an external resource in order to assess a dataset against it. Such resources are usually loaded before the start of the assessment. This process is formally defined by a complex quality metric pattern (QMP; cf. Section 4.3.5). Population completeness is one metric that requires a gold standard or a reference dataset. Our concrete implementation of population completeness for data cubes defined in terms of the Data Cube Vocabulary [40] requires a *code list*, i.e. a finite set of predefined values, to verify the completeness of observations (if any) in datasets.

A typical complex QMP will load the external resource and other configurations prior to commencing the assessment. For example, the Data Cube Population metric requires a configuration file with the URI of the code list and the data cube dimensions on whose complete population the quality assessment should focus, e.g., whether for every municipality of a country there is an observation (cf. Appendix B for more information on the Data Cube population completeness metric).

4.4.2 Processing Linked Datasets

Luzzu provides a scalable RDF dataset processor which streams a dataset's triples into all initialised metric processors (cf. Figure 4.2 (label 2)). Streaming ensures scalability beyond the limits of main memory, and *parallelisability*, since the parsing of a dataset can be split into several streams to be processed on different threads, cores or machines.

The input dataset can be either a serialised Linked Data dump or a SPARQL endpoint. Any such RDF data source is processed triple by triple. In addition, the framework includes a *Spark*[6] processor (an in-memory equivalent to Hadoop), thus enabling Luzzu to exploit Big Data infrastructures to their full potential. The rationale is to *map* the processing of large resilient distributed datasets (RDD) on multiple clusters, and then using a *reduce* function to populate a queue that feeds the metrics. Metric computations are not associative in general, which is one of the main requirements to implement a MapReduce job; therefore, instantiated metrics are split into different threads in the master node. For example, the *Extensional Conciseness* metric cannot be split into multiple threads or different cluster nodes (as in MapReduce). The reason is that in this metric we check if in a dataset there are two or more resources having the same set of properties and objects, but with different subject URI. If triples are allocated in different threads or cluster nodes, these cannot be accessed and thus possible violating resources are not identified.

Processing SPARQL Endpoints Luzzu provides a dataset processor to assess the quality of a Linked Data knowledge base (KB) from a SPARQL endpoint when dumps are not available. For this job, triple statements are fetched from the KB and streamed to the metrics in the thread pool. Public SPARQL endpoints often truncate results for queries that are very expensive to compute. For example, a query asking for the entire KB in the public DBpedia endpoint will return only the first 10,000 results. Therefore, statements are retrieved in intervals using the SPARQL `OFFSET` and `ORDER BY` modifiers.

[6] http://spark.apache.org/. Date Accessed 3rd October 2016

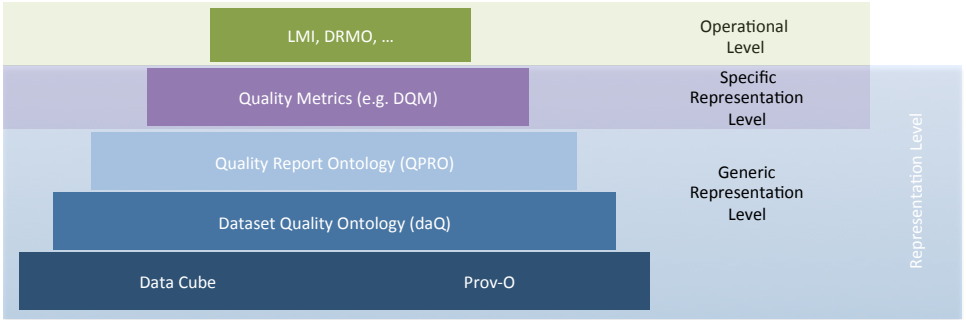

Figure 4.3: The Luzzu ontology stack comprising various levels of quality representation based on the W3C Data Cube and PROV ontologies.

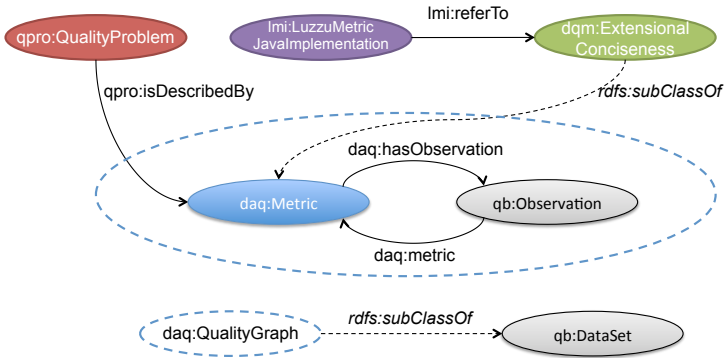

Figure 4.4: Key relationships between the ontologies in the Luzzu ontology stack.

This combination is required as using the `OFFSET` on its own would not be effective since the returning results would not be predictable [66, §15.4]. Assessing the data quality of a knowledge base from a SPARQL endpoint comes with further challenges. For example, if a SPARQL *write* transaction (insert or update) is performed during the assessment, the whole quality assessment might be compromised. Thus, we discourage the use of SPARQL endpoints when possible.

4.4.3 Ontology-Driven Framework

Luzzu employs an underlying semantic knowledge layer to capture the results of assessing the quality of a dataset. The *semantic schema layer* consists of a *representational* (split into generic and specific sub-levels) and an *operational* level.

Figure 4.3 shows the framework's ontology schema stack, where the lower level comprises generic ontologies that form the foundations of the quality assessment framework, and the upper level comprises specific ontologies required for the various quality assessment tasks. The ontologies used in this stack allow for a comprehensive and holistic representation of Linked Data quality information. Figure 4.4 depicts the relationships between the ontologies.

Generic Representation Level

The generic representation level is domain independent, and can be easily re-used in similar frameworks for assessing data quality. The two vocabularies on this level are the Dataset Quality Vocabulary (prefix:

daq, discussed in Chapter 6), which provides queryable metadata, and the Quality Problem Report Ontology for assembling detailed quality reports (cf. Figure 4.2 label 3).

The *Quality Problem Report Ontology (QPRO)*[7] enables the fine-grained description of quality problems found while assessing a dataset. It comprises two core classes: `qpro:QualityReport`, representing a report on the problems detected during the assessment of quality on a dataset, and `qpro:QualityProblem`, representing the individual quality problems contained in that report. Each `qpro:QualityProblem` is described by the following properties:

- `qpro:computedOn` refers to the URI of the dataset on which a certain quality assessment has been performed. This property is attached to a `qpro:QualityReport`;
- `qpro:hasProblem` identifies problem instances in a report and links a `qpro:QualityProblem` to a `qpro:QualityReport`;
- `qpro:isDescribedBy` describes each `qpro:QualityProblem` using an instance of a `daq:Metric`;
- `qpro:problematicThing` represents the actual problematic instance from the dataset. This can be a list (`rdf:Seq`) of resources or of reified RDF statements;
- `qpro:inGraph` refers to the assessed graph, since quality assessments can be performed on multiple named graphs [36];
- `qpro:exceptionDescription` describes a particular quality problem/error raised by the resource or triple in question, causing a quality problem;
- `qpro:extraExceptionProperty` enables the description of extra exception properties, such as the expected value (cf. `actualContentType` and `expectedContentType` in Figure 4.5). This is an abstract property.

The problem report is envisaged to be used by Linked Data cleaning tools, as it identifies all problematic triples. Figure 4.5 depicts the T-Box of the QPRO ontology, together with a typical A-Box example.

[7] http://purl.org/eis/vocab/qpro

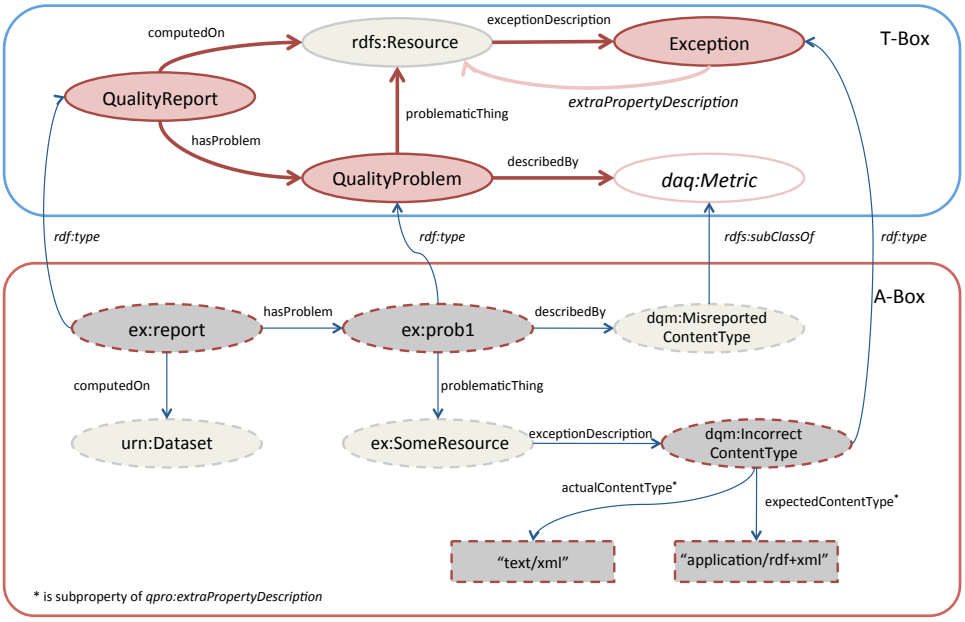

Figure 4.5: An A-Box and T-Box example of the Quality Problem Report Ontology.

Specific Representation Level and Operational Level

The specific representation level consists of semantically defined quality metrics, based on the three layers of the abstract level of daQ [47]. The semantic definition of a metric together with its concrete implementation becomes part of a shared library, thus facilitating re-use of metrics in different Luzzu instances. These semantic definitions are also used to create the quality metadata of a dataset. In Section 6.3.1, we explained how concrete metrics are defined in daQ and how their instances form a quality metadata graph. Complementing Luzzu, we have implemented a number of Linked Data quality metrics[8], most of which were identified in [160].

Luzzu provides a small configuration vocabulary, the Luzzu Metric Implementation (LMI) vocabulary, which enables the linking between a semantically defined metric and its Java or LQML implementation. In order to create a link between the metric and its implementation, an instance can be created using the defined `lmi:LuzzuMetricJavaImplementation`. The two properties `lmi:referTo` and `lmi:javaPackageName` have a range of `daq:Metric` and `xsd:string` (in practice the string reflects the full Java package name of the metric) respectively. Since metrics might have a precursor or successor function, the LMI provides two properties (`lmi:before` and `lmi:after`) whose values can be configured per use case. Furthermore, the LMI vocabulary is used during the invocation of assessment by the agent (cf. Figure 4.2 – Metrics identification List), creating an instance of `lmi:MetricConfiguration` with a list of `lmi:metric`.

4.4.4 User-Driven Ranking

In Luzzu, we implement an algorithm based on the definitions described in Section 4.3.8. The quality metadata following dataset assessment are queried in order to rank datasets based on the user's preference of metrics. The set of datasets fed into ranking can be obtained by cutting any slice out of the cube structure of the quality metadata. For example, one can restrict the "time of assessment" dimension of

[8] These can be downloaded from https://github.com/diachron/quality.

the data cube to the most recent time at which an assessment was performed, or restrict the "dataset" dimension of the data cube to one dataset and analyse its evolution over time. *Quality Profiles* can be created to persist a set of quality measures and weights that can be re-used in Luzzu. Listing 4.2 illustrates one such profile where different weights are given to different categories, based on the user's needs.

```
[
  {
    "type": "category",
    "uri": "http://purl.org/eis/vocab/dqm#Intrinsic",
    "weight": "0.3"
  },
  {
    "type": "category",
    "uri": "http://purl.org/eis/vocab/dqm#Accessibility",
    "weight": "0.0"
  },
  {
    "type": "category",
    "uri": "http://purl.org/eis/vocab/dqm#Representational",
    "weight": "0.2"
  },
  {
    "type": "category",
    "uri": "http://purl.org/eis/vocab/dqm#Contextual",
    "weight": "0.5"
  }
]
```

Listing 4.2: An example of a Luzzu quality profile.

4.4.5 Luzzu Web Interface

On top of this framework, a web interface is provided to assist users in the assessment, and the subsequent analysis of the quality metadata in a visual manner. More specifically, *Luzzu Web* [46] enables the (a) exploration and ranking of quality assessed datasets (Figure 4.6); (b) visualisation of quality metadata (Figure 4.7); and (c) assessment of datasets (Figure 4.8). The exploration and ranking features allow users to search through quality assessed datasets according to their quality criteria fit for their use. Furthermore, quality metadata can be visualised as charts based on the data cube structure definition. A visualisation wizard helps the user to choose the right visualisation type and charts: (a) multiple datasets vs metric; (b) dataset vs metric over time (as displayed in Figure 4.7); (c) quality of dataset. The assessment interface provides the user with step-by-step instructions to start assessing the quality of a dataset or data from a SPARQL endpoint, using the described Luzzu framework. Although the Luzzu framework only requires the input file (i.e. dataset being assessed) to be in a semantic format (e.g. RDF/XML, Turtle, N3, NQuads) or a SPARQL endpoint, any user-custom pre-processing such as sorting (if required) on data should be done before beginning the quality assessment. On the whole, the Luzzu web interface ensures that users (both publishers and consumers) can easily assess and analyse data quality from a single visual entry point.

4.4.6 Limitations of the Framework

In this section we analyse the technical (marked as **T**), functional (marked as **Fn**), and feature (marked as **Ft**) limitations of the framework from different aspects vis-a-vis the described methodology and

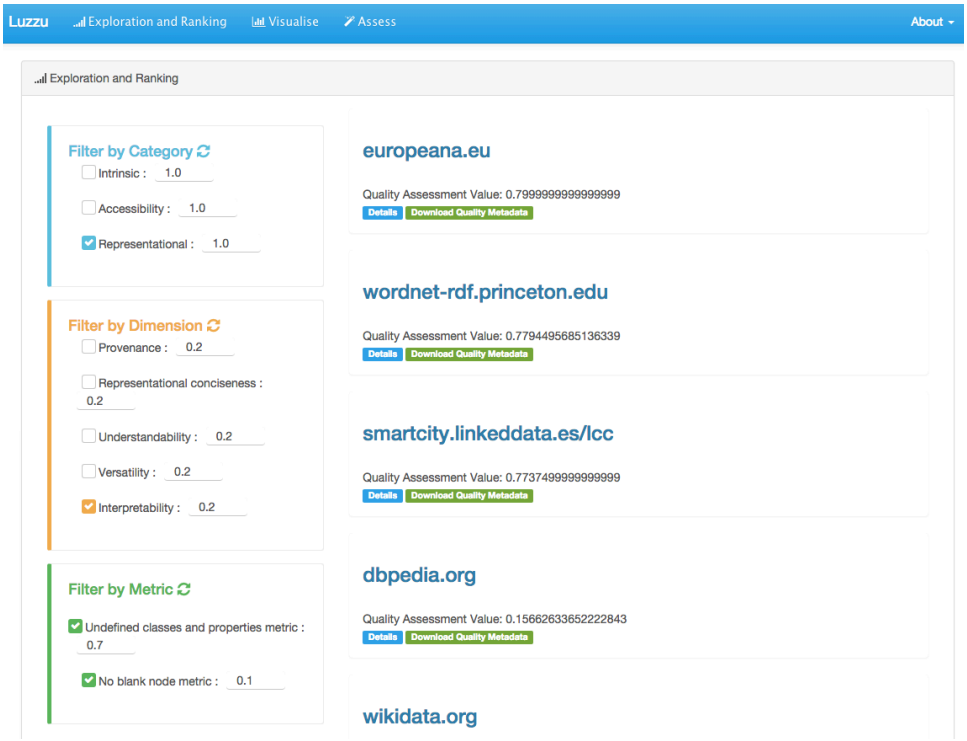

Figure 4.6: Faceted ranking of datasets.

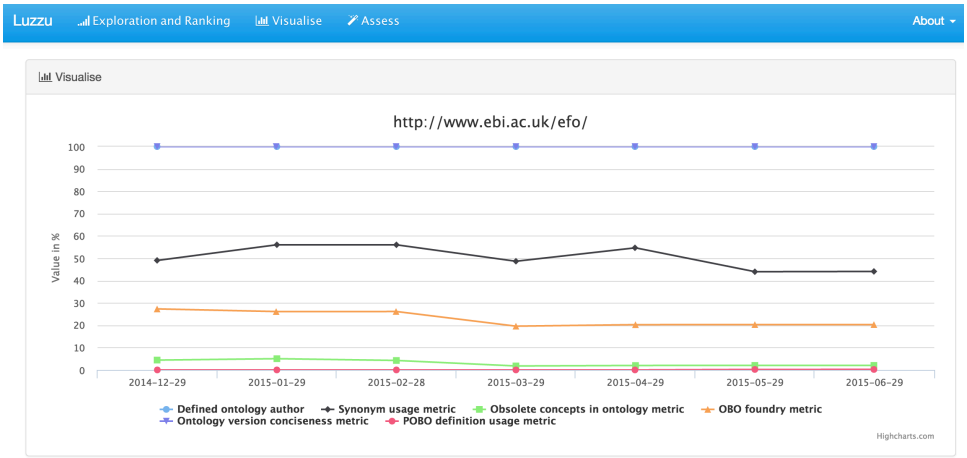

Figure 4.7: Visualising a dataset over time.

formalisation. Technical limitations are due to external libraries or the development environment our framework is based on, functional limitations arise from our methodology or formalisation, whilst feature limitations refer to features we would like to have but that are difficult to achieve in the current state of the framework or research.

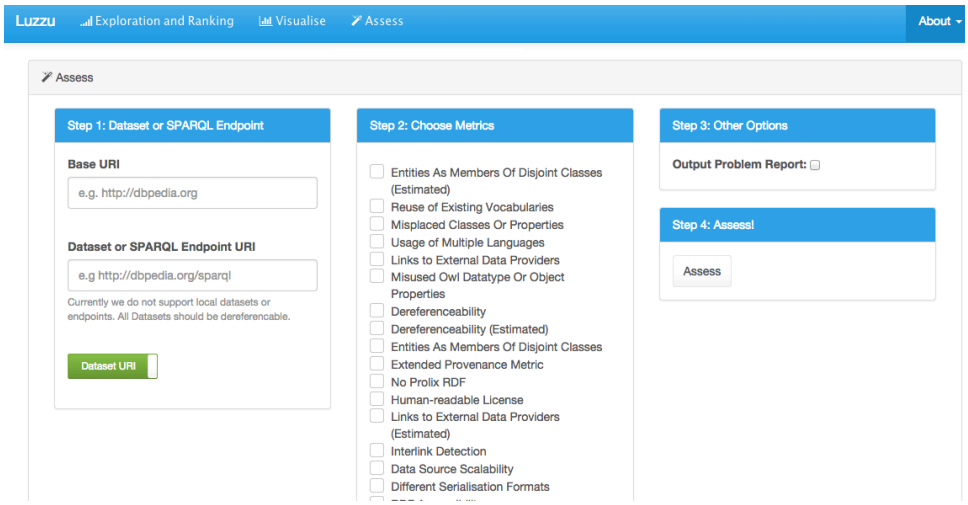

Figure 4.8: The Web Interface showing the Assessment Process.

Luzzu

1. (**Fn/Ft**) Regardless of the metric's computation, datasets are fully parsed till the very last triple. Therefore, if a particular metric just checks if a triple with a particular predicate exists in a dataset, the dataset is parsed till the end regardless of whether the required object was found before the end. This makes some metrics inefficient time-wise, but since the dataset processors have a linear performance, this limitation does not affect the user's overall productivity.

2. (**Ft**) Luzzu is not capable of automatically identifying the right quality metrics required during assessment for a particular task at hand. Therefore, as yet, the user has to manually choose the quality metrics she needs to assess a dataset on, and the right weights for ranking.

3. (**T**) Since no quality metric is defined in the SPARQL language, knowledge bases available through a SPARQL endpoint have to have triples streamed to Luzzu. In contrast to tools such as RDFUnit [99] where a quality metric is executed directly as a query to the endpoint, the Luzzu framework performs a series of SPARQL `SELECT` queries with `LIMIT` and `OFFSET` to retrieve the data from the triple store. The larger the triple store, the more expensive these queries become to perform on an endpoint, and thus the quality assessment could fail at any time (before or during the assessment itself) due to common problems such as timeouts. Furthermore, various endpoints have different settings, for example (i) (lack of) support of scrollable cursors required for the query to stream triples; or (ii) different timeout settings (500 Server Error).

4. (**Fn/Ft**) Although we employ a SPARK processor, the current state of Luzzu does not allow associative (one of the main requirements to implement a SPARK job) quality metrics that can be distributed amongst different clusters. However, if this is allowed, one has to take into consideration that in a quality assessment one might have a mixture of quality metrics distributed amongst clusters, and others that cannot be done in such a way. For example, a simple metric checking for a triple with a particular predicate can be designed in an associative manner. On the other hand, a more complex metric, such as the Extensional Conciseness metric [49], can be more difficult to achieve since the different clusters would need to know what triples have been processed previously.

Ranking Function

1. (**Fn**) The ranking approach we implement is based on the definitions in Section 4.3.8, i.e a weighted sum, which does not take into consideration profiled features (such as number of triples assessed) of the dataset being assessed. These dataset features can be obtained from VoID [7] descriptions or quality metadata if the framework embeds such profile features with metrics' observations.

2. (**Ft**) Taking into consideration profiled features of the dataset being assessed would enable more complex ranking. For example, if there is a tie between two datasets, there is no tie-breaker and these two datasets will be ranked randomly. Ideally, our ranking algorithm would take into consideration characteristics such as the number of triples in a dataset, in order to influence the final ranking in such cases. Currently, the Luzzu framework does not cater to store such statistical information, but these can be attached to the quality information as *Provenance* statements. For example, metrics record the number of assessed triples and attach this information together with the metrics' observation resource. This would require the revision of the Ranking definitions in Section 4.3.8.

4.5 Performance Evaluation

The aim of this experiment is to assess the scalability of the Luzzu framework. Runtime is measured for the Stream, SPARK and SPARQL processors against a number of datasets ranging from 10K to 125M triples. Since the main goal of the experiment is to measure the processors' performance, having datasets with different quality problems is considered irrelevant at this stage. Furthermore, this evaluation avoids external time-affecting factors, in order to achieve values solely related to the performance of the implemented processors. For example, the *Dereferenceability* metric [49] is a metric that requires consistent online access. This would deteriorate the overall computation time due to possible network latencies.

We generated synthetic datasets of different sizes using the Berlin SPARQL Benchmark (BSBM) V3.1 data generator[9]. We generated datasets with a scale factor of 24, 56, 128, 199, 256, 666, 1369, 2089, 2785, 28453, 70812, 284826, 357431, which translates into approximately 10K, 25K, 50K, 75K, 100K, 250K, 500K, 750K, 1M, 10M, 25M, 50M, 100M, and 125M triples respectively. Since these generated datasets will have no relevant quality problems, it was not necessary to initialise any metric processors and thus use them during performance evaluation. The processing capability will not be affected by the introduction of quality metrics in the framework, though once these metrics are introduced, the overall computation time will increase according to the time complexity of the initialised metric processors. Furthermore, the functionality of the dataset processors is to stream data from datasets (Stream and SPARK processors) or endpoint (SPARQL processor) to metrics. The stream and SPARQL processor tests were performed on a Unix based machine with an Intel Core i5 2.4GHz and 4GB of RAM, whilst for the SPARK processor three worker clusters were set up.

How quality metrics do not affect the processors' time complexity

In terms of software engineering, Luzzu employs a *low degree of coupling* between the modules. The framework's modules pass data to each other but do not care about the inner workings of other

[9] BSBM is primarily used as a benchmark to measure the performance of SPARQL queries against large datasets; cf. http://wifo5-03.informatik.uni-mannheim.de/bizer/berlinsparqlbenchmark/spec/. Date Accessed 3rd October 2016

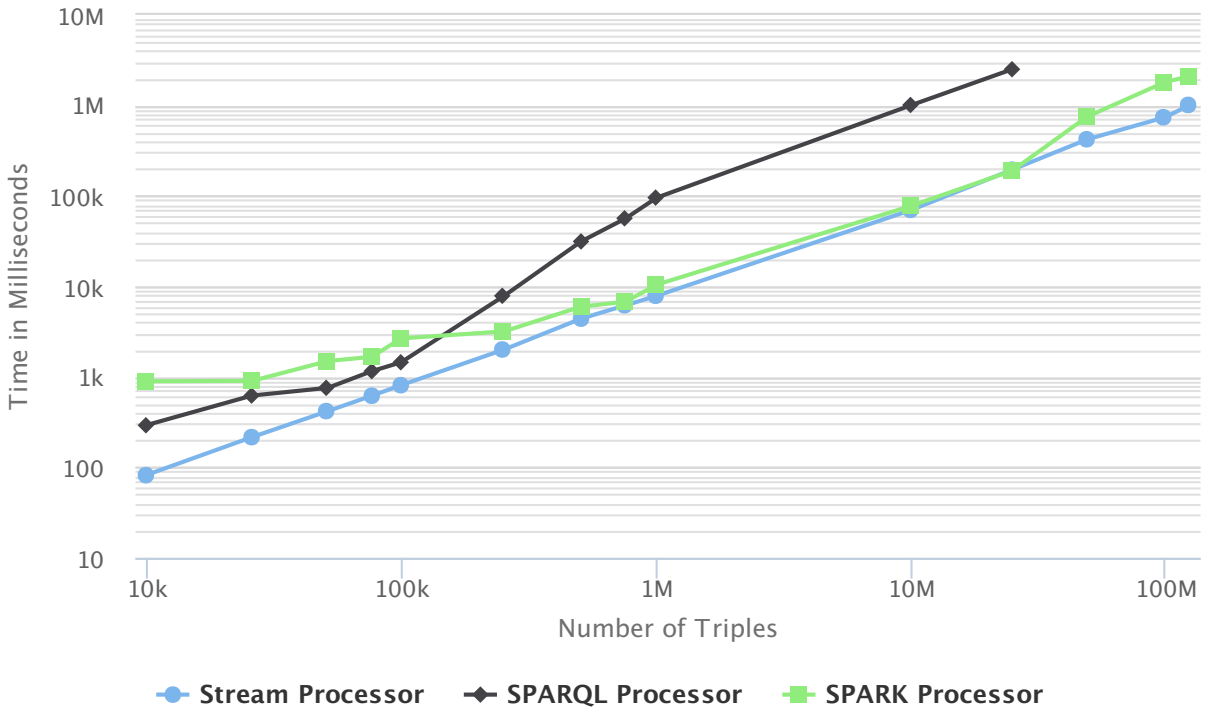

Figure 4.9: Time vs. Dataset Size in triples – Comparing Stream, SPARQL and SPARK dataset processors.

modules. This ensures that third parties can define quality metrics and add them to the framework as needed. The dataset processors do not know what the processors for these metrics do, but the important aspect is that the dataset processors know that these metric processors *exist* and that data (i.e. triples) should be passed through them for assessment. Therefore, any initialised quality metric will not affect the dataset processor's time complexity, as the dataset processors just read the triples and pass them to the metric processors, whereas the main method implemented by every metric processor (`QualityMetric.compute(Quad)`, corresponding to Definition 4.2 in Section 4.3.2) merely receives triples/quads without caring where they have been obtained from (e.g. a file stream, a resilient distributed dataset (RDD), a SPARQL endpoint or an in-memory dataset). Therefore, the performance of the dataset processors themselves will be the same for any number of simple or complex metric processors. Nonetheless, quality metrics will affect the *overall quality assessment* running time. We show such results in Chapter 7, where we assess the performance of a number of probabilistic-based quality metrics implemented for Luzzu. However, this is out of the scope of this performance evaluation of the Luzzu framework.

Results Figure 4.9 shows the time taken (in ms) to process datasets of different sizes. We normalised the values with a log (base 10) function on both axes to improve readability. All dataset processors scale linearly as the number of triples grows. The SPARQL processor was not responding in acceptable time for datasets larger than 25M triples. The results also confirm the assumption that Big Data technologies such as Spark are not beneficial for smaller datasets, whilst processing data from a SPARQL endpoint takes more time than the other two processing approaches.

From these results, we also conclude that for up to 125M triples, the stream processor performs better than the SPARK processor. A cause for this difference is that the SPARK processor has to deal with the extra overhead needed to enqueue and dequeue triples on an external queue, however, as the number of

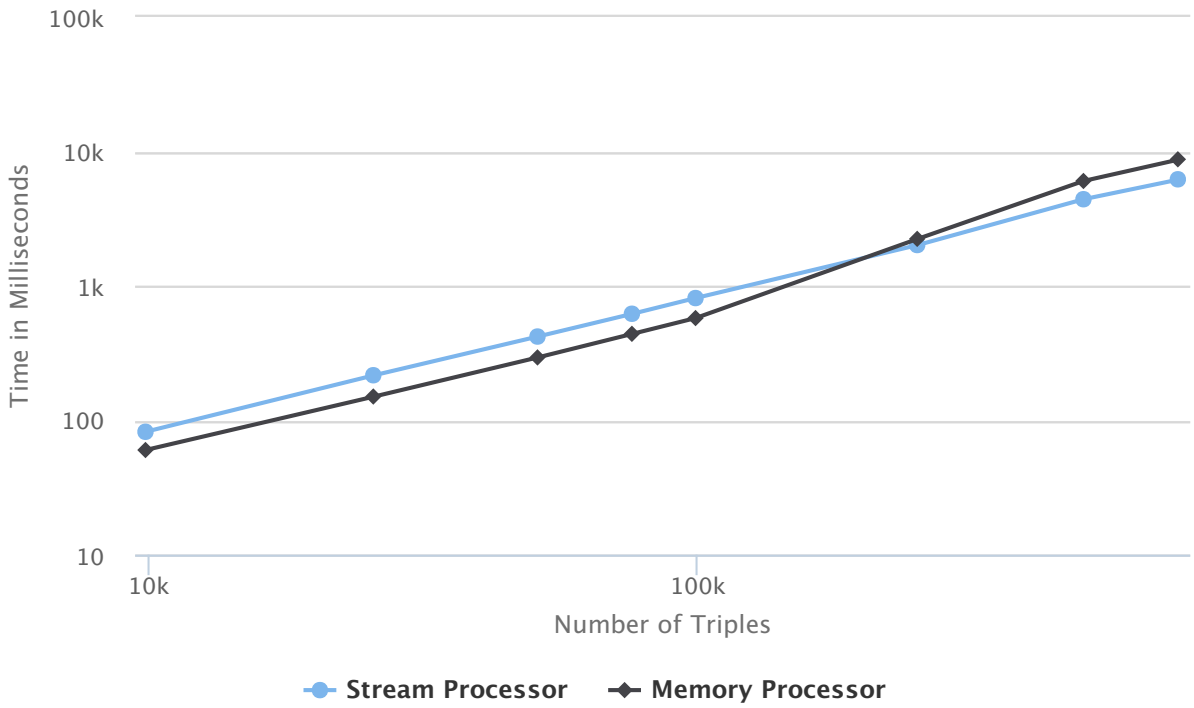

Figure 4.10: Time vs. Dataset Size in triples – Comparing In-Memory processor against Stream processor.

triples increases, the performance of both processors converges. With an increasing number of metrics to be computed, the execution time of both processors increases but remains linear with regard to the size of the dataset.

Following this experiment, we implemented an in-memory processor, which, rather than streaming triples directly from a dataset, loads the dataset under assessment in the available memory[10]. In Figure 4.10 we compare the two processors and see that, although faster, the in-memory processor does not significantly improve performance. When the dataset was larger than 250K triples, the stream processor fared better. One reason why this occurred is that the in-memory processor has the extra overhead of loading the dataset into memory. However, the main drawback of the in-memory processor is that the memory space has to be large enough to fit the dataset.

4.6 Concluding Remarks

Assessing the quality of linked datasets is of crucial importance for the Web of Data and its users – both data producers and consumers. Having high quality datasets and even more importantly being aware of the quality indicators ensures re-usability and thus helps to decrease the number of duplicate and redundant resources on the Web. Moreover, our refined data quality life cycle impacts the Web of Data positively with regard to data quality assessment, by providing a set of steps that enables the co-evolution of datasets. From a research perspective, Luzzu provides a framework for which data quality researchers can implement different metrics without the need of developing their own framework for processing.

Based on a conceptual methodology for assessing Linked Data quality, we have developed Luzzu, an *extensible*, *scalable*, *interoperable* and *customisable* framework as a means to assess data quality

[10] In our tests, the processor ran out of memory with 1M triples.

in Linked Datasets. This framework encompasses four major components: (1) an interface for adding custom quality metrics in Luzzu, using either traditional Java classes or the LQML domain specific language; (2) lightweight vocabularies to represent quality metadata and quality problems; (3) a big-data ready dataset and SPARQL endpoint processor; and (4) a user-driven quality-based weighted ranking algorithm.

The main contribution of the extensible custom metric definition component (first component) is that with Luzzu we ensure that datasets in various domains with different schemas can be assessed according to domain-specific quality measures. Furthermore, with the introduction of LQML (cf. Chapter 5), we aim to simplify Linked Data quality assessment in the long run. The vocabularies (second component) developed for Luzzu are novel with regard to data quality on the Web of Data, whilst enhancing interoperability across different quality frameworks. While the quality report vocabulary bridges the gap between data quality assessment and cleaning/repairing tools, the lightweight Data Quality Vocabulary (daQ - cf. Chapter 6) is adopted by the W3C Data on the Web Best Practices WG as a core module of their Data Quality Vocabulary [5]. The streaming approach (third component) of the Luzzu dataset processors ensures that large datasets can be fully assessed in a linear fashion, whilst the introduction of the Spark processor enables the realisation of the full potential of Big Data infrastructures. Finally, our contribution for the last component is mainly targeted towards data consumers, who can now filter and rank quality assessed datasets to find a 'fit' one, based on their specific quality priorities. In order to make Luzzu available immediately, we implemented a number of quality metrics (cf. Chapter 9) that can be used to assess linked datasets.

Overall, the key contribution of this chapter stems from the integration of these four components to create a **holistic** framework that enables the whole process of quality assessment in Linked Data with (1) the aim of helping data consumers to separate the wheat from the chaff to find *fit for use* datasets; and (2) the possibility to include the framework in a stack of tools (e.g. the LOD2 stack [12]) for co-evolution and curation of linked datasets.

With our approach we tried to fill the gaps of the other known state-of-the-art, in order to build the foundations to answer the research questions of this thesis. More specifically, with Luzzu we provide an answer for RQ 1 (cf. Section 1.3), whilst partially answering the *scalability* issue in RQ 3 from a framework's perspective. Furthermore, our framework indirectly tackles the first challenge (cf. Section 1.1) with regard to the *detection of quality problems*, as it easily extended by quality metrics to detect quality problems, of which we define a number based on the survey presented in [160]. Our approach allows stakeholders to *manually* choose the right quality indicators (cf. Challenge 2 in Section 1.1) required for their use case, to assess a datasets or to rank a number of dataset based on some quality indicators.

CHAPTER 5

Luzzu Quality Metric Language – A Domain Specific Language for Linked Data Quality Assessment

Quality assessment requires a lot of effort and consideration before processing a dataset. Although domain experts can decide on a number of quality factors for a particular dataset, in the end it is up to data consumers to see if a dataset is suitable for their use case or not. Furthermore, in order to have a more comprehensive view of a dataset's quality, domain specific quality metrics might be required. For example, for a geographical dataset it is of utmost importance that each resource has a geographic latitude, longitude and altitude.

In Chapter 4 we described a generic quality assessment framework that can be extended by third-party quality metrics. It often occurs that *quality assessors*, whose spectrum ranges from data publishers and consumers to domain experts and knowledge engineers, might not be confident in programming using traditional third generation languages. Nevertheless, they are considered to be the ideal drivers for defining domain specific quality metrics, which can be used on linked open datasets. The main challenge is to create a language that acts as a leveller between the conceptual methodology defined in Section 4.3 and the quality assessor, ensuring that anyone can easily define domain specific quality metrics.

The main contribution of chapter is the definition and implementation of the *Luzzu Quality Metric Language* (LQML - cf. Section 5.1), a domain specific language (DSL) that enables declarative definition of quality metrics for Luzzu. LQML offers notations, abstractions and expressive power, focusing on a the representation of quality metrics for linked dataset assessment. A particular challenge in the definition of LQML was to balance between providing the expressive power for defining sophisticated quality metrics on the one hand and ensuring their efficient processing and execution on the other hand. LQML is designed in a way that quality metrics can be written by non-programmers that are experts in the domain (cf. [52, 85, 115]). Hudak [85] suggests that DSLs have the potential to improve productivity in the long run.

The usability of the LQML is systematically assessed (cf. Section 5.3) against the "cognitive dimensions of notation" (CD) evaluation framework. These dimensions provide a comprehensive view of how users can manage and use a defined language. We also briefly outline the state-of-the-art in domain specific languages (cf. Section 3.2).

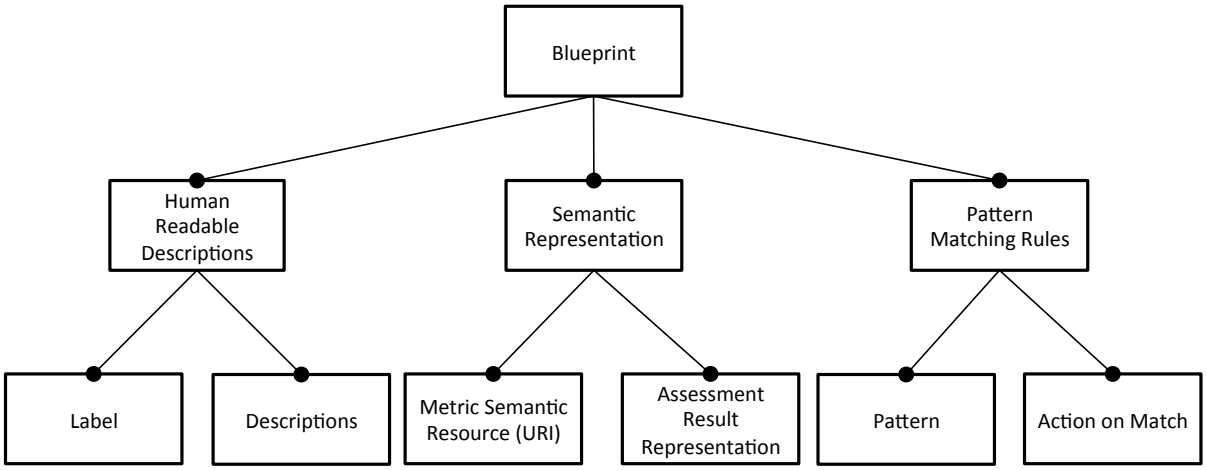

Figure 5.1: Feature Model for Blueprints.

5.1 Luzzu Quality Metric Language

The *Luzzu Quality Metric Language* (LQML) is a structural declarative language that enables the definition of quality metrics in Luzzu. Based on our experience in implementing quality metrics (for the large-scale evaluation described in Chapter 9), we anticipate that most domain-specific quality metrics are very similar structure-wise, with minor changes required only in the rules' conditions.

5.1.1 Analysis

Data quality assessment varies from one domain to another. Although there exist a number of generic quality metrics as defined in [160], different domains might require the assessment of different features. For example, where in geographical datasets the properties `geo:long` and `geo:lat` are absolutely required for resources that are defined as a place (such as country and city), these properties might be redundant in health oriented datasets. The idea of LQML is that quality assessors can define various quality metrics over a dataset (or a domain of datasets). These declarative definitions are translated into Java byte-code (see Section 5.1.3) and integrated within the Luzzu framework. For the proposed domain specific language, we identified a domain terminology based on a number of quality metrics identified in [160]. We need to keep in mind that new use cases require new metrics, so the language should be extensible by new functions to reflect this need.

Domain Terminology: A typical quality metric definition for linked open datasets consists of a *pattern matching condition*, (i.e. matching the subject (`?s`), predicate (`?p`), object (`?o`), or a mixture of these three with possibly advanced inspection), and an *consequent action*. This resembles the traditional *if... then* statements of programming languages. The full representation of an LQML metric definition is termed as *blueprint*. The feature model in Figure 5.1 describes the features required to create a quality metric blueprint. A blueprint description should have enough information to assess a dataset based on the quality criteria (*Pattern Matching Rules* in Figure 5.1), and to enable the semantic description (*Semantic Representation*) of the quality metadata for the criteria in question. A description should also have a *Human-Readable Description*. This is required since blueprints could be shared amongst different quality assessors and thus should enable anyone to understand complex patterns and actions. All features are necessary in order to create one blueprint.

5.1.2 Design

Having identified the features for the proposed domain specific language, we here concisely describe its design and its features. Mernik et al. [115] describe a number of design patterns, three based on *language exploitation* (designs based on existing languages), and another one for *language invention*. Our proposed DSL is based on the *language invention* design pattern, where we fuse a number of specific terms from known quality metrics together with variable binding expressions used in the syntax of SPARQL (i.e `?s ?p ?o`) to refer to specific elements in a triple.

Quality Metric (Blueprint) Structure: A blueprint definition of a metric starts with the `def` keyword and has a rule semantics. Each blueprint consists of the three features mentioned in Section 5.1.1.

```
def{Dereferenceability}:
```

Listing 5.1: An example of the `def` keyword.

Pattern Matching Rules Feature: Declarative patterns start with the keyword `rule`. Each rule has a `match` block (*Pattern* feature), the condition part c of one $c \mapsto a$ mapping of a rule, which if statisfied triggers the `action` block (*Action on Match* feature), the action part a of one mapping of a rule. Together, these represent the λ assessment rule in the QMP defined in Section 4.3.5.

Any input triple $< s, p, o >$ is matched against the conditions that follow the `match` keyword, enclosed into curly brackets ({ }). A `match` can have one or more conditions. Conditions can be connected via the *logical and* (`&`) operator or the *logical or* (`|`) operator. LQML has the following match functions defined:

- `typeOf(?s | ?o)` checks the type of the subject or the object (given that the object is a URI);
- `isURI(?s | ?o)` checks if the subject or object are valid URIs;
- `isBlank(?s | ?o)` checks if the subject or object are blank nodes;
- `isLiteral(?o, datatype)` (e.g. `isLiteral(?o, xsd:date)`) checks if the object is a literal value of the defined datatype.

Furthermore, a user can use the equality operator == to match objects:

- `?s | ?p == <U>` matches the subject (`?s`) or the predicate (`?p`) against a given IRI (`<U>`);
- `?o == <U> | x` matches the object (`?o`) against a given IRI (`<U>`) or a literal (`x`).

A satisfied `match` block can then trigger one or more of the following `actions`:

- `count` increments a counter;
- `countUnique(?s | ?p | ?o)` increments a counter only if a unique instance of `?s`, `?p`, or `?o` is encountered.

In order not to limit LQML expressiveness, programmers can develop custom functions, such as `custom.IsDereferencable` in Listing 5.2, that can be used in `match` and `action` blocks. Many domain specific languages, such as XPath, provide the functionality where users can implement external functions[1]. Furthermore, a *rule* can have multiple `match` $\mapsto$ `action` blocks. In this example, `var.x` and `var.y` holds the number of URIs (in the assessed dataset), and the number of dereferenceable URIs respectively.

[1] http://www.w3.org/TR/xpath-30/#id-function-calls. Date Accessed 3rd October 2016

```
rule{
  var.x = match{isURI(?o)} => action{ count(?o) }
  var.y = match{custom.IsDereferencable(?o)} => action{ count(?o) };
};
```

Listing 5.2: An example of the pattern matching rules feature.

Human Readable Descriptions: Descriptive human-readable comments are also required in these blueprints. We provide the keywords `label` and `description` to provide the metric's name and its textual description; they translate to `rdfs:label` and `rdfs:comment`.

```
label{"Dereferenceablity of Resources"};
description{"Measures the percentage of dereferenceable object URIs with respect to
    the total object URIs"};
```

Listing 5.3: An example of the human readable descriptions feature.

Semantic Representation: The definition also expects other information that describes a quality metric. The `metric` (*Metric Semantic Resource (URI)* feature) keyword expects a quality metric resource URI. These resources are defined in a vocabulary that extends the Dataset Quality Vocabulary (daQ - cf. Chapter 6). The action(s) in the `finally` (*Assessment Result Representation* feature) block is triggered after all triples in a dataset have been assessed with the rule block λ (cf. Section 4.3.5). Its output is used as a daQ observation value.

The `finally` keyword can have one of the following parameters:

- `add(x,y)` takes the output of two `rule` value variables and adds them together;
- `ratio(x,y)` takes two parameters, which can either be integer or float numbers, the output of a *rule* value or even a function (e.g. `totalTriples`) that returns a numeric value. The ratio function divides `x` by `y`;

```
metric{<http://example.org/quality/Dereferenceability>};
finally{ratio( add(var.x, var.y), var.z )};
```

Listing 5.4: An example of the semantic representation feature.

5.1.3 Implementation

The LQML grammar is implemented in JavaCC (Java Compiler Compiler)[2]. JavaCC is a parser generator and a lexical analyser, where the grammar is specified in EBNF notation. Blueprints defined in LQML are interpreted by the JavaCC compiler where each blueprint is then interpreted and transformed into a Java class during Luzzu's runtime.

The following listings (Listings 5.5 to 5.7) shows the EBNF grammar for the main parts of the LQML syntax. A blueprint starts with the `header` definition, which includes information about the creator of the blueprint metric and information about the final (Java) package of the generated metric (since the blueprint will be compiled into a Java class that will be used within Luzzu (Section 4.4) as a metric). Following the `header`, the `def` marks the start of the blueprint, followed by a permutation of the `metric`, `label`, `description`, `rule` blocks.

[2] https://java.net/projects/javacc. Date Accessed 3rd October 2016

The `rule` block is made up of one or more `match` and `action` pairs, whose result of each pair is stored in a variable that can be used later in the `finally` block. The match condition is made up of one or more *atomic* and *simple* conditions. A *simple* condition is one that assesses simple equality between two values, possibly returned by some function. Given a number of *simple* conditions, an *atomic* condition is one that returns *true* or *false*. An action is triggered if the condition is satisfied. The `finally` block uses the values of the variables assigned in the `rule` block.

```
<Blueprint> :=  <Header> <Def> <DefinitionPerm> <Finally> <Period>

<Header>  :=  <Header_Indicator> <Author> | <Package>

<Def> :=  "def" <LBrace> <Strict_Str> <RBrace> <Colon>

<DefinitionPerm>  :=  (
      (<Metric> <Label> <Description> <Rule> ) |
      (<Label> <Rule> <Description> <Metric> ) |
      (<Rule> <Metric> <Label> <Description>) |
          ... # 21 other permutation possibilities
    )

<Finally> :=  "finally" <LBrace>  (<FinallyActions>) <RBrace>

<FinallyActions>  :=  (<Add> | <Ratio>)

<Add> :=  "add" <LParen> (<TotalTriples> | <Vars> | <FinallyActions>) <Comma>  (<
    Vars> | <FinallyActions>) <RParen>

<Ratio> :=  "ratio" <LParen> (<TotalTriples> | <Vars> | <FinallyActions>) <Comma>
    (<Vars> | <FinallyActions>) <RParen>
```

Listing 5.5: LQML main Grammar.

```
<Author>  :=  "author" <Colon> <Quoted_Str>
<Package> :=  "package" <Colon> <Quoted_Str>
```

Listing 5.6: LQML header grammar.

External Functions: External functions, defined as Java classes, are preloaded into Luzzu. These can only be used within a `match` and `action` patterns. The structure (as described in the EBNF `<Custom>`) requires a function name (as a string) and one or more parameters. In Appendix C we show a typical implementation of a custom function.

```
<Metric>  :=  "metric" <LBrace> <IRIref> <RBrace> <SemiColon>

<Label> :=  "label" <LBrace> <Quoted_Str> <RBrace> <SemiColon>

<Description> :=  "description" <LBrace> <Quoted_Str> <RBrace> <SemiColon>

<Rule>  :=  "rule" <LBrace> (<Vars> "=" <Match> "=>" <Action>)+ <RBrace> <SemiColon
    >

<Vars>  := "var" <Period> <Strict_Str>

<Match> :=  "match" <LBrace> (<Condition>)+ <RBrace>

<Condition> :=  ((<LogicNegation> <AtomicCondition>) | <AtomicCondition> | <
    SimpleCondition>)
    (<LogicBinaryOp> <Condition>)*

<SimpleCondition> := <LParen> "?s" <BooleanOp> <IRIref> <RParen>
  | <LParen> "?p" <BooleanOp> <IRIref> <RParen>
  | <LParen> "?o" <BooleanOp> ( <IRIref> | <Quoted_Str> ) <RParen>

<AtomicCondition> := <TypeOf> | <IsURI> | <IsBlank> | <IsLiteral> | <Custom>

<Custom>    :=  "custom" <Period>  <Strict_Str> <Params>

<Params>    :=  <LParen> ("?s" | "?p" | "?o") (<Comma> ("?s" | "?p" | "?o"))* <
    RParen>

<Action>    :=  "action" <LBrace> <Count> | <CountUnique> | <Custom> <RBrace>
```

Listing 5.7: LQML definition constructs.

5.2 Complete Blueprint Examples

In this section, we provide the reader with a number of complete blueprints. Listing 5.8 show a simple LQML blueprint definition of a metric that calculates the number of blank nodes used in an assessed dataset. The blueprint's rule counts the number of subjects and objects that are blank, and once the assessment is done, the quality value is given as a ratio of blank nodes against the total number of triples assessed.

```
%% author: "Jeremy"
%% package: "org.test.metrics"

def{NoBlankNodeMetric}:
  metric{<http://purl.org/eis/vocab/dqm#NoBlankNodeMetric>};
  label{"Usage of blank nodes."};
  description{"Calculates the percentage of blank nodes used in a dataset."};
  rule{
    var.x = match{isBlank(?s)} => action{count(?s)};
    var.y = match{isBlank(?o)} => action{count(?o)};
  };
  finally{ratio(add(var.x, var.y),totalTriples)} .
```

Listing 5.8: Usage of Blank Nodes Metric in LQML

In Listings 5.9 and 5.10 we introduce three new custom functions, `IsDereferenceable`, `IsDeprecatedClass`, and `IsDeprecatedProperty`. In Listings 5.9, the `actions` are triggered

if the subject and the object are both URIs and dereferenceable (metric is discussed further in Section 7.1.1 and Section 9.3.5). On the other hand, Listing 5.10 `actions` are triggered when (i) a used class (triple with the predicate `rdf:type`) is member of `owl:DeprecatedClass`, and (ii) a used property (skipping `rdf:type`) is member of `owl:DeprecatedProperty`.

```
%% author: "Jeremy"
%% package: "org.test.metrics"

def{DereferenceabilityMetric}:
  metric{<http://purl.org/eis/vocab/dqm#Dereferenceability>};
  label{"Dereferenceability of Resources"};
  description{"Measures the percentage of valid dereferenceable resources in a
    dataset"};
  rule{
    var.x = match{(isURI(?s) & custom.IsDereferenceable(?s))} => action{countUnique
    (?s)};
    var.y = match{(isURI(?o) & custom.IsDereferenceable(?o))} => action{countUnique
    (?o)};

  };
  finally{ratio(add(var.x,var.y), totalTriples)}.
```

Listing 5.9: Dereferenceability of resources in LQML.

```
%% author: "Jeremy"
%% package: "org.test.metrics"

def{DeprecatedClassesAndPropertiesMetric}:
  metric{<http://purl.org/eis/vocab/dqm#UsageOfDeprecatedClassesOrProperties>};
  label{"Usage of Deprecated Classes and Properties"};
  description{"Measures the percentage of owl:DeprecatedClasses and owl:
    DeprecatedProperties used in a dataset"};
  rule{
    var.x = match{?p == <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> & custom.
    IsDeprecatedClass(?o)} => action{countUnique(?o)};
    var.y = match{?p == <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> & custom.
    IsDeprecatedProperty(?p)} => action{countUnique(?p)};
  };
  finally{ratio(add(var.x,var.y), totalTriples)}.
```

Listing 5.10: Usage of deprecated classes or properties in LQML.

5.3 Initial Assessment of the Luzzu Quality Metric Language

As an initial assessment of the LQML, we gauge the language systematically against the "cognitive dimensions of notation" (CD) evaluation framework, a methodology developed in [29]. This evaluation framework has previously been applied to Semantic Web languages (e.g. [130]). These cognitive dimensions provide a comprehensive view of how users can manage and use a defined language, and help us identify possible future improvements. Each dimension describes a specific aspect in relation to the language notation. Blackwell and Green [28] describe the following dimensions:

1. **Viscosity** questions the effort required by the user to lead out a change.
   Assessment: LQML metrics can be defined using a simple text editor. Each statement is defined for a particular definition and is not related to other definitions. Therefore, changing a statement

in a definition does not require a change in any other place, thus resulting in a low viscosity. If modifications are required in custom functions, changes can be done directly to the function without affecting the syntax of the LQML definitions. On the other hand, we cannot commit ourselves to the validity of the semantics of the definitions after such changes.

2. **Premature Commitment** measures any planning required before leading out a task.
   Assessment: Based on declarative programming, LQML users only need to define rules based on the patterns they want to match. The only premature commitment is that metrics have to be defined in an ontology based on the daQ vocabulary.

3. **Hidden Dependencies** measures if dependencies are specifically indicated in all existing directions.
   Assessment: LQML definitions cannot be connected to each other, therefore each definition has a fixed rule block, together with other descriptions.

4. **Error-proneness** measures the possibility of users making mistakes while using the language.
   Assessment: A definition is made up of a small set of defined constructs plus the extra defined custom functions. This means the learning curve is not too steep. With regard to custom functions, the user has to know exactly the class name and the exact parameters that are required by the function. In any case, any syntax errors are detected by the JavaCC compiler and are displayed to the user.

5. **Abstraction** measures high level concepts which are not easily grasped by the users, since they do not refer to concrete instances. This dimension thus measures the language's abstraction level.
   Assessment: In LQML, we try to keep the number of keywords to a minimum, such that users can have full control of their declarative patterns. In this way there is a very low level of abstraction.

6. **Secondary Notation** indicates the availability of options for encoding extra context information within the syntax itself, such as comments.
   Assessment: A definition requires a human-readable description; further important information can be added in an unstructured way as comments (starting with #, extending to the end of the line).

7. **Closeness of Mapping** measures the degree of similarity between the representation language and the real-world domain.
   Assessment: Our aim is to try to simplify the definition of metrics as much as possible, keeping in mind that possible non-Java experts are using this tool. Despite having this beneficial feature that widens the tool's audience, expert users who require to create more complex metrics, for example, calculating the response time of a server serving a resource, must implement LQML custom functions in Java.

8. **Consistency** measures the usability of the language; in other words, how easy is it for a user to write similar LQML definitions once the notation pattern has been learned.
   Assessment: Similar to error-proneness, the small set of defined constructs enables user to write definitions quickly. On the other hand, the ability of LQML users to define more than just simple quality metrics relies on a sufficient pool of custom extension functions.

9. **Progressive Evaluation** measures the understandability of the language even for a solution that is incomplete. The possibility to try out a partial solution helps users in further understanding their work.
   Assessment: It is possible to incrementally refine definitions by, e.g., starting with a partial match

and a simple "count" action, and then to further refine the matching pattern by adding conditions, thus defining a more complex action.

10. **Role Expressiveness** indicates the language's notation and its expressiveness vis-a-vis the whole solution.
Assessment: Our tool is aimed towards the definition of quality metrics for linked data. In a definition, all required information is adequately labelled to enable easy identification.

11. **Visibility** measures the degree of visibility of the language's notation. If concepts are encapsulated into concepts of a more abstract level, this reduces the visibility of the notation.
Assessment: All available notation is directly visible to the user.

12. **Provisionality** measures the ability of the language to allow users to explore potential options.
Assessment: Similarly to the secondary notation and progressive evaluation dimensions, potential options can be explored by temporarily commenting out parts of a definition.

Together, the assessment with regard to these dimensions provides a comprehensive heuristic guide of LQML, particularly focusing on language features that have not been implemented in an immediate response to the given quality assessment requirements. From this evaluation we can identify certain problems in the current implementation of the syntax. These heuristics stress the importance of the need of a better visual presentation tool (graphical interface) for the user, while also highlighting that whilst we are widening the scope of metric definition for non-Java experts, the main shortcoming of LQML is that the defined language for the conditions and actions cannot be elaborated for more complex tasks. These heuristics also pointed out two limitations of our initial approach:

1. (**T**[3]) Expressivity of the core language is rather shallow in contrast to how the language maps to the real-world domain. The current state of the language relies on external functions to be implemented by third parties. Moreover, metrics involving states (i.e. State-aware Inductive AQMs) cannot be expressed by the core language, though these can be achieved by its extensibility feature, i.e. developing custom functions for *match* and *action* constructs.

2. (**Ft**[4]) Complex QMPs (defined in Section 4.3.5) cannot be expressed in the current state of LQML. This would require additional definition constructs defined in the language's BNF without adding too much effort on the user.

These measurements will help us in the next phase of the language definition, before conducting the usability study with interested parties.

5.4 Concluding Remarks

With the Luzzu Quality Metric Language we empower domain experts who are not proficient in using third generation programming languages to define domain specific quality metrics for linked open datasets. Using terminology extracted from the various quality metrics classified in [160], we developed a DSL that quality assessors without prior knowledge of traditional object oriented languages can define their own simple quality metrics in Luzzu. Furthermore, the language's flexibility of allowing custom functions widens the spectrum the quality metrics that can be defined. Following Chapter 4, this chapter

[3] Technical Limitation
[4] Feature Limitation

adds to the answer for RQ 1 (cf. Section 1.3), where stakeholders can define their own (possibly domain specific) quality indicators and assess their dataset in Luzzu using these defined indicators. With regard to the first challenge (cf. Section 1.1), LQML enables the definition of quality metrics (hence together with Luzzu, LQML metric definitions enable the *detection of quality problems*).

With this language, we provide the first step towards having a Linked Data quality assessment framework that can be used by non-Linked Data users. In WIQA [26], the authors use a SPARQL-like language to enable users to apply different filtering policies on datasets, to include the definition of domain-specific assessment metrics. RDFUnit [99] requires some knowledge of SPARQL in order to define quality metrics, whilst in Sieve [114], XML patterns are used to defined metrics. Furthermore, LiQuate [137] requires the knowledge of Bayesian rules, and LinkQA [61] allows the creation of quality metrics using Java.

LQML was evaluated systematically against the Cognitive Dimensions of Notation, a methodology developed purposely to assess formal notations such as those of programming languages. The evaluation pointed out shortcomings in the current implementation of the DSL, more specifically the lack of a construct for defining complex QMPs. Moreover, the DSL still has to be evaluated and tested by external quality assessors to gauge the language's benefits and barriers in more detail.

Part III

Semantification of Quality Metadata

In Part II we described an infrastructure, Luzzu, to assess linked datasets. However, in order to make the assessment results available for public consumption as Linked Data, the Luzzu framework has to support a schema that represent such results. In Chapter 6 we describe the Dataset Quality Vocabulary (in short daQ), a core vocabulary for representing the results of quality benchmarking of a linked dataset. It represents quality metadata as multi-dimensional statistical observations. Quality metadata are organised as a self-contained graph, which can then be embedded into linked datasets to support quality-based retrieval and ranking. We discuss the design considerations, and current and potential uses of daQ. daQ is used in a number of initiatives and applications, including the broader W3C Data Quality Vocabulary. The contributions from this part answers research question 2 defined in Section 1.3.

Chapter 6 is an updated version of the following publications:

- **Jeremy Debattista**, Christoph Lange, Sören Auer. *Representing dataset quality metadata using multi-dimensional views.* In Proceedings of the 10th International Conference on Semantic Systems (SEM '14), 92-99, ACM;
- **Jeremy Debattista**, Christoph Lange, Sören Auer. *daQ, an Ontology for Dataset Quality Information.* In Linked Data on the Web (LDOW) 2014 at WWW'14, CEUR-WS.org vol. 1184, 2014.

CHAPTER 6

Dataset Quality Vocabulary (daQ) - Semantically Representing the Quality of Linked Datasets

A substantial amount of Linked Datasets have already been published on the Web. This heterogeneity implies a great variance in quality, thus causing problems such as inconsistencies and incompleteness, and consequently potentially rendering datasets to be not fit for a certain task. Furthermore, data sources are bound to evolve during their life-span, leading to an increase or decrease in quality. Whilst various metadata might be available for different sources, quality information is rarely available and data consumers more often than not rely on measures such as trust or popularity when choosing a dataset for their use case. However, this does not necessarily mean that consumers are choosing the dataset that is most fit for use.

The Data on the Web Best Practice affirm the importance of having quality information available to the consumer, stating that:

> *"Data quality might seriously affect the suitability of data for specific applications, including applications very different from the purpose for which it was originally generated. Documenting data quality significantly eases the process of dataset selection, increasing the chances of re-use. Independently from domain-specific peculiarities, the quality of data should be documented and known quality issues should be explicitly stated in metadata."* – [105, §8.5]

In this chapter we tackle Research Question 2 defined in Section 1.3. We look into different ontology engineering techniques and other related vocabularies (cf. Section 3.3) to define a schema that enables the representation of dataset quality. Hence, our aim is to enable the definition and use of data quality metrics in a standardised manner, and to represent the quality of data over time as concrete values.

Following these requirements, we introduce the *Dataset Quality Vocabulary (daQ)* (cf. Section 6.2), which is the main contribution of this chapter. daQ is a light-weight core vocabulary for representing the results of quality benchmarking of a linked dataset again as linked data. This allows the embedding of quality metadata into datasets, thus "stamping" them with a number of quality measures and exposing this information in a re-usable way. In this thesis, daQ metadata is an output of the Luzzu framework (described in Chapter 4) following an assessment in order to describe the quality of a number of linked datasets (such as the datasets used for the evaluation in Chapter 9), and is subsequently used for the ranking functionality described in Section 4.3.8.

This interoperable metadata can then be used by data consumers in semantic-based tools for querying, analysis, and visualisation, amongst others. We discuss a number of use cases (cf. Section 6.1) to illustrates the potential use of quality metadata. We also discuss how potential users can extend and use daQ in their applications (cf. Section 6.3). In this chapter, we also describe a tool used to validate daQ-based quality models (cf. Section 6.5), and how daQ was adopted in a W3C vocabulary recommendation (cf. Section 6.4).

In this thesis, we often refer to daQ as the **meta-model** or the **vocabulary**. The rationale is to disambiguate the abstract daQ model from concrete quality measure **schemas**[1].

6.1 Use Cases

Linked Open Data quality has different stakeholders in a myriad of domains, but in any case the stakeholders can be cast under either *publishers* or *consumers*.

Publishers are mainly interested in publishing data that others can re-use. The five star scheme for deploying open data[2], defines a set of widely accepted criteria that serve as a baseline for assessing data re-usability. The re-usability criteria defined by the five star scheme and by quality metrics are largely measurable in an objective way. Thanks to such objective criteria, one can assess the re-usability of any given dataset without the major effort of, for example, running a custom survey[3] to find out whether its intended target audience finds it useful. In [87], Hyvönen et al. propose two additional stars to the 5 star criteria, of which the seventh star describes the importance of explicitly stating the quality against a particular schema. On this basis, we can modify the criteria of this quality star such that the benefits for a data publisher includes: (i) that published data conforms to the established domain quality metrics; (ii) catalogued and archived datasets can be easily discovered when consumers filter by quality aspects; and (iii) encourage re-usability.

For a data *consumer* (both machine and human), the benefits of a quality star is that datasets can be identified on whether quality assessment has been performed or not, and thus can be filtered via different quality measures. Nevertheless, without an objective rating that is easy to determine, consumers may find it challenging to identify the quality of a dataset, i.e. its fitness for use.

Prior to the introduction of standardised quality vocabularies (in particular daQ and DQV), publishers relied on their mutual trust they have with data providers, believing that the data they provide is good for data consumers, leaving the data publishers in the dark of the value and quality of their data. Whilst this might still be the case, together with quality assessment tools, data publishers can now assess the quality and provide quality metadata in a semantic and interoperable format, thereby data consumers can make informed decisions on what data is fit for their use case.

The following use cases (UC) show how both data publishers and consumers can benefit from having quality metadata about datasets. The use cases thus motivate the need for developing a standard representation like daQ.

6.1.1 UC1: Analysis of Data Versions

Ideally, data publishers update their published datasets regularly to (a) keep the data fresh and up-to-date; (b) clean data to improve quality; (c) keep up with the data curation life cycle. However, it is sometimes difficult to identify which aspects of the data are lacking quality standards. Furthermore,

[1] The Data Quality Metric (DQM) http://purl.org/eis/vocab/dqm is one such example
[2] http://5stardata.info/. Date Accessed 3rd October 2016
[3] Such a survey may, of course, still help to get an *even better* understanding of quality issues.

it is even more difficult to analyse how data quality changed over time. Assuming that there are tools that automatically analyse data quality and output daQ metadata, the daQ structure enables a multi-dimensional representation, where the versions of a dataset form one dimension, and the different quality metrics form the other dimension.

6.1.2 UC2: Cataloguing and Archiving of Datasets

Software such as the *CKAN* data portal engine, which is driving the Open Knowledge Foundation data management portal (datahub.io) and many national open data portals, makes datasets accessible to consumers by providing a variety of publishing and management tools as well as search facilities. A data publisher should be able to upload to such platforms, whilst on the other hand the platform should be able to automatically compute quality metadata (if not already available) regarding the dataset's quality. With this metadata, the platform should be able to catalogue datasets according to its available quality assessed attributes, which would then be used for ranking and retrieving datasets. The structure of the daQ meta-model enables the archiving of previously assessed data quality values. This facilitates the job of a data consumer of discovering potential datasets based on (i) its quality aspects; and (ii) the publishers' willingness to update a dataset to improve its quality.

6.1.3 UC3: Retrieval of *Fitness for Use* Datasets

Alexander et al. [6] provide the readers with a motivational use case with regard to how the VoID ontology can help with effective data selection. The authors describe how a consumer can find the appropriate dataset by:

- criteria related to content (what is the dataset mainly about);
- its links from and to other datasets;
- the vocabularies used in the dataset.

The daQ vocabulary gives the *fit for use* factor to *appropriateness* by providing the consumer with quality criteria on the candidate datasets.

An objective assessment of data quality enables data consumers to determine if a dataset is fit for a certain use case. Currently, tools targeting human data consumers, such as semantic web search engines [81] or Data Web browsers [24, 67, 75], do not focus on dataset quality when presenting search results. With the introduction of the daQ framework, tools that provide faceted browsing facilities, such as CKAN, are enabled to provide more information about a dataset's quality attributes. Such functionality is attributable to the flexibility of the ontology, enabling various filtering and ranking possibilities of the dataset quality metrics. This would permit human data consumers to better understand the quality attributes of a dataset, and thus choose which dataset is the most fitting to their use case. The daQ model enables data consumers to track and follow quality improvements of data publishers on their datasets over time. This also opens a sundry of opportunities leading to the assessment of data publishers regarding their willingness to enhance the value of the data in terms of quality. To keep the quality metadata of open datasets easily accessible, we recommend that each dataset contains the relevant daQ metadata as a distinct named graph within the dataset itself.

6.1.4 UC4: Link Identification

Identifying links between existing datasets is one of the main drivers that makes the Linked Open Data Cloud more coherent. Tools such as *LIMES* [121] or *Silk* [152] support the automatic identification of

links according to built-in as well as user-defined criteria. The introduction of quality metadata to datasets will add another criterion for link identification, in that linking algorithms can also take the quality of the target dataset into consideration before linking to it. Linking tools could also consider the needs of a data consumer who might not only require to link to any high quality entity, but possibly even to those datasets which the consumer deems "fit" to her cause. This can be done by filtering candidate datasets according to criteria such as weights on specific quality metrics defined by the consumer. Linking resources of proven quality helps to improve the quality of both datasets participating in the linkage.

6.2 The Dataset Quality Vocabulary (daQ)

Due to the subjective nature of *quality*, defining a generic dataset quality vocabulary is difficult. This is attributable to the fact that *fitness for use* means that different domains and use cases requires different quality measures to be assessed on the data. For instance, the W3C Data on the Web Best Practices working group defines different use cases [101] with different quality requirements.

The *Dataset Quality Vocabulary*[4] (prefix: `daq`), illustrated in Figure 6.1, is an abstract metadata model that provides a pragmatic solution for easily extending the vocabulary with custom data quality concepts that will be interoperable with other daQ measures. Therefore, any linked dataset can have a quality metadata graph attached to it, with assessment results computed by different parties. daQ is based on two W3C standard vocabularies, the RDF Data Cube Vocabulary [40] and the Provenance Ontology (PROV-O) [100], and is a direct implementation of the quality indicators classification defined in the data quality survey in [160]. To keep the vocabulary lightweight, mainly RDF Schema was used, with a few OWL constructs. In terms of the ontology design patterns (ODP), the daQ vocabulary follows the *Architectural Ontology Pattern*[5]:

> *Architectural ODPs affect the overall shape of the ontology: their aim is to constrain how the ontology should look like.*

In our case the constraints refer to the defined concrete quality measures based on daQ.

6.2.1 The Quality Graph

Quality metadata is intended to be represented as a `daq:QualityGraph` (Figure 6.1 – Box A) within the linked dataset itself. The semantics of a quality graph is defined in Listing 6.1[6].

```
daq:QualityGraph
  a rdfs:Class, owl:Class ;
  rdfs:subClassOf rdfg:Graph , qb:DataSet ,
    [
      rdf:type owl:Restriction ;
      owl:onProperty qb:structure ;
      owl:hasValue daq:dsd
    ];
  rdfs:comment "Defines a quality graph which will contain all metadata about
    quality metrics on the dataset." ;
  rdfs:label "Quality Graph Statistics" .
```

Listing 6.1: The quality graph definition.

[4] Available at `http://purl.org/eis/vocab/daq` with content negotiation

[5] `http://ontologydesignpatterns.org/wiki/Category:ArchitecturalOP`. Date Accessed 3rd October 2016

[6] All namespace prefixes are according to `http://prefix.cc` on 1st August 2016 unless stated otherwise.

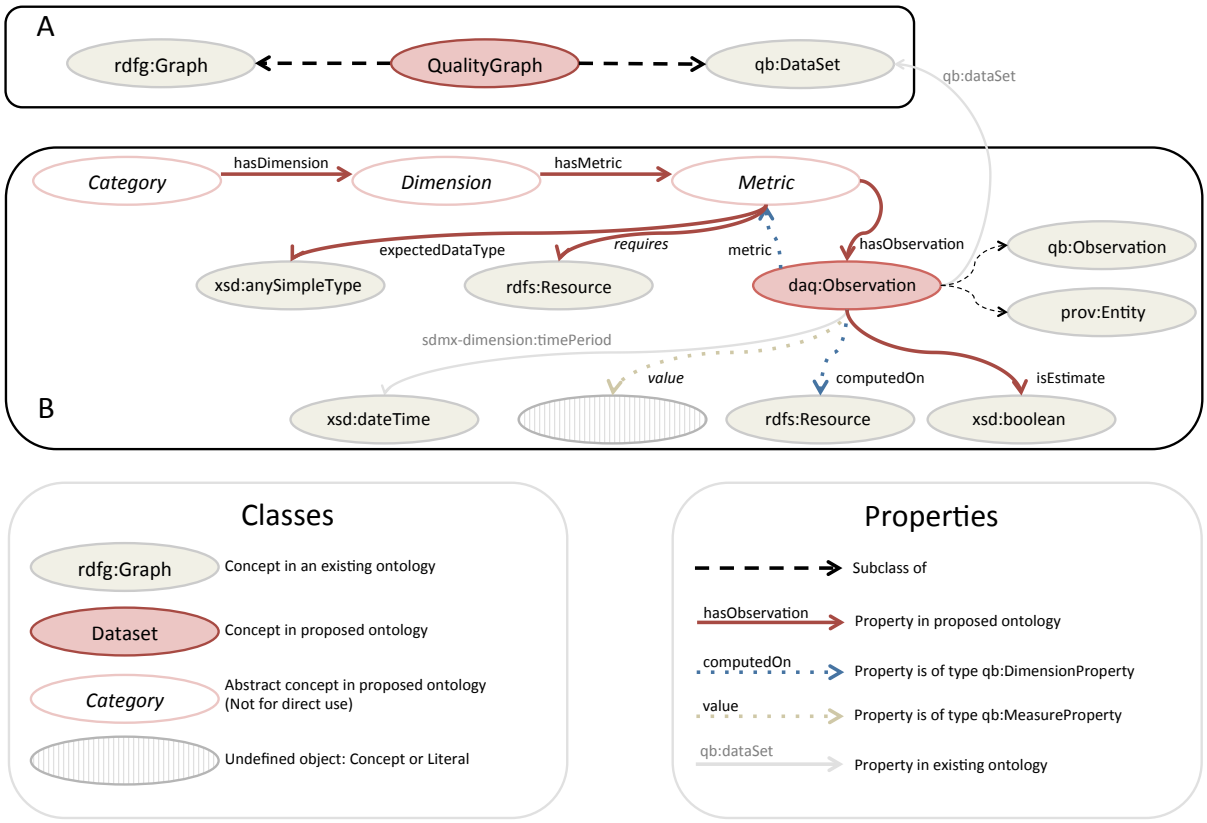

Figure 6.1: The Dataset Quality Vocabulary (daQ).

`daq:QualityGraph` is a subclass of `rdfg:Graph` [36]. This means that quality metadata is aggregated, stored and managed in a named graph (cf Section 2.2.1) that is on the one hand separate from the dataset, but on the other hand embedded into the dataset itself. Furthermore, named graphs can be digitally signed, thus ensuring trust in the computed metrics and defined named graph instance [36].

The *Quality Graph* is also a special type of `qb:DataSet`, which allows us to conceive a collection of quality observations as a data cube complying with a defined dimensional structure (`daq:dsd` – see Listing 6.2). This structure is defined as an OWL property restriction on the property `qb:structure`. Having a standard definition ensures that all *Quality Graph*s conform to a common data structure definition, and thus ensures that all datasets with attached quality metadata can be compared. Three dimensions are applied in this data structure: *metric* (via the `daq:metric` property), *dataset* (via the `daq:computedOn` property), and *time* (via the `sdmx-dimension:timePeriod` property). This enables data cube viewers such as *CubeViz* [109] or *Payola* [76] to visualise quality metadata according to different dimensions. The *time* dimension enables a view on how a dataset's quality evolved over time. Similarly, the *dataset* dimension enables a view of how multiple datasets fare in one or more metrics. Finally, the *metric* dimension enables the view of selected metrics over the assessed datasets.

6.2.2 Quality Representation

Quality metadata comprises three levels of abstraction (as illustrated in Figure 6.1 – Box B): *Categories*, *Dimensions*, and *Metrics*. To formalise this three level abstraction, two inverse role expressions (that are

```
daq:dsd a qb:DataStructureDefinition ;
  # Dimensions
  qb:component [ qb:dimension daq:metric ; qb:order 1 ; ] ;
  qb:component [ qb:dimension daq:computedOn ; qb:order 2 ; ] ;
  qb:component [ qb:dimension sdmx-dimension:timePeriod ; qb:order 3 ; ] ;
  # Measures
  qb:component [ qb:measure daq:value ; ] .
```

Listing 6.2: The data structure definition.

not in the daQ schema[7]) are introduced for convenience:

$$inDimension \equiv hasMetric^{-} \tag{6.1}$$

$$inCategory \equiv hasDimension^{-} \tag{6.2}$$

Using these newly defined properties, we can define the three level abstraction as:

$$\begin{aligned} C &\sqsubseteq Category \\ D &\sqsubseteq Dimension \sqcap \exists\, inCategory.C \\ M &\sqsubseteq Metric \sqcap \exists\, inDimension.D \end{aligned} \tag{6.3}$$

where C is a possible quality measure category, $D = \{d_1, d_2, \ldots, d_y\}$ is the set of all possible quality measure dimensions, $M = \{m_1, m_2, \ldots, m_z\}$ is the set of all possible quality measure metrics, and $x, y, z \in \mathbb{N}$.

Using the definition in Equation 6.3, we describe the quality metadata in Equation 6.4, where the graph g entails one or more instances of a category c, having one or more dimensions d, each of which defines one or more metrics m.

$$\begin{gathered} g : \textbf{QualityGraph} \,;\, c : \textbf{C} \,;\, d : \textbf{D} \,;\, m : \textbf{M} \,; \\ c\ \textbf{hasDimension}\ d \,;\, d\ \textbf{inCategory}\ c \,; \\ d\ \textbf{hasMetric}\ m \,;\, m\ \textbf{inDimension}\ d \,; \\ g \models \{c : \textbf{C}\} \end{gathered} \tag{6.4}$$

Recommended Best Practice - Constraining the Grouping of Dimensions and Metrics

Whilst we acknowledge that a metric might be perceived in one or more categories by different parties, the main goal of the daQ model is to create a generic schema that allows the semantification of quality measures in the abstract three level hierarchy. When defining these quality measures, a common understanding is required (the unified view presented by the survey in [160] is a good example for this), such that these measures can be used by any framework without any prejudice. If such restrictions were not in place, *doubt* and *ambiguity* might be created between the consumers and the quality assessors (who could be a data enthusiast or the publisher himself). One fundamental aspect of assessing a dataset for its quality is to start eliminating the doubts consumers may have about the data. Therefore, if the same metric has multiple definitions and categorisations, incertitude is created for the consumer. On the other

[7] The `owl:inverseOf` properties were avoided in order to ensure only one standard way of defining the three level abstraction.

	Term	Description	Usage[8]
Classes	`daq:Category`	The broadest class of quality observations; groups a number of dimensions related to each other.	`:Accessibility a rdfs:Class ; rdfs:subClassOf daq:Category .`
	`daq:Dimension`	Each dimension belongs to exactly one category. Each dimension comprises a number of metrics. A dimension is linked with a category using the `daq:hasDimension` property.	`:Availability a rdfs:Class ; rdfs:subClassOf daq:Dimension .`
	`daq:Metric`	The smallest unit of measuring quality; belongs to exactly one dimension. Each metric has one or more observations (`daq:hasObservation`), which records data quality assessment value resulting from a computation.	`:RDFDataDump a rdfs:Class ; rdfs:subClassOf daq:Metric .`

Table 6.1: Description and usage of daQ abstract classes.

hand, the quality assessor ends up in an ambiguous situation to try to identify the right quality measure structure. In Equation 6.5 we present these restrictions as a set of optional axioms in daQ.

$$\begin{aligned} C &\sqsubseteq \mathit{Category} \\ D &\sqsubseteq \mathit{Dimension} \sqcap \exists\, \mathit{inCategory}.(=1\ C) \\ M &\sqsubseteq \mathit{Metric} \sqcap \exists\, \mathit{inDimension}.(=1\ D) \end{aligned} \tag{6.5}$$

6.2.3 Abstract Classes and Properties

Abstraction – "hiding" the complex meaning of an object whilst retaining the most generic view of it – is one of the main principles in the object oriented paradigm. The rationale behind using abstraction in daQ is to have a lightweight meta-model that is the basis on which tangible concepts of categories, dimensions and metrics can be defined upon. Applications consuming quality metadata, such as Luzzu Web Interface (cf. Section 4.4.5), can easily re-use these defined concepts without having to know any detail about new schemas, but by applying a number of simple SPARQL queries using the daQ schema (cf. Section 6.3), to retrieve quality metadata information from assessed datasets. Abstract classes are not intended to be instantiated (`rdf:type`), but should only be extended by quality measure concepts using the appropriate `rdfs:subClassOf` properties. The idea behind this inheritance is that the semantically defined quality measures (in a schema) follows the daQ structure. Hence, the newly defined subclasses will inherit all properties. This does not inhibit the creation of new concepts and properties in the schema related to the defined quality measures. Unfortunately, this cannot be enforced since in RDF the semantics of *abstraction* is not defined. The daQ vocabulary defines three abstract concepts (`daq:Category`, `daq:Dimension`, `daq:Metric`). High-level description and usage examples of these terms are illustrated in Table 6.1. These three abstract concepts are arranged in a hierarchical manner using the properties `daq:hasDimension` and `daq:hasMetric`, described in Table 6.2.

	Term	Description
Properties	`daq:hasDimension`	The category concept classifies dimensions related to the measurement of quality for a specific criteria.
	`daq:hasMetric`	A dimension is an abstract concept which groups an number of more concrete metrics to measure quality of a dataset.

Table 6.2: Description of daQ main properties.

[8] All examples are taken from [160]

	Term	Description	Usage
Class	`daq:Observation`	A quality observation comprises statistical and provenance information related to the assessment of the metric.	`:obs1 a daq:Observation ;`
Properties	`daq:expectedDataType`	Each metric should have an expected data type for its observed value (e.g. `xsd:boolean`, `xsd:double`, etc.). **Range:** `xsd:anySimpleType`	`:RDFDataDump daq:expectedDataType xsd:boolean ;`
	`daq:requires`	A metric implementation might require external resources (e.g. a gold standard) to be able to measure quality. To cater for the most general requirements, this property links a metric to the required resource (e.g. a URI of the gold standard dataset used). **Range:** `rdfs:Resource`	`:RDFDataDump daq:requires <uri:SemanticFormatsFile> ;`
	`daq:hasObservation`	Each metric can have one or more observations attached to it. This property (whose inverse property is *daq:metric*) has the range of `daq:Observation`, defining quality metadata for a particular metric in a specific point in time. **Range:** `daq:Observation`	`:inst1 a :RDFDataDump ; daq:hasObservation :obs1 .`

Table 6.3: Description and usage of metric properties.

6.2.4 Metric Representation

The metric concept is in the domain of three properties: `daq:expectedDataType`, `daq:requires`, `daq:hasObservation`. The former two should be used by metric concepts extending the `daq:Metric` class. On the other hand, the `daq:hasObservation` property (described further in Section 6.2.4) is used by instances of the quality measure concepts. This property has the range of `daq:Observation`, representing statistical (subclass of `qb:Observation`) and provenance information (subclass of `prov:Entity`) of the instantiated metric's assessment activity. A clearer view of the properties usage, together with their description is provided in Table 6.3

Metric Observations

A metric's observation is unquestionably the most important part of a dataset's quality metadata. Each observation resource holds values and properties related to the computed metric, with the main intent of providing an insight to possible data consumers on the use of a particular dataset. A quality assessment should be *replicable* and *traceable*. An observation can have more meta-information through a provenance chain attachment. We recommend that a `daq:Observation` instance should have information about the *activity* (e.g. the algorithm, implementation or workflow that generated the observation (using `prov:wasGeneratedBy`), and the *agent* that the observation is attributed to (using `prov:wasAttributedTo`). Moreover, such observations can have other provenance metadata, such as parameter settings for probabilistic techniques employed to compute certain metrics efficiently, such as those discussed in Chapter 7 and Chapter 8.

Further provenance aware properties include the date and time of the assessment (`sdmx-dimension:timePeriod`). This property is one of the dimensions described in the data structure definition (cf. Listing 6.2) together with `daq:value` and `daq:metric`. An observation should also have an indication whether a quality metric was assessed (`daq:isEstimate`) using some estimation technique. In Table 6.4, the properties whose domain is `daq:Observation` are described together with a usage example.

Term	Description	Usage
`daq:computedOn`	Quality metrics can (in principle) be computed on different datasets. This vocabulary allow the owner/user of such RDF data to calculate metrics on multiple (and different) resources. **Range:** `rdfs:Resource`.	`:obs1 daq:computedOn <http://dbpedia.org> ;`
`daq:metric`	Represents the metric being observed. Inverse of `daq:hasObservation`. **Range:** `daq:Metric`.	`:obs1 daq:metric :RDFDataDump ;`
`daq:value`	Each metric will have a value computed. To deal with the different return types of the metric computation, this property links a metric with a value object (e.g. boolean, double). The value's datatype is related to the metric's expected type. **Range:** open	`:obs1 daq:value "true"^^xsd:boolean ;`
`daq:isEstimate`	This property flags true if an assessed observation of a metric gives an estimate result instead of a more accurate one. **Range:** `xsd:boolean`	`:obs1 daq:isEstimate "false"^^xsd:boolean ;`
`sdmx-dimension:timePeriod`	Holds a time stamp for the completion of the assessment. **Range:** `xsd:dateTime`	`:obs1 sdmx-dimension:timePeriod "2015-03-27..."^^xsd:dateTime ;`

Table 6.4: Description and usage of properties in a metric's observation.

6.3 Creating and Using the Quality Metadata

6.3.1 Extending daQ with Custom Quality Metrics

Extending the daQ vocabulary means adding new quality measures that inherits the abstract concepts (Category-Dimension-Metric). Figure 6.2 illustrates how the daQ vocabulary can be extended with specific quality measures (T-Box), and how these can be used to represent concrete quality metadata (A-Box). The T-Box illustrates the representation of the *RDF Availability* metric, which is part of the *Availability* category in the *Accessibility* dimension.

In accordance with LOD best practices, extensions by custom quality metrics should be made in the extenders' own namespaces. Extending the daQ vocabulary with additional metrics assumes that their exact semantics (such as how they are to be computed) is understood by the software implementation. Therefore, a user extending the daQ would not normally need to specify the technical requirements of the

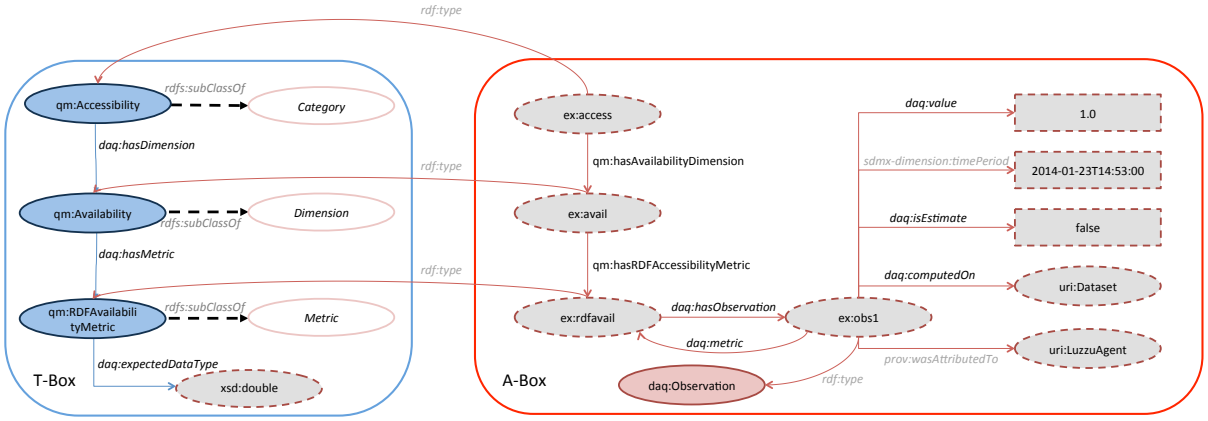

Figure 6.2: Extending the daQ vocabulary – T-Box and A-Box.

quality metric, although pointers to such requirements descriptions can be given via specialisations of the `daq:requires` property. As part of this thesis, a number of quality metrics identified from the survey in [160] were semantically described[9], and used for creating the quality metadata of a number of datasets (cf. Chapter 9).

6.3.2 Querying daQ Metadata

Extending daQ enables data consumers to re-use a number of generic queries for tasks such as identifying all Categories, Dimensions and Metrics, or retrieving the latest quality observations made on a dataset. This section provides queries that can be performed on the schema level for validation purposes, e.g., checking the recommendation constraints defined in Section 6.2.2, and also on the metadata level.

Querying on a Schema Level

Having a schema complying with the daQ vocabulary means that its subsequent metadata can be used within any framework that employs daQ as its underlying model as a means to explore datasets based on their quality. Ideally, users of daQ comply with the best practices constraints, i.e. that a dimension should be in exactly one category, and a metric should be exactly in one dimension. This is not enforced, and users can still use daQ metadata that are not abiding by these best practices.

Constraint 1 – A Dimension should be in exactly one Category. The SPARQL query in Listing 6.3 checks if there are any dimensions in the user defined schema that are violating this best practice, that is, returns *true* if a dimension is linked with more than one category, or with no categories. This constraint is correlated with the second statement in Equation 6.3, and in addition it checks that a dimension belongs to exactly one category.

Constraint 2 – A Metric should be in exactly one Dimension. Similar to Constraint 1, the SPARQL query in Listing 6.4 checks if the best practice of having a metric in exactly one dimension is violated or not.

Listing of Categories, Dimensions and Metrics of a Schema.

In order to retrieve all compliant Categories, Dimensions and Metrics from a schema (Listing 6.5), two inner `SELECT` statements are used in order to get dimensions linked to one category and metrics linked to one dimension.

Querying the Metadata

Quality metadata can be imported into RDF Data Cube visualisation applications such as *CubeViz* [109], *Payola* [76], and Linked Data Visualisation Model Interface (LDVMi) [96], since the daQ is built on the Data Cube vocabulary. Furthermore, SPARQL can be used to query the quality metadata and retrieve quality results. In order to retrieve the latest observed quality metric values of a dataset using SPARQL, a `FILTER NOT EXIST` clause is used in order to eliminate any triples that have a date value older than the newer observation. Listing 6.6 illustrates this query.

[9] http://purl.org/eis/vocab/dqm

```
ASK {
    {
      ?category rdfs:subClassOf daq:Category .
      ?hasDimension rdfs:domain ?category .
      ?hasDimension rdfs:range ?dimension .
      {
        SELECT DISTINCT ?dimension {
          ?dimension rdfs:subClassOf daq:Dimension .
          ?hasDimension rdfs:domain ?cat .
          ?hasDimension rdfs:range ?dimension .
          ?hasDimension rdfs:subPropertyOf daq:hasDimension .
        }
        GROUP BY ?dimension
        HAVING(COUNT(?dimension) > 1)
      }
    } UNION {
      ?dimension rdfs:subClassOf daq:Dimension .
      MINUS{
        ?category rdfs:subClassOf daq:Category .
        ?hasDimension rdfs:domain ?category .
        ?hasDimension rdfs:range ?dimension .
        ?hasDimension rdfs:subPropertyOf daq:hasDimension .
      }
    }
}
```

Listing 6.3: SPARQL query for constraint 1.

```
ASK {
    {
      ?dimension rdfs:subClassOf daq:Dimension .
      ?hasMetric rdfs:domain ?dimension .
      ?hasMetric rdfs:range  ?metric.
      {
        SELECT DISTINCT ?metric {
          ?metric rdfs:subClassOf daq:Metric .
          ?hasMetric rdfs:domain ?dim .
          ?hasMetric rdfs:range  ?metric.
          ?hasMetric rdfs:subPropertyOf daq:hasMetric .
        }
        GROUP BY ?metric
        HAVING(COUNT(?metric) > 1)
      }
    } UNION {
      ?metric rdfs:subClassOf daq:Metric .
      MINUS {
        ?dimension rdfs:subClassOf daq:Dimension .
        ?hasMetric rdfs:domain ?dimension .
        ?hasMetric rdfs:range  ?metric.
        ?hasMetric rdfs:subPropertyOf daq:hasMetric .
      }
    }
}
```

Listing 6.4: SPARQL query for constraint 2.

```
SELECT DISTINCT ?category ?dimension ?metric {
  ?category rdfs:subClassOf daq:Category .
  ?hasDimension rdfs:domain ?category .
  ?hasDimension rdfs:range ?dimension .
  ?hasMetric rdfs:domain ?dimension .
  ?hasMetric rdfs:range  ?metric.
  {
    SELECT DISTINCT ?dimension {
      ?dimension rdfs:subClassOf daq:Dimension .
      ?hasDimension rdfs:domain ?cat .
      ?hasDimension rdfs:range ?dimension .
      ?hasDimension rdfs:subPropertyOf daq:hasDimension .
    }
    GROUP BY ?dimension
    HAVING(COUNT(?dimension) = 1)
  }
  {
    SELECT DISTINCT ?metric {
      ?metric rdfs:subClassOf daq:Metric .
      ?hasMetric rdfs:domain ?dim .
      ?hasMetric rdfs:range  ?metric.
      ?hasMetric rdfs:subPropertyOf daq:hasMetric .
    }
    GROUP BY ?metric
    HAVING(COUNT(?metric) = 1)
  }
}
```

Listing 6.5: SPARQL query listing all compliant quality measures.

```
SELECT DISTINCT ?metric ?value {
  ?obs a daq:Observation ;
    daq:metric ?metric_instance ;
    daq:value ?value ;
    sdmx-dimension:timePeriod ?dateTime

  # This filter eliminates any solution (triples) that
  # have a date time older than the newest observation
  FILTER NOT EXISTS {
        ?obs2 daq:metric ?metric_instance ;
              sdmx-dimension:timePeriod ?newerDateTime .
        FILTER (?newerDateTime > ?dateTime)
  }

    ?metric_instance a ?metric .
} ORDER BY ASC(?metric)
```

Listing 6.6: SPARQL query retrieving the latest observed values of each quality measure for a datasets.

6.3.3 Adding Provenance Value in daQ Observations

Provenance is an important aspect in the Web of Data. Providing provenance information with each observation could, in principle, enable data consumers to trust both the quality metadata and thus also the dataset's actual quality. In [89], Janowicz demonstrates that trust and provenance information goes

hand-in-hand. Information regarding observations on the quality of a dataset is of utmost importance, since consumers can trace back facts such as which authority computed a particular observation, or how a value was computed by reporting details about the assessment activity. Since a `daq:Observation` is a subclass of `prov:Entity`, such provenance information is easily attached to daQ instances, as shown in Listing 6.7.

```
# quality metadata triples
  ex:AnObservation  a daq:Observation ;
    daq:computedOn  <http://somedataset.example> ;
    daq:isEstimate  true ;
    daq:metric  ex:DereferenceabilityMetricInstance ;
    daq:value "0.8942"^^xsd:double ;
    cube:dataSet  ex:someQualityGraph
    sdmx-dimension:timePeriod "2016-04-22T01:27:51.899Z"^^xsd:dateTime ;
    prov:wasGeneratedBy ex:AssessmentActivity .

  ex:AssessmentActivity a prov:Activity ;
    prov:wasAssociatedWith  ex:AnAgent ;
    ex:estimateTechnique  <http://dbpedia.org/resource/Reservoir_sampling> ;
    ex:parameter  "50000"^^xsd:integer ;
    ex:totalTriples "250000"^^xsd:integer ;
    ex:totalTriplesAssessed "50000"^^xsd:integer .
# further metadata triples
```

Listing 6.7: Attaching provenance information to a metric's observation.

6.4 Adoption of daQ in the W3C Data Quality Vocabulary

The daQ model serves as a basis for a broader Data Quality Vocabulary (DQV)[10] initiative by the W3C Data on the Web Best Practices working group. The goal of the Data on the Web Best Practices W3C working group is to derive a set of guidelines (best practices) that should be used for data exchange over the Web [105]. The Data Quality Vocabulary (DQV) presented in [5] is envisaged to act as a model that covers many quality aspects of a dataset. Whilst daQ caters for generic data quality measures and can be extended as required, the DQV goes a step further, putting emphasis on feedback, annotation, agreements and quality policies, all of which describe the quality of a dataset. In DQV, daQ is adapted and used as the core component to describe quantitative quality measures (for example having a measure for *dereferenceability*) in a dataset's quality metadata. The two main differences are:

- the Category, Dimension and Metric concepts are defined as abstract classes in daQ with quality indicators defined as subclasses of these concepts, whilst in DQV these are "normal" concepts with quality indicators defined as instances;
- the systematic organisation of quality indicators in DQV is inverse to that in daQ.

6.4.1 Converting from daQ to DQV and back

All of the DQV concepts that are equivalent to daQ concepts are marked as such using the OWL terminology. Therefore, converting metadata from one schema to another is just a matter of inferring statements using an OWL reasoner. Table 6.5 lists all the equivalent concepts.

[10] `http://www.w3.org/ns/dqv#`. Last Access: 1st July 2016

DQV Concepts	Equivalent daQ Concepts
dqv:computedOn	daq:computedOn
dqv:hasQualityMeasurement	daq:computedOn⁻
dqv:value	daq:value
dqv:expectedDatatype	daq:expectedDatatype
dqv:isMeasurementOf	daq:metric⁻
dqv:inDimension	daq:hasMetric⁻
dqv:inCategory	daq:hasDimension⁻
dqv:Category	daq:Category
dqv:Dimension	daq:Dimension
dqv:Metric	daq:Metric
dqv:QualityMeasurement	daq:Observation
dqv:QualityMeasurementDataset	daq:QualityGraph

Table 6.5: Equivalent concepts between DQV and daQ.

Listing 6.8 shows a T-Box example of how quality indicators are defined in daQ, and a subsequent A-Box example of quality metadata in daQ. We used an out-of-the-box OWL reasoner in order to infer A-Box DQV statements from our example, whose results can be seen in Listing 6.9.

6.5 daQ Validator

The daQ validator[11] is an online tool that enables the validation of schemas extending daQ. This tool validates the daQ constraints described in Listings 6.3 and 6.4, by executing the SPARQL queries described in the previous section on the schema. It also lists all Categories, Dimensions and Metrics (Listing 6.5) in a user friendly fashion, enabling the easy visualisation of the underlying RDF representation. The tool also provides warnings if a metric or a dimension is not linked to a dimension or category respectively, and an error if a metric or a dimension is linked to multiple dimensions or categories. Figure 6.3 depicts the visualisation screen listing all quality measures in the daQ compliant schema.

[11] http://jerdeb.github.io/daqvalidator/

```
# T-BOX
dqm:Representational  a rdfs:Class ;
  rdfs:subClassOf daq:Category .

dqm:Interoperability  a rdfs:Class ;
  rdfs:subClassOf daq:Dimension .

dqm:ReuseExistingVocabularyMetric a rdfs:Class ;
  rdfs:subClassOf daq:Metric ;
  daq:expectedDataType xsd:double .

dqm:hasInteroperabilityDimension  a rdfs:Property ;
  rdfs:subPropertyOf daq:hasDimension ;
  rdfs:domain dqm:Representational ;
  rdfs:range dqm:Interoperability .

dqm:hasReuseExistingVocabularyMetric  a rdfs:Property ;
  rdfs:subPropertyOf  daq:hasMetric ;
  rdfs:domain dqm:Interoperability ;
  rdfs:range  dqm:ReuseExistingVocabularyMetric .

# A-BOX
ex:31ff158c-98a1-4461-bb26-564f6c2b3d2e a dqm:Representational ;
    dqm:hasInteroperabilityDimension ex:f4b85d8e-0fe4-4f55-950e-d26d195735fd .

ex:f4b85d8e-0fe4-4f55-950e-d26d195735fd a dqm:Interoperability ;
    dqm:hasReuseExistingTermsMetric ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c .

ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c a dqm:ReuseExistingVocabularyMetric ;
    daq:hasObservation ex:39df34d5-053f-4e16-b7c6-04517166e294 .

ex:39df34d5-053f-4e16-b7c6-04517166e294 a daq:Observation ;
    daq:computedOn <http://somedataset.org/> ;
    daq:isEstimate false ;
    daq:metric ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c ;
    daq:value 2.272727e-01 ;
    qb:dataSet ex:32e37417-9a94-4c48-a8fa-ba02023b9ae5 ;
    sdmxdim:timePeriod "2016-04-22T01:27:51.782000+00:00"^^xsd:dateTime .
```

Listing 6.8: A T-Box and A-Box snippet of defined quality indicators using daQ and the subsequent quality metadata using the daQ schema.

```
ex:31ff158c-98a1-4461-bb26-564f6c2b3d2e a daq:Category, rdfs:Resource, owl:Thing,
    dqv:Category ;
    daq:hasDimension ex:4b85d8e-0fe4-4f55-950e-d26d195735fd ;
    owl:sameAs ex:31ff158c-98a1-4461-bb26-564f6c2b3d2e .


ex:f4b85d8e-0fe4-4f55-950e-d26d195735fd a daq:Dimension, rdfs:Resource, owl:Thing,
    dqv:Dimension ;
    owl:sameAs ex:f4b85d8e-0fe4-4f55-950e-d26d195735fd ;
    dqv:hasCategory ex:31ff158c-98a1-4461-bb26-564f6c2b3d2e .

ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c a daq:Metric, rdfs:Resource, owl:Thing, dqv
    :Metric ;
    owl:sameAs ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c .

ex:39df34d5-053f-4e16-b7c6-04517166e294 a qb:Observation, rdfs:Resource, owl:Thing,
     dqv:QualityMeasure ;
    owl:sameAs ex:39df34d5-053f-4e16-b7c6-04517166e294 ;
    dqv:computedOn <http://somedataset.org/> ;
    dqv:hasMetric ex:906bebb4-a0fe-4b0e-ba25-c95ccfca246c ;
    dqv:value 2.272727e-01 .

<http://somedataset.org/> dqv:hasQualityMeasure ex:39df34d5-053f-4e16-b7c6-04517166
    e294 .
```

Listing 6.9: An inferred (snippet) version of Listing 6.8 using the DQV schema.

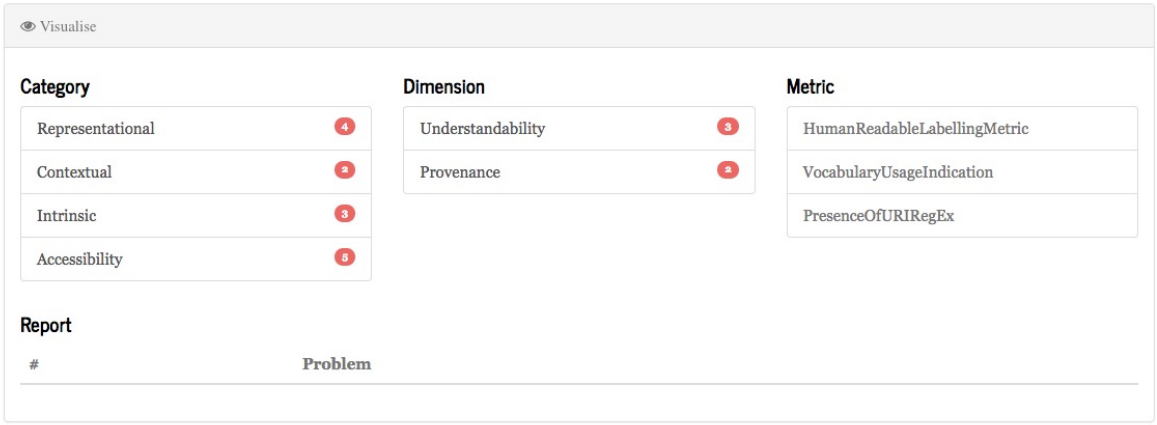

Figure 6.3: A Screenshot from the daQ Validator. The interface is divided into three parts, each part representing the three level abstraction - Category, Dimension, Metric. The numbers, for example 2 in *Contextual* category means that there are two dimensions in that category.

6.6 Concluding Remarks

In this chapter, we presented the Dataset Quality Vocabulary (daQ), a core vocabulary for representing quality assessment results of linked open datasets. daQ is a lightweight abstract metadata model, providing a pragmatic solution to defining quality concepts, making them interoperable with other defined quality

concepts defined using the daQ meta-model. daQ follows the best practice of re-using Linked Data vocabularies in that its core concepts are built on the RDF Data Cube Vocabulary and the Provenance Ontology, both W3C standards. The transparent metadata approach enables, amongst others, consumers to chose the most appropriate dataset for their task, whilst fostering competition among publishers to improve their datasets. Therefore, with daQ we provide means to tackle part of the first challenge (cf. Section 1.1) regarding *making information about quality accessible*, and provide an answer for the defined RQ2 in Section 1.3.

However, from a social perspective, we still need to pervade the idea of having quality metadata attached to datasets to data publishers. Publishers might be doubtful about the data they publish, and thus are afraid of having such metadata included with their dataset, thus preferring staying in their comfort zone. This metadata approach that we are suggesting might be seen as a "label" of quality, and thus publishers believe that having such information available might scare away (or in a positive way, get more) possible customers. However, as datasets evolve, our approach enables data publishers to show that their datasets improved over time. Furthermore, publishers should be convinced that having quality information about their dataset visible might attract a different spectrum of consumers. In this regard, the W3C Data on the Web Best Practices[12] is trying to foster this idea amongst open data publishers, by publishing best practices and other recommendations, and by reaching out to the audience via various communication channels (e.g. mails[13], tweets[14], etc . . .).

Since its initial development in 2014, the daQ vocabulary has been both used and planned to be used in a number of initiatives and development projects. Currently, it is one of the main contributions to a broader Data Quality Vocabulary (DQV) initiative started by the Data on the Web Best Practices group[15] [5]. Other projects making use of daQ include:

- **DIACHRON**[16] - an EU project that aimed at preserving the evolving Data Web, using daQ metadata to rank and filter harvested datasets [50];
- **eENVplus**[17] - aims to integrate a large amount of environmental data, making use of daQ to semantically represent quality results and using third party visualisation to enable users to browse through the metadata [4];
- **EEXCESS**[18]- a recommendation tool that makes use of daQ together with the Data Quality Vocabulary (DQV) proposed in [5] in order to report the quality EEXCESS records [125];
- **WDAqua**[19] - aiming at creating a data-driven question answering infrastructure, the daQ serves as the underlying data model to represent quality metadata to make quality-based decisions on the source selection [146].

Currently, using daQ, we semantically defined[20] a number of domain-specific and domain-independent metrics, following a survey of Linked Data quality metrics [160]. Since quality metadata describes the

[12] https://www.w3.org/2013/dwbp. Date Accessed on 7th October 2016
[13] DWBP public mailing list archive: http://lists.w3.org/Archives/Public/public-dwbp-wg. Date Accessed 26th September 2016
[14] From various WG members using the handle #DWBP. https://twitter.com/hashtag/DWBP. Date Accessed 26th September 2016
[15] The first author is a member of this Working Group, contributing towards this vocabulary.
[16] Part of this thesis was an output from this project where we were responsible for the Appraisal and Ranking Service http://www.diachron-fp7.eu. Date Accessed on 7th October 2016
[17] http://www.eenvplus.eu. Date Accessed on 7th October 2016
[18] http://eexcess.eu. Date Accessed on 7th October 2016
[19] http://wdaqua.eu. Date Accessed on 7th October 2016
[20] http://purl.org/eis/vocab/dqm

co-evolution of a dataset quality, it makes sense to maintain it close to the dataset. Having good quality Linked Datasets ensures their re-usability, and thus helping in creating a higher quality Web of Data. The daQ vocabulary provides a mean to represent quality information (metadata) regarding a Linked Dataset in a transparent manner, thus ensuring a maximum impact of the quality assessment.

Part IV

Scaling Quality Metrics for Big Linked Datasets

In Chapter 4 we described Luzzu as a generic framework that is extensible by third party quality metrics. We showed that Luzzu itself is scalable, though for the whole assessment to be time efficient, the metrics implemented for Luzzu should also be scalable. Furthermore, certain metrics might prove to present an intractable time, as dataset size increase. Keeping in line with research question 3 defined in Section 1.3, in this part we describe a number of techniques in order to investigate whether probabilistic techniques and distance-based clustering can provide a good approximative quality value in an acceptable time. In Chapter 7, we employ probabilistic techniques such as sampling, Bloom Filters and clustering coefficient estimation for implementing a broad set of data quality metrics in an approximate but sufficiently accurate way. In this thesis we extend our work in [49] by comparing two different sampling techniques, namely reservoir sampling and stratified sampling. In Chapter 8 we present a preliminary approach for identifying potentially incorrect RDF statements using distance-based outlier detection.

Chapter 7 is an updated version of the following publication:

- **Jeremy Debattista**, Santiago Londoño, Christoph Lange, Sören Auer. *Quality Assessment of Linked Datasets using Probabilistic Approximation.* (Nominated for the Best Paper Award) In 12th European Semantic Web Conference Proceedings 2015, 221-236, Springer.

Chapter 8 is based on the following publication:

- **Jeremy Debattista**, Christoph Lange, Sören Auer. *A Preliminary Investigation Towards Improving Linked Data Quality using Distance-Based Outlier Detection.* To Appear at Joint International Semantic Technology Conference 2016

CHAPTER 7

Quality Assessment of Linked Datasets using Probabilistic Approximation

The wide spectrum of Linked Data quality indicators brings about various degrees of computational complexities. However, this varying complexity can be a problem when assessing large datasets, as metrics can become intractable. In a recent article, Dan O'Brien [124] discusses how Big Data, which is now being applied in many scenarios and applications, challenges data governance, an aspect that includes data quality. However, it is equally important that assessing the quality of large datasets is not a process that can be achieved only by high-end Big Data machines.

In relation to challenge regarding the scalability of quality metrics (cf. Section 1.1), in this chapter we discuss and apply a number of probabilistic techniques to assess Linked Data quality. In particular, we employ three approximation techniques commonly used in Big Data applications: *Sampling*, *Bloom Filters* and *Clustering Coefficient estimation*.

The main contribution of this chapter is that we develop strategies to showcase (Section 7.2) how approximation techniques can be applied to boost the running time of quality metric computations for big linked datasets. Therefore, with such techniques we aim to improve the scalability (related to RQ 3 in Section 1.3) of the whole quality assessment process for large linked datasets. We also thoroughly evaluate (Section 7.3) the quality metrics to tweak the required parameters for more accurate results yet keeping the running time acceptable, using a low-end machine for the evaluation setup. All implemented quality metrics are part of *Luzzu* (cf. Chapter 4).

7.1 Linked Data Metrics

Zaveri et al. present a comprehensive survey [160] of quality metrics for linked open datasets. Most of the quality metrics discussed are *deterministic* and computable within *polynomial time*. On the other hand, once these metrics are exposed to large datasets, the metrics' upper bound grows and as a result, the computational time becomes intractable. In this section we discuss some metrics that are known to suffer from this phenomenon.

7.1.1 Dereferenceability

HTTP URIs should be dereferenceable, i.e. HTTP clients should be able to retrieve the resource identified by the URI. A typical web URI resource would return a `200 OK` code indicating that a request is successful and a `4xx` or `5xx` code if the request is unsuccessful. In Linked Data, a successful request

should return an RDF document containing triples that describe the requested resource. Resources should either be *hash* URIs or respond with a `303 Redirect` code [142]. The dereferenceability metric assesses a dataset by counting the number of valid dereferenceable URIs (according to these LOD principles) divided by the total number of URIs. Yang et. al [159] describe a mechanism[1] to identify the dereferenceability process of a Linked Data resource.

A naïve approach for this metric is to dereference all URI resources appearing in the subject and the object of all triples. In this metric we assume that all predicates are dereferenceable. This means that the metric performs at worst $O(2n)$ HTTP requests, where n is the number of triples. It is not possible to perform such a large number of HTTP requests in an acceptable time.

7.1.2 Existence of RDF Links to External Data Providers

This metric measures the degree to which a resource is linked to external data providers. Ideally, datasets have a high degree of linkage with external data providers, more specifically links to external RDF resources, since interlinking is one of the main principles of Linked Data [22].

The simplest approach for this metric is to compare all objects' resource PLD[2]. Whilst this metric computes linearly and the power of the polynomial is low (at worst $O(n)$, where n represents the number of triples), this is multiplied with a big constant factor as all resources have to be dereferenced (therefore requires network operations) for Linked Data descriptions

7.1.3 Extensional Conciseness

At the data level, a linked dataset is concise if there are no redundant instances [114]. This metric measures the number of unique instances found in the dataset. The uniqueness of instances is determined from their properties and values. An instance is unique if no other instance (in the same dataset) exists with the same set of properties and corresponding values.

The most straightforward approach is to compare each resource with every other resource in the dataset to check for uniqueness. This gives us a time complexity of $O(i^2t)$, where i is the number of instances in the datasets and t is the number of triples. The major challenge for this algorithm is the number of triples in a dataset, since each triple (predicate and object) is compared with every other triple streamed from the dataset.

7.1.4 Clustering Coefficient of a Network

The clustering coefficient metric is proposed as part of a set of network measures to assess the quality of Linked Data mappings [61]. This metric aims to identify how well resources are connected, by measuring the density of the resource neighbourhood. A network has a high clustering cohesion when a node has a large number of neighbouring nodes, all of which are connected to each other. Guéret et al. [61] explains that having a fully connected graph is not ideal in the Web of Data, since most links could result to be meaningless, as the Web of Data should be more oriented towards a *small world* of topic clusters. For example, it might not be meaningful to connect a cluster of links related to electronics and another cluster related to animals. However, unlike the Web of Data, *local* datasets (for example, LinkedGeoData[3] is a dataset related to geographic matters) are usually topic-centric and thus the higher the coefficient means

[1] Also used in the Semantic Web URI Validator Hyperthing (`http://www.hyperthing.org`)

[2] "PLDs allow us to identify a realm, where a single user or organisation is likely to be in control." [120] against the data source PLD being assessed, and if they are different, then this objected checked to see if a Linked Data description is available at that URI. For example the PLD for `http://dbpedia.org/resource/Malta` is `dbpedia.org`.

[3] `http://linkedgeodata.org/`. Date Accessed 27th September 2016

Probabilistic Approximation Technique	**Linked Data Metric**
Sampling	Dereferenceability
	Links to External Data Providers
Bloom Filters	Extensional Conciseness
Clustering Coefficient Estimation	Clustering Coefficient of a Network

Table 7.1: Mapping probabilistic approximation techniques with Linked Data quality metrics.

that there is more cohesion between the links. Therefore, high clustering coefficient is a sign of good quality in a linked dataset.

When assessing the clustering coefficient of a network, a graph is built where the *subject* and *object* of a triple (either URI resources or blank nodes) are represented as vertices in the graph, whilst the *predicate* is the edge between them. As this ignores triples with literal objects, there is no direct correlation between the number of triples in a dataset and number of vertices. Calculating this measure on a network takes at most $O(n^3)$, where n is the number of nodes in the network. This is because each vertex in the network has to be considered: for each vertex v in the graph (takes $O(n)$), we first identify the neighbours of v, and then the number of links between the neighbours of v (i.e. how many of v's neighbours are connected together), which takes $O(n^2)$.

7.2 Implementation

Based on the probabilistic techniques described in Section 2.3, we analyse how they can help in assessing quality in linked datasets. These metrics are implemented as quality metrics for *Luzzu* (cf. Chapter 4). Table 7.1 shows which approximation can be used for each respective metric.

7.2.1 Metrics using Sampling Technique

Approach 1: Reservoir Sampling

Our implementation is based on the *rejection-acceptance* technique [151], described in Section 2.3.1. The trade-off parameter is the definition of the maximum number of items (k) that can be stored. Various factors have to be taken into consideration to define k, such as the rough estimation of the size of the dataset and available memory, since this reservoir is stored in-memory.

When attempting to add an *item* to the reservoir sampler, an item counter (n) is incremented. This increment is required to calculate the *replacement probability*, since the exact size of the source (in our case the dataset) is unknown. The *item* can be (i) *added* to the reservoir, (ii) become a *candidate* to replace another item, or (iii) be *discarded*. The first possible operation is straightforward. If the reservoir sampler has free locations ($n < k$), the *item* is added. On the other hand, when the reservoir is full, the *item* can either replace another item in the list, or rejected. The decision is made by generating a random number (p) between 0 and n. If p lies in the range of the reservoir list length (i.e. $p < k$), then the new *item* replaces the current item stored in that position of the reservoir, else it is rejected. This simulates the k/n *replacement probability* for all items. In order to exploit this technique to its full potential, and have the best possible sample, the whole dataset is parsed, however the quality assessment is then made on just the sampled objects.

Estimated Dereferenceability Metric

Each resource URI is split into two parts: (1) the Pay-Level domain (PLD)[4], and (2) the path to the resource. This is analogous to a dictionary data structure. For this metric we employ a "global" reservoir sampler for the PLDs. Furthermore, for each PLD we employ another sampler holding an evenly distributed sample list of resources to be dereferenced. Envisaging the possibility of multiple HTTP requests to the same domain or resource, we make use of Luzzu's caching mechanism, to store HTTP responses. The metric value is calculated as a ratio of the total number of dereferenced URIs against the total number of sampled URIs.

Estimated Links to Dereferenceable External Data Providers Metric

In order to measure the use of external data providers, the metric must first extract the pay-level domain of the data source being assessed. All external resources in the assessed dataset, retrieved from each triple's *object*, are mapped to a reservoir of resources bearing the same external PLD. This means that we have a reservoir for each external PLD. Sampled external PLD resources are then checked for Linked Data descriptions, and if such descriptions exist, then the PLD is considered to be an external RDF data provider. One limitation of using PLDs is that since resources are grouped under their identified PLD, there might be some domains such as `purl.org`, or more recently `w3id.org` that act as permanent identifiers for web resources. Therefore, in this case, our approach resolves these permanent identifiers to their actual location on the web. Moreover, we are aware that datasets might hijack foreign namespaces in the subject position (i.e redefining external resources), thus would require a small modification in the implementation to include such resources. However, in this implementation, we assume that there are no such instances in local linked datasets.

Approach 2: Stratified Sampling

Since the resources can be grouped by their PLD, the data that should be sampled can be easily partitioned. Our implementation is a hybrid of the *rejection-acceptance* technique described in the first approach, and the idea behind *stratified sampling* (cf. Section 2.3.1). Similar to the first approach, the trade-off parameter is the size of the sample, known as the population percentage (p). Whilst the computation time is similar between both approaches, the stratified approach has two steps. The first step is to partition the data based on their PLD. For this partitioning, a two-dimensional reservoir sampler (similar to what was described in the estimated dereferenceability metric), is used in order to have a sample of the dataset prior to the second round of sampling. Therefore, rather than partitioning the whole dataset, we are partitioning a sample of the dataset. Considering that the reservoir might not be big enough and that groups might have the same amount of resources partitioned, we also keep a counter on the actual amount of resources pertaining to each PLD.

Having the partitions and PLD instance count, the second step is to extract a *representative sample* (using a proportionate allocation method) from each partition group (from the two-dimensional reservoir sampler), where the number of elements required from each PLD is calculated as follows:

$$PLD^i_{rs} := PLD^i_c \times p$$

where PLD^i is the PLD being sampled, PLD^i_c is the count of actual resources for that PLD in the assessed dataset. This approach is applied for the dereferenceability metric. The main difference between the

[4] A Pay-Level Domain is the authoritative domain of a URI, e.g. The PLD for `http://www.example.org/` is `example.org`

reservoir sampler and the stratified sampler is that in the latter we carefully choose our sample to represent the original dataset.

7.2.2 Metrics using Bloom Filters

Bera et al. [19] introduced some modifications to the mechanics of Bloom Filters (cf. Section 2.3.2) to enable the detection of duplicate elements in data streams. These modifications allow items to be inserted indefinitely by probabilistically resetting bits in the filter arrays when they are close to getting overloaded. The Randomised Load Balanced Biased Sampling based Bloom Filter (RLBSBF) is used to implement the detection of duplicate instances. The authors show that this approach is efficient and generates a low *false positive* rate.

An RLBSBF algorithm is initialised with (1) the total memory used by filter arrays in bits (M); and (2) a threshold value (t_{FPR}) for the false positive rate. The bit vector is initialised with k Bloom Filters. Each bloom filter has a size of M/k and a hash function is mapped to it. The authors in [19] suggest that k is calculated using the threshold value t_{FPR}. A high threshold value means faster computation but less accurate results.

Whenever a new element is processed, the Bloom Filter sets all k bit positions using the hash functions mapped to them. If the bit positions were previously set in the bit vector, it means that a duplicate was detected. Otherwise, the probabilistic resetting of bits is performed before the new element is added to the bit vector. Figure 7.1 illustrates how Bloom Filters help to identify a Linked Data resource that already exists in a dataset.

Estimated Extensional Conciseness Metric.

When triples are streamed to the metric processor, the *predicate* and *object* are extracted and serialised as a string. The latter string is stored in a sorted set. This process is repeated until a triple with a different *subject* identifier is processed. The sorted set is then flattened to a string and added to the Bloom Filter,

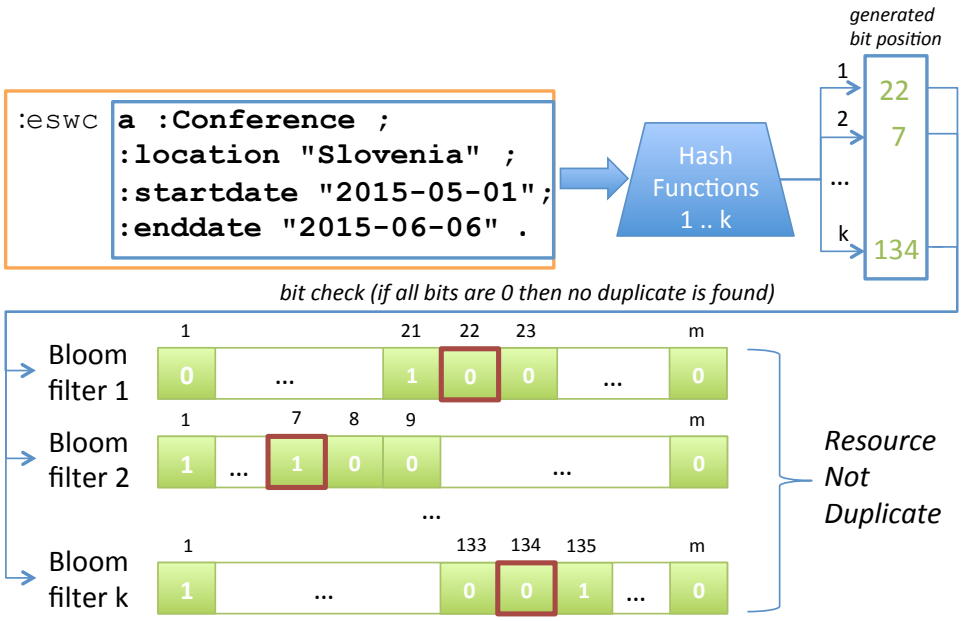

Figure 7.1: Illustrating Bloom Filters with an example.

discovering any possible duplicates. The set is then initialised again for the new resource identifier and the process is repeated until no more triples are streamed.

The main drawback of our proposed algorithm is that a dataset must be sorted by *subject*, such that all triples pertaining to the same instance are streamed one after another. Although it is common practice to publish datasets sorted by subject (e.g. DBpedia), this cannot be guaranteed in the general case. In our experiments we pre-process RDF dumps by converting them to the N-Triples serialisation, which can be sorted by subject in a straightforward way.

7.2.3 Metrics using Clustering Coefficient Estimation

In [62], the authors propose an approach for estimating a social network's clustering coefficient (cf. Section 2.3.3) by creating a random walk. In their proposed algorithm, Hardiman and Katzir use $log^2 n$ as the base *mixing time*, i.e. the number of steps a random walker takes until it converges to a steady-state distribution. However, different network characteristics lead to different *mixing time*s, where well-connected networks have a small (fast) mixing time [119].

To calculate an estimate of the clustering coefficient given a random walk $R = (x_1, x_2, \ldots, x_r)$, Hardiman and Katzir propose the Estimator 4:

Definition 4:

$$\Phi_l = \frac{1}{r-2} \sum_{r-1}^{k=2} \phi_k \frac{1}{d_{x_k} - 1}$$

$$\Psi_l = \frac{1}{r} \sum_{r}^{k=1} \frac{1}{d_{x_k}}$$

$$\hat{c}_l \triangleq \frac{\Phi_l}{\Psi_l}$$

where r is the total number of steps in the random walk R, x_k is the index of the k^{th} node in the random walk, d_{x_k} is the degree of node x_k and ϕ_k represents the value in the adjacency matrix A in position $A_{x_{k-1},x_{k+1}}$.

Estimated Clustering Coefficient Metric.

When triples are streamed into the metric, the vertices are created by extracting the *subject* and the *object*, whilst the *predicate* acts as a directed edge between the two nodes. We use URI resources and blank nodes to create the network vertices. To calculate the estimated clustering coefficient value, a random walk is performed on the graph. Similarly to the approach in [62], we view the graph as undirected. The idea is that if the random walker ends up in a dead-end (i.e. cannot move forward), it can go back to continue crawling the network. Our mixing time parameter is $mlog^2 n$. Since Linked Data advocates interlinking and re-use of resources, we expect that such datasets have a low mixing time. The multiplier factor m thus enables us to increase or decrease the mixing time as required. The reason behind this is to enable a parameter modifier to the base mixing time ($log^2 n$), since it is difficult to find a one size fits all mixing time. Estimator 4 is used to obtain a close estimate of the dataset's clustering coefficient. Finally, the estimated value is subtracted from the *ideal* value 1 (which means that every node is connected to every other node in the local dataset) as described in [61]. Guéret et al. describe ideal values as the *target* goal Linked Data(sets) should attain.

7.3 Metric Analysis and Experiments

Having implemented the metrics using probabilistic approximation techniques, we measure the computed quality metric values and runtime for the approximate metrics and compare them with the actual metrics. For each approximate metric, we experimented with different parameter settings to identify the *ideal* parameter values. All tests are run on a Unix virtual machine with an Intel Xeon 3.00 GHz, with 3 cores and a total memory of 3.8 GB. We chose a number of datasets of varying sizes and covering different application domains. We found these datasets on datahub.io, looking for datasets tagged with the *lod* tag. These are:

- Learning Analytics and Knowledge (LAK) Dataset[5] ≈ 75K triples;
- Lower Layer Super Output Areas (LSOA) Dataset[6] ≈ 280K triples;
- Southampton ECS E-Prints Dataset[7] ≈ 1M triples;
- WordNet 2.0 (W3C) Dataset[8] ≈ 2M triples;
- Sweto DBLP Dataset[9] ≈ 15M triples;
- Semantic XBRL Dataset[10] ≈ 100M triples;

7.3.1 Parameter Setting

In order to maximise accuracy, the parameters of the algorithms have to be tweaked. Therefore, we experimented with different parameter values and analysed the metric results. Parameter settings were obtained by observing the algorithm's parameters in correlation with the datasets and metrics. The rationale behind this experiment is to identify a single parameter that when used in a metric gives acceptable results within reasonable time. The naïve approaches did not finish its computation on all datasets within reasonable time, hence the most ideal single parameter for all datasets could not be identified. However, we still identified the parameter that gave the most accurate result in the least possible time for each technique. Furthermore, the values of all metrics (except for existence of links to external data providers, which returns an integer value, counting the number of external links found in a dataset) return a value between 0 and 1.

The *Dereferenceability* metric was first implemented using reservoir sampling (approach one), then using the stratified sampling as described in the second approach. Table 7.2 shows the time taken (in seconds) and the approximate value for different parameter settings. The biggest time factor in this metric is the network access time, i.e. the time an HTTP request takes to respond. The parameter settings employed for this experiment are: (P1) global reservoir size: 10, PLD reservoir size: 1000; (P2) global reservoir size: 50, PLD reservoir size: 1000; (P3) global reservoir size: 10, PLD reservoir size: 3000; (P4) global reservoir size: 50, PLD reservoir size: 3000; (P5) global reservoir size: 10, PLD reservoir size: 10000; (P6) global reservoir size: 50, PLD reservoir size: 10000. Whilst the approximate metrics completed the computation for all datasets, the exact computation was cancelled for the rest of the

[5] https://datahub.io/dataset/lak-dataset. Date Accessed 27th September 2016.
[6] https://datahub.io/dataset/lower-layer-super-output-areas. Date Accessed 27th September 2016.
[7] https://datahub.io/dataset/southampton-ecs-eprints. Date Accessed 27th September 2016.
[8] https://datahub.io/dataset/w3c-wordnet. Date Accessed 27th September 2016.
[9] https://datahub.io/dataset/sweto-dblp. Date Accessed 27th September 2016.
[10] https://datahub.io/dataset/semantic-xbrl. Date Accessed 27th September 2016.

	Time (s)	Value	Time (s)	Value
Actual (LAK)	1611.642	0.99485		
	Reservoir Sampling		**Stratified Sampling**	
P1	419.489	0.93254	70.088	0.995
P2	423.397	0.93333	71.069	0.996
P3	668.432	0.80418	308.163	0.99066
P4	663.559	0.80639	312.694	0.99133
P5	849.727	0.86549	510.571	0.991
P6	819.379	0.86671	509.007	0.991

	Time (s)	Value	Time (s)	Value
Actual (LSOA)	7703.372	0.744414		
	Reservoir Sampling		**Stratified Sampling**	
P1	415.704	0.61078	244.427	0.744
P2	409.671	0.61447	242.149	0.73
P3	1245.393	0.68574	1070.293	0.7045
P4	1237.327	0.69439	1067.711	0.71218
P5	2051.330	0.72020	1791.630	0.73501
P6	2046.433	0.73178	1794.056	0.74195

Table 7.2: Dereferenceability metric (using both approaches) with different parameter settings.

datasets as the computation was not ready in an acceptable time. Based on the available results from the datasets, we can conclude that the optimal parameter for this metric is close to the P2 setting, using approach two. Furthermore, the results and times are dependent on network factors, such as downtime and latency. For example, when we first conducted the experiments in [49], the results for the LSOA dataset were 0 due to the fact that all resources returned a `4xx/5xx` error[11].

Another application of the Reservoir Sampling was the *Existence of Links to External Data Providers*. Table 7.3 shows the time taken (in seconds) and the estimated value for different parameter settings. The parameter settings used for the PLD sampler were: (P1) 5; (P2) 25; (P3) 100; (P4) 250; (P5) 1000. This means that for P1, if at least one of 5 sampled resources is a Linked Data, then the PLD qualifies as an external PLD resource. The results show that the approximation technique did not record any significant differences against the naïve approach. The approximation technique fares better when there are a number of different PLDs where resources have no Linked Data descriptions. This is due to the fact that the naïve approach has to go through all external resources, whilst the approximation technique would only need to go through a sample of resources for each PLD. The results also show some fluctuations, depending on the number possible external data sources. For example, the Sweto DBLP dataset had around 2090 possible external RDF data sources whilst the Semantic XBRL dataset had only one possible external RDF data source. Therefore, the Sweto dataset will take more time since at least 2090 resources have to be checked for Linked Data content. It is also worth noting that almost all approximate techniques gave the same value as the actual. This means that the reservoirs (for each PLD), except in one case (P1 for the Southampton dataset), were big enough to hold at least one resource with Linked Data content for that PLD. Nonetheless, there might be PLDs that host a variety of resources, Linked Data or not, thus in this case, the reservoirs have to be large enough to hold at lease one resource. For example, in the Southampton dataset, the P1 setting reported only one external link to an RDF data source, instead of two. Therefore, based on the reservoir trade-off and the available results, we deem that the optimal setting is close to P2 (i.e. reservoir sampler size of 25). In Section 9.3.6 we describe how absolute values such as the one returned by this metric can be normalised using a positional-based ranking.

The *Extensional Conciseness* metric was implemented using Bloom Filters. Table 7.4 shows the time taken (in seconds) and the estimated value for different parameter settings. We applied 4 different settings for experimentation: (P1) 2 filters (k) with a size (M) of 1,000; (P2) 5 filters with a size of 10,000; (P3) 10 filters with a size of 100,000; (P4) 15 filters with a size of 10,000,000. This technique showed a lot of potential in the de-duplication process. The time taken in the approximate algorithms are lower than the actual (naïve), with results being almost as accurate. Based on the Bloom Filter trade-off, a setting between P3–P4 would exploit the potential of this technique in assessing the quality of linked datasets with regard to duplication problems.

For the *clustering coefficient* metric we multiplied the base mixing time (i.e. m) of $log^2 n$ with 0.1, 0.5,

[11] In [49], this was also verified manually

	Time (s)	Value
Actual (LAK)	9.583	3
P1	5.088	3
P2	5.162	3
P3	5.212	3
P4	5.324	3
P5	5.838	3

	Time (s)	Value
Actual (LSOA)	29.903	0
P1	1.868	0
P2	3.097	0
P3	4.334	0
P4	6.16	0
P5	16.893	0

	Time (s)	Value
Actual (S'OTON)	66.295	2
P1	44.618	1
P2	50.364	2
P3	39.049	2
P4	49.053	2
P5	57.216	2

	Time (s)	Value
Actual (WN)	13.153	0
P1	7.333	0
P2	7.351	0
P3	7.903	0
P4	7.54	0
P5	7.452	0

	Time (s)	Value
Actual (Sweto)	13498.161	0
P1	5005.891	0
P2	8508.236	0
P3	10050.518	0
P4	9996.632	0
P5	9969.93	0

	Time (s)	Value
Actual (XBRL)	678.175	1
P1	547.053	1
P2	548.125	1
P3	625.847	1
P4	549.42	1
P5	555.347	1

Table 7.3: Existence of links to external data providers metric with different parameter settings.

	Time (s)	Value
Actual (LAK)	81.334	0.994860
P1	1.348	0.621315
P2	1.377	0.962249
P3	1.67	0.993946
P4	2.212	0.994593

	Time (s)	Value
Actual (LSOA)	375.873	1
P1	1.043	0.617729
P2	1.328	0.966795
P3	1.807	0.999240
P4	2.98	1

	Time (s)	Value
Actual (S'OTON)	7366.225	0.737523
P1	24.304	0.512887
P2	20.217	0.782946
P3	17.512	0.783529
P4	20.275	0.660193

	Time (s)	Value
Actual (WN)	96511.334	0.948
P1	7.407	0.570991
P2	11.502	0.885790
P3	17.653	0.900407
P4	35.381	0.844733

Table 7.4: Extensional conciseness metric with different parameter settings.

0.7 and 1.0 respectively to test with fast mixing time. Table 7.5 shows the time taken (in seconds) and the estimated value for different parameter settings. The results show that for the assessed datasets the $log^2 n$ mixing time is not ideal. This is due to the fact that the smallest multiplier setting, i.e. 0.1, proved to be the "closest" to the actual result in all cases. Determining a more accurate average mixing time for assessment of linked datasets, and hence a more accurate estimate as described in Section 2.3.3, requires the evaluation (such as in [119]) of the Web of Data.

	Time (s)	Value
Actual (LAK)	42.729	0.961040
Mixing time 0.1	4.595	0.978220
Mixing time 0.5	4.595	0.997945
Mixing time 0.7	4.766	0.998665
Mixing time 1.0	4.832	0.998974

	Time (s)	Value
Actual (LSOA)	62.618	1
Mixing time 0.1	7.657	0.999995
Mixing time 0.5	6.829	0.999999
Mixing time 0.7	6.561	0.999503
Mixing time 1.0	6.528	0.999999

	Time (s)	Value
Actual (S'OTON)	408.358	0.933590
Mixing time 0.1	46.373	0.993067
Mixing time 0.5	46.362	0.997634
Mixing time 0.7	46.238	0.997939
Mixing time 1.0	46.225	0.998312

	Time (s)	Value
Actual (WN)	9012.454	0.759257
Mixing time 0.1	243.009	0.810405
Mixing time 0.5	248.925	0.999919
Mixing time 0.7	251.396	0.999917
Mixing time 1.0	252.522	0.999967

Table 7.5: Clustering coefficient metric with different parameter settings.

7.3.2 Dereferenceability - Reservoir Sampling vs Stratified Sampling

In Table 7.2 we also compare the reservoir sampling approach and the stratified sampling approach with regard to time and value. For the stratified sampling, we use the same parameters as used in the first approach. We can observe that the time taken in both dataset sizes is significantly lower in the second approach than that of the first approach. Furthermore, the values of the second approach seemed to be closer to the actual value. Therefore, the benefits of lower running time and more accurate metric values when using the stratified approach can be seen from these two datasets.

The improvement on the running time can be narrowed down to the total number of `HTTP GET` requests performed. In the first approach, if we take P2, then the maximum number of `HTTP GET` requests is 50000, since each PLD (maximum 50 in the case of P2) can have at most 1000 data objects to dereference. On the other hand, the second approach will have at most only 1000 `HTTP GET` requests, where these 1000 requests are representing a proportional sample of all data objects. In order to explain this further, let us take into consideration the first dataset evaluated - Learning Analytics and Knowledge (LAK) dataset. Table 7.6 shows the occurrences of each PLD in the subject and/or object of the LAKs dataset's triples.

Domain	Occurrence
http://data.linkededucation.org	129832
http://dbpedia.org	2602
http://purl.org	1
http://dblp.l3s.de	668
http://data.semanticweb.org	216
http://lak12.sites.olt.ubc.ca	1
http://lakconference2013.wordpress.com	1
http://www.ifets.info	1
https://tekri.athabascau.ca	1
http://lak14indy.wordpress.com	2
http://ceur-ws.org	2

Table 7.6: PLDs and the corresponding occurrences in the subject and/or object of the LAKs' dataset's triples.

In this case, the first approach has to check the dereferencability of 1000 resources for the

	Time (s)	Value	Time (s)	Value
SOTON	**Approach One**		**Approach Two**	
P1	242.056	0.02092	129.180	0.02697
P2	242.530	0.02335	163.603	0.02597
P3	638.067	0.01618	327.909	0.02170
P4	647.717	0.01354	333.741	0.02537
P5	1043.133	0.01392	526.649	0.02385
P6	1039.038	0.01452	528.570	0.02546

	Time (s)	Value	Time (s)	Value
WN	**Approach One**		**Approach Two**	
P1	146.576	0.915	85.711	0.913
P2	137.895	0.905	141.599	0.916
P3	347.668	0.92	339.658	0.91166
P4	351.158	0.78633	335.209	0.914
P5	*	*	*	*
P6	*	*	*	*

	Time (s)	Value	Time (s)	Value
Sweto	**Approach One**		**Approach Two**	
P1	5345.065	0.00229	113.744	0.0
P2	7544.902	0.00720	282.019	0.0
P3	5225.544	0.00801	237.847	0.0
P4	14739.160	0.03423	254.679	0.0
P5	7704.807	0.00371	430.79	0.0
P6	6882.000	0.00514	687.682	0.0

	Time (s)	Value	Time (s)	Value
XBRL	**Approach One**		**Approach Two**	
P1	1883.181	0.04595	1912.185	0.034
P2	2102.561	0.03396	2538.540	0.034
P3	2083.078	0.03465	2711.345	0.032
P4	2288.651	0.02965	2714.614	0.033
P5	2295.527	0.02919	2344.056	0.0274
P6	2529.421	0.03059	2897.722	0.0248

Table 7.7: Dereferenceability values with different parameter settings to compare reservoir sampling (approach one) and stratified sampling (approach two).

`data.linkededucation.org` and `dbpedia.org` PLDs, and all other resources with the different PLDs. This approach does not take into consideration the proportion difference between `data.linkededucation.org` and `dbpedia.org`, hence, if `dbpedia.org` resources were not dereferenceable, the quality value would have been affected as the "statistical weight" given to `dbpedia.org` is the same given to `data.linkededucation.org`.

On the other hand, the proportional sample obtained with the stratified sampler looks as follows (the numbers denote the total number of resources with that PLD in the final sample):

- http://data.linkededucation.org - 974
- http://dblp.l3s.de - 5
- http://data.semanticweb.org - 2
- http://dbpedia.org - 20

We can observe that based on a sample of 1000, `data.linkededucation.org` will take the largest chunk, followed by a considerably smaller `dbpedia.org`. The second approach dereferences just the resources in the final proportionate sample.

In Table 7.7 we compare the time taken and quality values of both approaches. We can observe that the stratified sampler fares better with regard to time for the Southampton ECS E-Prints and the Sweto DBLP datasets, no difference in the WordNet dataset, whilst the reservoir sampler was slightly better for the XBRL dataset. We can also observe that there was no difference between the two approaches in the WordNet dataset. This is because the dataset had only one PLD (`www.w3.org`) used in all resources, therefore there is no advantage of using the second approach of the first one in such cases. In this case, the second method will yield comparable results to the first approach. However, the host (W3C) was blacklisting our IP address during the assessment after trying to dereference a large number of URIs pertaining to the `www.w3.org` PLD.

In the Sweto DBLP dataset we notice that the stratified sampler approach gave a quality value of 0 in all cases, whilst for the reservoir sampler we got values that are less than 0.009 (and one case 0.03). Upon further investigation, we found out that the dataset contained around 5044 unique PLDs extracted from the subject and object of the dataset's triples. In this regard, the global reservoir set parameters for

both approaches was very low, and therefore both approaches will miss the majority of PLDs. From the 5044 unique PLDs a large concentration of URIs had the same PLD, namely `dblp.uni-trier.de` - ≈ 8.7M URIs, and `www.informatik.uni-trier.de` - ≈ 2.8M URIs. The rest, ≈ 39K, is split between the other 5042. The third highest PLD, `www.springer.de`, had 5030 resources in a 100M triple dataset, followed by the `www.computer.org` with 1392 resources, and `www.acm.org` with 890 resources.

The reservoir sampler takes into consideration a sample of resources from all PLDs. Therefore, the increase of the global reservoir parameter to cover all PLDs (e.g. in P2 - 50 $\mapsto$ 5050) might change the overall quality value. On the other hand, this increase does not necessarily mean improvement on the quality value for the alternative approach.

For the second approach, ≈ 39K data objects are considered to be relatively small in relation to the objects pertaining to the two "largest" PLDs and the total data objects. Therefore, in this case, even if we consider a larger global reservoir of 5050 (to cover all unique PLDs in this dataset), the PLD reservoir sampler of 1000 still represents a relatively small portion of PLDs. With a sampler of 1000, only the two largest populations (i.e `dblp.uni-trier.de` and `www.informatik.uni-trier.de`) are represented. Therefore, enlarging the sampler would make the population sample more representative, which also means that the "larger" PLDs would have more resources represented together with some resources from smaller PLDs, thus maintaining the ratio of the population sample. With a reservoir sampler of 10000 we expect that resources from 5 PLDs are represented, with 50000 21 PLDs, with 100000 60 PLDs, etc .... The size of the reservoir sampler is the main trade-off of this approach, where a larger reservoir sampler means more accurate results at the expense of taking more time. However, when we increased the value for both parameters and re-run the experiments the quality value still resulted in 0. Upon further inspection, we manually checked a number (around 30) of resources for their dereferenceability. The top two PLDs had no resource that was dereferenceable (Listing 7.1 shows how manual dereferenceability is done using just a terminal). This may imply that around 99.65% of the resources in this dataset might be non-dereferenceable. However, this can only be confirmed by assessing the dataset using the non-approximate metric.

```
$ curl -H "Accept: application/rdf+xml" -I -L "http://www.informatik.uni-trier.de/~
    ley/db/books/collections/kim95.html#DittrichD95"

HTTP/1.1 301 Moved Permanently
Date: Wed, 10 Aug 2016 08:28:08 GMT
Server: Apache
Location: http://dblp.uni-trier.de/db/books/collections/kim95.html
Content-Type: text/html; charset=iso-8859-1

HTTP/1.1 200 OK
Date: Wed, 10 Aug 2016 08:25:12 GMT
Server: Apache/2.4.7 (Ubuntu)
Set-Cookie: dblp-view=y; path=/; expires=Fri, 09 Sep 2016 10:25:12 +0200
Vary: Accept-Encoding
Cache-Control: max-age=28800
Expires: Wed, 10 Aug 2016 16:25:12 GMT
Content-Type: text/html; charset=utf-8
```

Listing 7.1: A manual cURL request and response to check if a resource is dereferenceable.

To conclude, with the stratified sampler, we can reduce the runtime even further than with the reservoir sampler, whilst still getting an acceptable (in some cases even better) estimate values. We observed that this approach might suffer when a dataset has a large set of unique PLDs and a large number of URIs

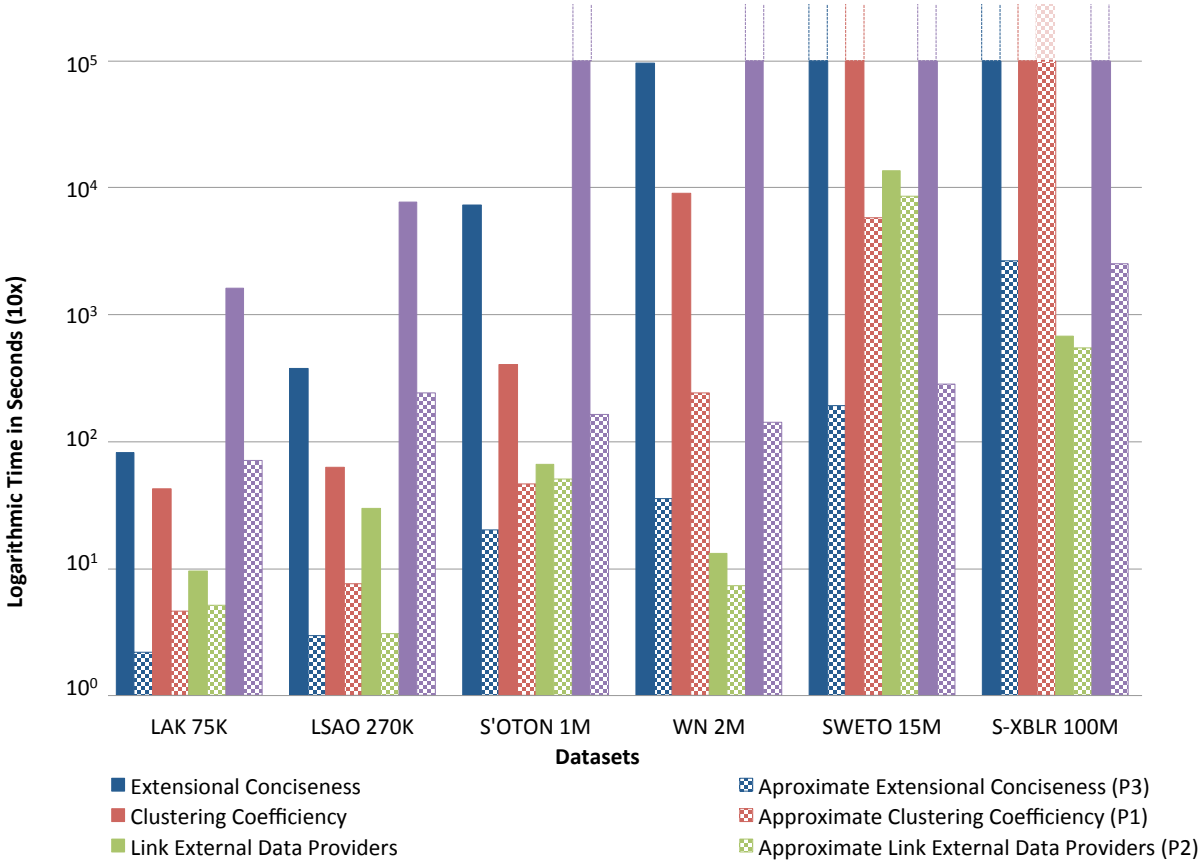

Figure 7.2: Runtime of metrics vs. datasets.

in a small set of these PLDs (as in the case of the SWETO DBLP dataset). In such cases, a larger PLD sampler (i.e. the unique sampler holding URIs with the same authoritative domain name) is required, in order to have a more accurate proportionate allocation.

7.3.3 Evaluation Discussion

Our experiments gave promising results towards the use and acceptance of probabilistic approximation for estimating the quality of linked open datasets. Figure 7.2 shows the time taken in all implemented metrics (actual and approximated) against the evaluated datasets. The graph clearly shows that all approximate metrics have a lower runtime than their equivalent actual metric. Whilst the approximate metrics for *link external data providers*, *dereferenceability* (approach two), and *extensional conciseness* computed for all datasets, the approximate *clustering coefficient* computed five datasets within a reasonable time. The actual *link external data providers* also computed all datasets within a reasonable time. The actual *dereferenceability* metric managed only to compute two datasets, while the rest computed up to the WordNet dataset. Table 7.8 shows the metric (actual and estimated) values for the datasets.

The approximate results are in most cases very close to the actual results. However, approximate measures are calculated in an acceptable time unlike their actual counterparts. As part of a larger effort to implement scalable LOD quality assessment metrics, we assessed the metrics identified in [160] and assigned to them possible approximation techniques discussed in chapter (cf. Table 7.9).

Overall, given that the results were obtained on yet small datasets, this chapter contributes towards

invaluable results that can be the basis for further studies. These results show that with probabilistic approximation techniques:

1. Runtime decreases considerably – for larger datasets easily by more than an order of magnitude;
2. Loss of precision is acceptable in most cases with less than 10% deviation from actual values;
3. Large linked datasets can be assessed for quality even within very limited computational capabilities, such as a personal notebook.

7.4 Concluding Remarks

Ensuring a scalable and time acceptable assessment of linked big datasets is important for the ever-growing Web of Data. Data consumers are usually satisfied with near-to-accurate data quality values, rather than the considerably more time consuming process that provides the exact results. In this chapter, we have demonstrated how three probabilistic techniques sampling, Bloom Filters and clustering coefficient estimation can be successfully applied for Linked Data quality assessment. Furthermore, we extend our previous work [49] by investigating the *stratified sampling* technique.

Using both *sampling techniques*, our idea was to sample the assessed dataset in order to reduce time-consuming processes. More specifically, we applied these techniques to those metrics that require network operations to complete. Such operations can be expensive, mostly depending on the bandwidth and data source latency. With *Bloom Filters* we aimed at finding a time efficient solution to detect possible duplicate resources in the assessed dataset. Using bit vectors stored in memory, duplicates can be detected in streams of data. Finally, the *clustering coefficient estimation* was applied on a graph representation of

	LAK 75K	**LSOA 270K**	**S'OTON 1M**	**WN 2M**	**SWETO 15M**	**S-XBRL 100M**
Extensional Conciseness	0.9948	1	0.7375	0.948	0,000370	N/A
Approx. Extensional Conciseness	0.9945	1	0.6601	0.8447	0.9998	0.1097
Clustering Coefficiency	0.9610	1	0.9335	0.7592	N/A	N/A
Approx. Clustering Coefficiency	0.9782	0.9999	0.9930	0.8104	1	0
Link External Data Providers	3	0	2	0	0	1
Approx. Link External Data Prov.	3	0	2	0	0	1
Dereferencibility	0.99485	0.744414	N/A	N/A	N/A	N/A
Approx. Dereferencibility (A2)	0.99600	0.73	0.02597	0.916	0.0	0.034

Table 7.8: Metric value (actual and approximate) per dataset.

Category	**Dimension**	**Metric**	**Approximation Technique**
Accessibility	Availability	Dereferenceability of the URI	Sampling
		Dereferenced Forward-Links	Sampling
	Interlinking	Detection of Good Quality Interlinks	Random Walk
		Dereferenced Back-Links	Sampling
	Performance	Usage of Slash-URIs	Sampling
Intrinsic	Syntactic Validity	Syntactically Accurate Values	Sampling
	Conciseness	High Extensional Conciseness	Bloom Filters
		High Intensional Conciseness	Bloom Filters
		Duplicate Instance	Bloom Filters
Contextual	Relevancy	Relevant Terms Within Meta-Information Attributes	Page Rank
		Coverage	Sampling

Table 7.9: Possible metric approximation implementation.

the assessed dataset in order to measure the neighbourhood density of a resource. However, this graph measure still needs refinement and in general the logical next step would be to create a sample of the graph before applying network measures.

Our comprehensive experiments have shown that we can reduce runtime in most cases by more than an order of magnitude, while keeping the precision of results reasonable for most practical applications. All in all, we have demonstrated that using these approximation techniques enables data publishers to assess their datasets in a convenient and efficient manner without the need of having a large infrastructure for computing quality metrics. Therefore, probabilistic techniques can make quality assessment scalable for large linked datasets (cf. RQ 3 in Section 1.3). The contribution in this chapter opens up further quests to apply other probabilistic techniques to important but otherwise intractable quality metrics.

CHAPTER 8

A Preliminary Investigation Towards Improving Linked Data Quality using Outlier Detection

As linked datasets usually originate from various structured (e.g. relational databases), semi-structured (e.g. Wikipedia) or unstructured sources (e.g. plain text), a complete and accurate *semantic lifting* process is difficult to attain. Such processes often contribute to incomplete, misrepresented and noisy data, especially for semi-structured and unstructured sources. Issues caused by these processes can be attributed to the fact that either the knowledge worker is not aware of the various implications of a schema (e.g. incorrectly using inverse functional properties), or because the schema is not well defined (e.g. having an open domain and range for a property). In this chapter we are concerned with the latter cause, aiming to identify potentially incorrect statements in order to improve the quality of the knowledge base.

When analysing the schema of the DBpedia knowledge base, we found out that from around 61,000 properties approximately 59,000 had an undefined domain and range. This means that the type of resources attached to such properties as the subject or the object of an RDF triple can be very generic, i.e. `owl:Thing`. For example, the property `dbp:author`, whose domain and range are undefined, has instances where the subject is of type `dbo:Book` and the object of type `dbo:Writer`, and other instances where the subject is of type `dbo:Software` and the object of type `dbo:ArtificialSatellite` (e.g. dbpedia:Cubesat_Space_Protocol dbp:author dbpedia:AAUSAT3). Without looking at a schema one would intuitively expect that a *Book* has a *Person* as its *author*, but if the *author* property is under-specified, that is without a domain and range specified in the schema, its semantics cannot be fully understood. Thus, lifting such domain and range restrictions on properties can undeniably increase the possibility of having incorrect RDF statements. Such incorrect statements decrease the quality of the dataset and as a result the precision of results when querying, in turn affecting *data consumers* who use the knowledge base, including service providers as well as end users.

In this chapter we explore the possibility of using distance-based outlier detection, in an attempt to automate the detection of incorrect RDF triples in a data source. In this preliminary investigation we do not consider object literals. Furthermore, we show how this approach can be used as a Linked Data quality metric.

Overall the particular contributions of this chapter are:

- the definition of an outlier detection for Linked Data method aiming at effectively (precision) and efficiently (scalability) detecting statements that are potentially incorrect w.r.t. domain and range;
- the preliminary analysis of the method with regard to various similarity measures, parameters as

well as regarding its time and space complexity.

Furthermore, we look into how this approach can provide interested stakeholders with an approximative quality value that detects the number of outliers in a dataset. We implement this metric for the Luzzu framework (cf. Chapter 4).

Our proposed approach is explained in Section 8.1, together with the analysis of its space and time complexity. Experiments and evaluations of our approach are documented in Section 8.2.

8.1 Improving Dataset Quality by Detecting Incorrect Statements

The detection and subsequent cleaning of potentially incorrect RDF statements aids in improving the quality of a linked dataset. There were a number of attempts to solve this problem in the best possible manner (cf. Section 3.5). More recently, Paulheim and Bizer presented an approach whereby a statistical distribution measure was used to validate the correctness of non-literal RDF statements [127]. Similarly, our approach focuses on detecting incorrect statements where both the subject and object are resources. We apply the distance-based outlier technique by Knorr et al. [97] in a Linked Data scenario. Exploiting reservoir sampling and semantic similarity measures, clusters of RDF statements based on the statements' subject and object types, are created, thus identifying the potentially incorrect statements. Furthermore, the 2D space clusters RDF statements that have the same property, hence for different predicates in the same dataset must have their own 2D space. As opposed to [127], our approach focuses on finding incorrect statements using outlier detection rather than using a statistical frequency (distribution) of types.

8.1.1 Approach

Following [97], our proposed Linked Data adapted method has three stages: *initial*, *mapping*, and *colouring*. These three stages automate the whole process of finding potentially incorrect statements for a certain property. In the *initial* stage, k (the size of the reservoir sampler) RDF statements are added to a reservoir sampler. Following the initialisations of the constants, the *mapping* stage groups data objects (in our case RDF statements) in various cells based on the properties described in Section 2.4.1. Finally, the *colouring* stage identifies the cells that contain outlier data objects.

Initial Stage

The initial steps (Algorithm 1) are crucial for achieving a better identification of potentially incorrect statements. We start by determining the approximate distance D that is used in the second stage to condition the mapping, and thus the final clustering of RDF statements. The approximate value D is valid for a particular property, i.e. the property of the triples (data objects) being assessed. Therefore, two properties (e.g. *dbp:author* and *dbp:saint*, i.e. the patron saint of, e.g., a town) will have different values of D according to the triples, their types, and ultimately the similarity measure chosen. Currently, in our approach we assume that a resource is typed with **only** one class, choosing the most specific type if a resource is multi-typed (e.g. *dbo:Writer* and not *dbo:Person*). Additionally, a threshold fraction p (between 0 and 1) is defined by the user during the initial phase, affecting the number of data objects in a cluster M. Therefore, p can be considered to be a sensitivity function that increases or decreases the amount of data objects in a cluster.

Determining the Approximate Distance Our approach makes use of reservoir sampling as we described in Section 7.2.1 (approach one). The rationale is that D is approximated by a sample of the data

objects being assessed, to identify the acceptable maximum distance between objects mapped together in a cell, in a quick and automated way. To determine the approximate distance we applied two different implementations (cf. Section 8.2.3 for their evaluation), one based on a simple sampling of triples and another one based on a modified reservoir sampler, which we call the *type-selective*. From the sample set (for both implementations), a random data object is chosen to be the *compare to* object, and is removed from the sampler. All remaining statements in the sampler are semantically compared with this randomly chosen data object individually, and their distance values are stored in a list. The median distance is then chosen from the list of distances. We chose the median value over the mean value as a central tendency since the latter can be influenced by outliers[1].

In the first implementation (simple sampling), the reservoir selects a sample of triples, irrelevantly of their subject and object types. The main limitation is that, irrelevant of the size of the reservoir, the approximate distance value D can bias towards the more frequent pairs of the subject and object types. Therefore, the sampler might not represent the diversity of types attached to the particular property being assessed. Furthermore, in [97] the authors explain that whilst sampling can be used to determine a starting D, the said technique might not provide an accurate D, due to "infrequent and unpredictable occurrence of outliers". We refer to this as the *sampler representation problem*. For instance, the *dbp:author* property has around 47% of its statements (excluding statements with literal objects) with a subject type of *dbo:Book* and an object type of *dbo:Writer*. Moreover, around 24% of the triples have a subject type of *dbo:Book* and an object type of *dbo:Person*. This leaves around 28% of the triples with 180 different pairs of subject and object types, which include subclasses of *dbo:Book*, *dbo:Person*, and *dbo:Writer*. Therefore, the reservoir in this implementation might have a large amount of type-matched statements (i.e. having matching subject type and object type), thus give a low approximate D value since there is no guarantee that all subject and object types are represented in the sampler.

In order to attempt to solve the *sampler representation problem*, we propose the *type-selective* reservoir sampler. The proposed reservoir sampler modifies the simple sampler by adding a condition that only one statement with a certain subject type and object type can be added to the reservoir. In other words, when there are two distinct statements with matching subject types and object types, only one of these statements will be added to the reservoir.

With respect to the two sampler approaches, we draw up these two hypotheses, which are validated in Section 8.2.

Hypothesis 1: *The simple reservoir sampler will give a lower approximate distance value D than the proposed type-selective reservoir sampler due to the sampler representation problem.*

Hypothesis 2: *Using the type-selective reservoir sampler, the precision of detecting incorrect triples will improve considerably when compared to the simple reservoir sampler.*

[1] http://www.quickmba.com/stats/centralten/. Date Accessed 29th September 2016

Algorithm 1 Initialisation Stage

```
vars: SampleDataObjects, DataObjects, cells[][], hostTriple, hostLocation
function INITIALISE:
    for triple in Dataset do
        SampleDataObjects.addOrSkip(triple);
    approxD ← []
    for triple_i in SampleDataObjects do
        vals ← []
        for triple_j in SampleDataObjects do
            vals.add(SIMCOMP(triple_i, triple_j))
        approxD.add(median(vals))
    D ← median(vals)
    hostTriple ← RANDOMTRIPLE(SampleDataObjects)
    hostLocation_x ← RANDOM(size(cells))
    hostLocation_y ← RANDOM(size(cells))
    cellLength = D / (2√2)
```

Mapping Stage

The mapping stage attends to the clustering of data objects in cells, based on the properties described in Section 2.4.1. An RDF statement is chosen at random from the whole set of data objects and is placed in a random cell. The is is called the *host* cell. Thereafter, every other RDF statement in the dataset is mapped to an appropriate cell by first comparing it to the data object in this host cell. We improve performance by using hash maps and sets to index:

1. the cell co-ordinates of a pair of subject and object types (identified with the variable $Map_{<k,(l_x,l_y)>}$ in Algorithm 2, where p is the index key identified by the pair and l denotes the allocated cell co-ordinates);
2. the semantic distance of two types; and
3. the non-empty cell locations.

The mapping process is described in Algorithm 2.

Semantic Similarity Measure In order to check if an RDF statement fits in a cell with other similar RDF statements, a semantic similarity measure (function *simComp* in Algorithm 1) is used. More specifically, since we are mostly concerned about the distance between two statements, we use a normalised semantic similarity measure. The similarity (ρ) between two statements S_1 and S_2 is defined as the average of the similarity between the statements' subjects, and the similarity between the statements' objects:

$$\rho(S_1, S_2) = \left(\frac{\text{sim}(S_{1sbj}, S_{2sbj}) + \text{sim}(S_{1obj}, S_{2obj})}{2} \right) \tag{8.1}$$

This average-based definition of ρ was chosen as it represents the statistical centrality of the two similarity values, i.e. the similarity between the two subject types and the similarity between the two object types.

Once the similarity of the two statements has been calculated, the normalised semantic distance [64] is calculated as follows:

$$d_{sem} = 1 - \rho(S_1, S_2) \tag{8.2}$$

Our approach is flexible towards the choice of the semantic similarity measure. Currently we reuse intrinsic measures (Zhou information content (IC) model [162] with the Mazandu measure [113]) available in the semantic measures library and toolkit by Harispe et al. [65], but users can easily implement their own similarity measures.

Colouring Stage

After mapping all data objects to the two-dimensional space, the *colouring* process colours cells to identify outlier data objects. In [97], the minimum number of objects (M) required in a cell such that data objects are not considered as outliers is calculated as:

$$M = N \cdot (1 - p) \tag{8.3}$$

where N is the total number of data objects, and p is the threshold fraction value determined in the *initial* stage. The authors also define the following conditions, which we adopt in our approach:

- if a cell has $> M$ data objects, then (a) the cell is coloured **red** and its encircling cells are coloured **pink**;
- if two adjacent cells have a total of $> M$ data objects together, but $< M + 1$ individually, then the two cells are coloured **pink**

Data objects in **red** and **pink** cells are not considered as outliers.

8.1.2 Time and Space Complexity Analysis

One of the main goals of this approach is to keep time and space complexity as low as possible. With regard to time complexity, we aim to achieve a *polynomial* worst-case time complexity. On the other hand, we aim that our data structures can be easily kept in memory.

Analysing the Data Structures used

To make sure that we keep the running time low, we make use of hash data structures (maps and sets) where the time complexity for adding and searching items is $O(1)$, assuming there are no rare collisions. The space complexity for storing hash data structures is $O(n)$. In our approach, the largest hash data structure (out of three in total) is the count of unique subject-object pairs of an assessed data property. The two-dimensional space, i.e. the cells (2D array) where data objects will be mapped, has a **worst case** space complexity of $O(k^2 \times j)$, where k is the number of cells in the square array and j is the number of data objects mapped inside each cell. In practice, our algorithm is not using all the space as cells are only initialised if required when an object is mapped into a cell location for the first time. In this preliminary approach the choice of the array size k was arbitrary, with no scientific correlation to the size of the datasets. The drawback of choosing an arbitrary value is that if the choice is very small, then the data objects might not all fit in the 2D plane, and thus this approach might not work. In this approach, for the number of triples n we assumed the following:

- $n \leq 100 \mapsto n$
- $100 < n \leq 10000 \mapsto n/10$ *capped at 100*
- $10000 < n \leq 100000 \mapsto n/100$ *capped at 1000*
- $n > 100001 \mapsto n/1000$

Algorithm 2 Mapping Data Objects in a two-dimensional Space - Adapted [97] for Linked Data.

vars: *SampleDataObjects*,*DataObjects*, $Map_{<k,(l_x,l_y)>}$, *cells*[][], *hostTriple*, *hostLocation*, l_1, l_2
function MAPCELLS(*triple*)
  $cellLocation \leftarrow (-1,-1)$
  $tripleSubjectType \leftarrow$ TYPEOF($triple_{subject}$)
  $tripleObjectType \leftarrow$ TYPEOF($triple_{object}$)
  if ($Map_{<k,(l_x,l_y)>}$.get(<*tripleSubjectType*, *tripleObjectType* >) $\neq$ null) **then**
    $cellLocation \leftarrow Map_{<k,(l_x,l_y)>}$.get(<*tripleSubjectType*, *tripleObjectType*>)
  else
    $distance \leftarrow 1 -$ SIMCOMP(*hostTriple*,*triple*)
    if ($distance \leq D/2$) **then**
      $cellLocation \leftarrow hostLocation$
    else if ($distance \leq D$) **then**
      $l_1 \leftarrow$ Encircling Cells of *hostLocation*
      $l_2 \leftarrow$ Buffer Cells of *hostLocation* $\setminus l_1$
      $p_{alloc} \leftarrow l_1 \cup l_2$
      $cellLocation \leftarrow$ ALLOCATECELL(p_{alloc},distance)
    else ▷ $distance > D$
      $l_1 \leftarrow$ Encircling Cells of *hostLocation*
      $l_2 \leftarrow$ Buffer Cells of *hostLocation* $\setminus l_1$
      $l_3 \leftarrow cells \setminus l_2$
      $p_{alloc} \leftarrow l_3$
      $cellLocation \leftarrow$ ALLOCATECELL(p_{alloc},distance)
    $Map_{<k,(l_x,l_y)>}$.put(<*tripleSubject*, *tripleObject*>, *cellLocation*)
  return *cellLocation*

Algorithm 3 Function for Allocating Cells in Knorr et al. [97]

function ALLOCATECELL(p_{alloc}, *distance*)
  if ($distance > cellLength$)) **then**
    return a diagonal cell location from p_{alloc}
  else
    return a horizontal/vertical cell location from p_{alloc}

Since we are indexing (using a hash set) the occupied cells (initialised array cells with data objects), the time complexity for accessing all occupied cells in the two-dimensional space is $O(n)$, where n is the size of the occupied cells hash set.

Analysing the Time Complexity for the Three Main Stages

After analysing the space and time complexity for the data structures used, we analysed each of the three stages.

Initial Stage Let R_s be the total available capacity of the reservoir sampler. The *initial* stage takes at worst $O(R_s^2)$, as elements in the reservoir sampler are compared with each other. Using our *type-selective* reservoir, the time taken to calculate the approximate value D is at worst the square of the count of unique subject-object pairs of an assessed data property. The space complexity of the *type-selective* reservoir

is $O(R_s + p)$, where p is the size of a hash data structure indexing the subject-object pairs of the added triples.

Mapping Stage Knorr et al. state that the mapping time complexity is $O(N)$, where N is the number of data objects. [97]. Our approach ensures that all mapped data objects observe the three axioms defined in Section 2.4.1. Mapping a *single* data object takes (**best case**) $O(1)$, if all cells within the possible allocation set (i.e. either *encircling cells* or *buffer cells* in case of axiom 2, or $Layer_3$ in case of axiom 3) are free, or if there exists a previous <subject,object> pair that had been assigned a cell. In cases when not all cells are free, we check if the data objects in the occupied cells comply with the axioms and thus reducing the number of possible cells the data object in question can occupy. In the **worst case** this takes $O(N)$, as all data objects might be in the cells that can be possibly occupied by this *single* data object.

Colouring Stage The colouring stage of our approach is the same as in [97], thus the time complexity is the same as defined in the literature.

8.2 Experiments and Evaluations

Having implemented our approach, we performed three experiments focusing on complementary aspects. The first experiment that we conducted was to investigate the effect on the precision and recall of using different similarity measures. For this experiment, the algorithm was executed a number of times with different similarity configurations on a property dump (cf. Section 8.2.1). The second experiment aims to evaluate the precision and recall in different parameter settings, i.e. different approximate D and different fraction p. Finally, we evaluate the algorithm to determine the number of automatic assessment and cleaning iterations required to achieve a steady approximate quality value, i.e. a state in which further iterations will not significantly improve the quality value.

8.2.1 Experiment Setup

For these experiments[2] we extracted a subset of DBpedia to generate *property dumps*. The SPARQL query in Listing 8.1 illustrates how property dumps are generated. The placeholder *%%property%%* should be replaced by some concrete property (e.g. *author*). A property dump is an N-Triples file with all triples related to a particular property (e.g. *dbp:author*), and the respective subject and object type for each triple. Having this property dump, all type statements were extracted to represent all statements in the dump itself. For example, the statement `dbpedia:Franziska_Linkerhand dbp:author dbpedia:Brigitte_Reimann` is represented as the `dbo:Book dbp:author dbo:Writer` *pseudo triple*. In simple terms, `dbpedia:Franziska_Linkerhand` is the title of a novel, written by the author represented by the resource `dbpedia:Brigitte_Reimann`. Although each resource might have had multiple types assigned to it, we took the most specific type that was dereferenceable. For example, the resource `dbpedia:Brigitte_Reimann` has 32 different types, ranging from `owl:Thing` to `yago:Winner110782791`. If on the other hand, the resource's type is unknown or cannot be dereferenced, we automatically assign `owl:Thing` as its type. Types are required in our approach in order to be able to (1) find an approximate D value; and (2) classify statements on the 2D space.

From the pseudo triples, we manually tagged which of these type pairs (by "type pair" we mean the subject and object types of an RDF statement, e.g., `dbo:Book` and `dbo:Writer`) are correct

[2] All experiments and generated dumps can be found at http://eis-bonn.github.io/Luzzu/www_eval.zip.

or incorrect for the corresponding property (in this case dbp:author). For our experiments we generated and tagged two property dumps, one for the property dbp:author and the other one for dbp:publisher. We identified 89 possibly incorrect subject-object pairs (37 had a correct object type – i.e. the object was a type or subclass type of dbo:Writer, 40 a correct subject type, and 12 had incorrect subject and object types) and 92 correct pairs for the dbp:author type. The tagging criterion was that a dbp:author should have a subject type of dbo:Work and an object type of dbo:Person, or any of its subclass types. With regard to the dbp:publisher property dump, the criterion was that for a statement to be tagged correct, the subject has to be of type dbo:Work and the object has to be either of type dbo:Company or dbo:Organisation, or any of its subclass types. After manually tagging the pseudo triples, we identified 99 possibly incorrect subject-object pairs (17 had a correct object type, 74 a correct subject type, and 8 had incorrect subject and object types) and 54 correct pairs. These manually tagged pseudo triples were then iterated and used within SPARQL ASK queries against the possibly incorrect statements, in order to calculate the precision and recall of our approach. For example, using a SPARQL ASK we checked if the subject and the object type of the possible incorrect statement identified by our approach is also manually tagged as incorrect.

```
SELECT * WHERE {
  ?instance  dbp:%%property%%  ?variable ;
    a ?type .
  ?type a owl:Class .
  FILTER NOT EXISTS {
    ?subtype  ^a ?instance ;
    rdfs:subClassOf  ?type .
    FILTER (?subtype != ?type) }
  FILTER (regex(str(?type),'^http://dbpedia.org'))
  OPTIONAL {
    ?x      a  ?type2 .
    ?type2  a  owl:Class .
    FILTER NOT EXISTS {
      ?subtype2  ^a ?x ;
        rdfs:subClassOf ?type2 .
      FILTER (?subtype2 != ?type2) }
    FILTER (regex(str(?type2),'^http://dbpedia.org')) }
  FILTER (isURI(?x)) }
ORDER BY ASC(?instance)
OFFSET %%offset%%
LIMIT 10000
```

Listing 8.1: Generating Property Dumps

8.2.2 Experimenting with Different Similarity Measures

The purpose of this experiment is to see how different similarity measure configurations affect the precision and the recall of detecting incorrect statements. In our approach we make use of the Semantic Measures Library & Toolkit[3] [63], and for this experiment we identified three different *intrinsic content* (IC) configurations (cf. Section 2.4.2) and four different *similarity measures*. These were chosen **arbitrarily**, as the primary aim of this experiment is not to find the best similarity technique for this approach, as different knowledge bases might require different similarity measures. For this experiment we used the authors property dump (10,192 triples) and set the initial threshold fraction value p to 0.992. The chosen threshold fraction value corresponds to cluster groups having a minimum of 82.536 (rounded up

[3] http://www.semantic-measures-library.org

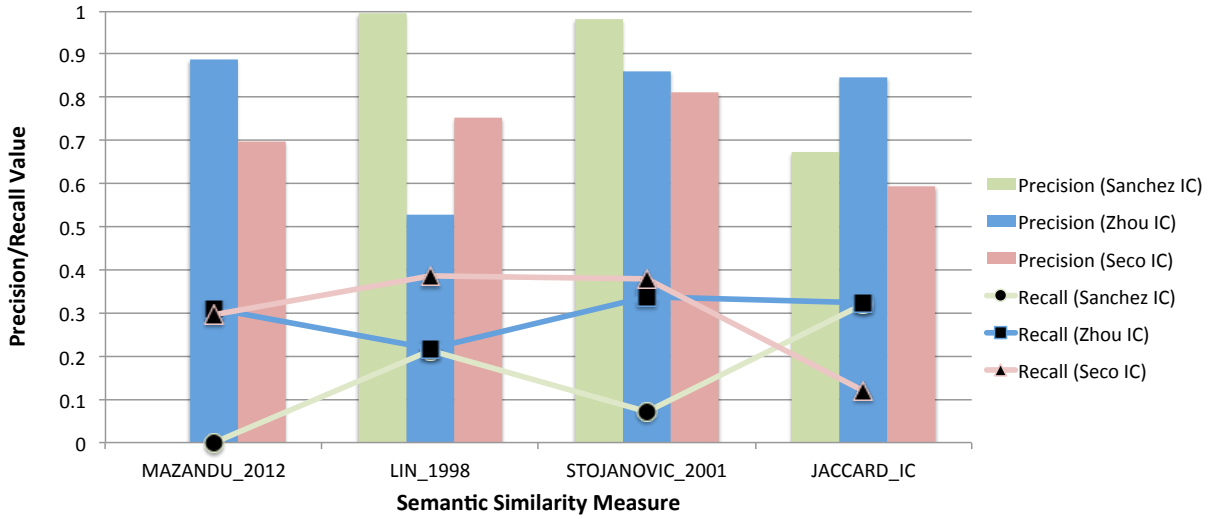

Figure 8.1: The precision and recall for different semantic similarity measures and configurations with a p value of 0.992.

to 83) data objects in order not to be considered as outliers. The approximate distance D was computed by the *type-selective* reservoir sampler.

Figure 8.1 shows the results of the precision and recall for this first experiment. The configuration that combined the Sanchez IC [140] and the Mazandu [113] measure was giving an infinity error. One possible cause for this infinity error is that the two measures cannot be combined together, however going into the details of these algorithms is out of the scope of this thesis. Therefore, it is not considered for these results. It can be observed that although a number of triples will be missed due to a low recall value, with regard to precision our approach performs well for different similarity measures, with the best similarity configuration giving us a precision value of almost 100%. Having a high precision ensures that there is a high probability that the retrieved outliers are really incorrect RDF statements. On the other hand, the low recall means that our approach will miss other possibly incorrect statements.

8.2.3 Evaluating the Proposed Approach's Precision with Different Parameters

After analysing the precision and recall for different similarity measures, we evaluate the precision and recall of our approach against different parameters. The primary aim of this experiment is to compare if the automatic approach of setting an approximate D value gives an advantage over the manual setting. Furthermore, in this experiment we aim to validate the hypotheses (Hyp 1 and Hyp 2). All experiments in this part of the evaluation used the same similarity measure configuration, i.e. Zhou IC [162] with the Mazandu measure [113], as implemented in the Semantic Measures Library & Toolkit [63]. This configuration was chosen as it gave the highest precision, with a relatively good recall when compared to other measures.

This experiment is split into two sub-experiments. In the first part, the two property dumps (authors and publishers) are evaluated using our approach to determine the precision and recall values. This experiment is conducted over various manually set values for the threshold fraction p and approximate distance D. The second sub-experiment evaluated the precision and recall of our proposed approach, again using the two property dumps, over a number of set values for p (as set in the previous experiment) but this time the approximate distance D is determined by the two automated approaches. For both experiments, p was set to: 0.99, 0.992, 0.994, 0.996, and 0.998, consecutively. This means that the minimum amount

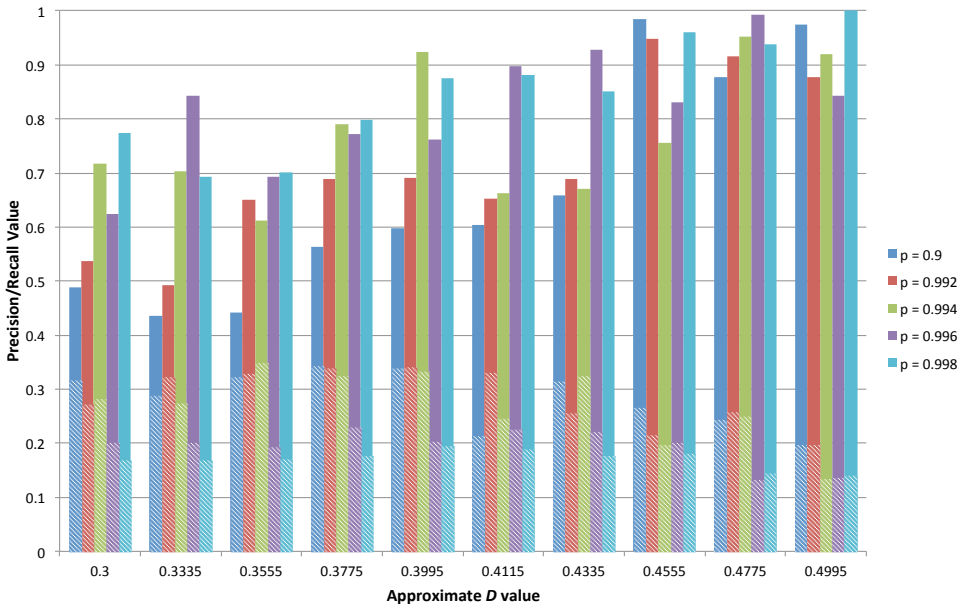

Figure 8.2: The precision and recall values for the authors property dump with different values for D and p. The solid bars denote precision values, whilst the striped overlapped bars denote recall.

(i.e. M) of data objects in each non-outlier cell is 102.92, 82.536, 62.152, 41.768, and 21.384 (for each different p respectively) for the authors property dump, and 105.16, 84.328, 63.496, 42.664, and 21.832 for the publishers property dump.

Sub-Experiment #1 – Setting Approximate D Manually

Both dumps had different sets of D values. These sets of values were obtained as rough estimates following a manual investigation of the triple types and a manual calculation of the similarity values between the different types.

From Figure 8.2, we observe that on average our approach achieved around 76% precision. On the other hand, the recall values were low, with an average of 31%. Low recall was expected as the threshold fraction p was closer to 1, therefore the a cell (and its surrounding cluster) was considered a non-outlier with a low number of data objects. We also observed that increasing the approximate value D does not result in an increasing precision. For example, in Figure 8.2 we spot that the precision value for the D value of 0.3335 is greater than that of 0.3555 when p was set to 0.996. We investigated this further and found out that although the latter value of D (0.3555) detected 39 more outliers, the number of true positives decreased (from 189 to 182) whilst the number of false positives increased by 42 data objects. This slight change in *true positives* and *false positives* was expected as the data objects cluster with similar data objects whose distance is the smallest. Therefore, the change in D might have moved some objects from one cell to another with the consequence that a previously non-outlier cell is now marked as an outlier, since a number of data objects might have moved to other cells. Another important factor that can affect the precision and recall values is the choice of the semantic similarity measure. Figure 8.3 represents the F1 score for the authors property dump manual experiment, showing an average of almost 0.43 for this harmonic mean score.

A similar trend of a higher precision than recall was also noted in Figure 8.4, where an acceptable average of 70% precision was achieved, whilst recall values were still considerably below the 50%

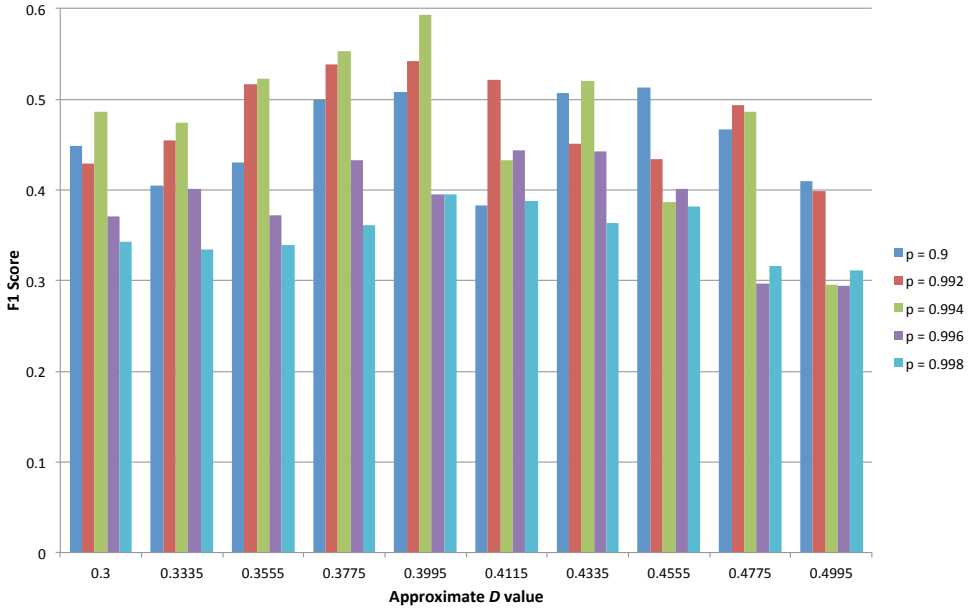

Figure 8.3: The F1 score for the authors property dump with different values for D and p.

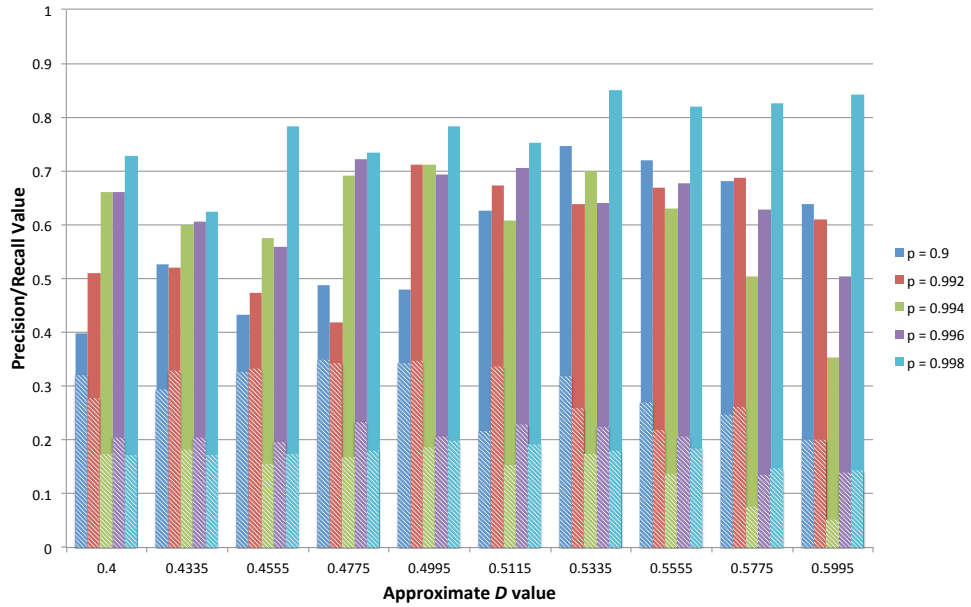

Figure 8.4: The precision and recall values for the publishers property dump with different values for D and p. The solid bars denote precision values, whilst the striped overlapped bars denote recall.

mark for the triples with a `dbp:publisher` property. We observed that the approximate D value with 0.4995 and a threshold p of 0.992 gave the best precision and recall. Similar to the previous experiment, these values are only applicable to the property being assessed, as a similarity distance has to be calculated between the subject and object types being used with the assessed property. The F1 score in this experiment is around 0.3 (cf. Figure 8.5). Overall, these results show that whilst our algorithm is retrieving many relevant data objects (i.e. a high number of *true positives*), there exist more, possibly incorrect triples in a dataset that need to be detected.

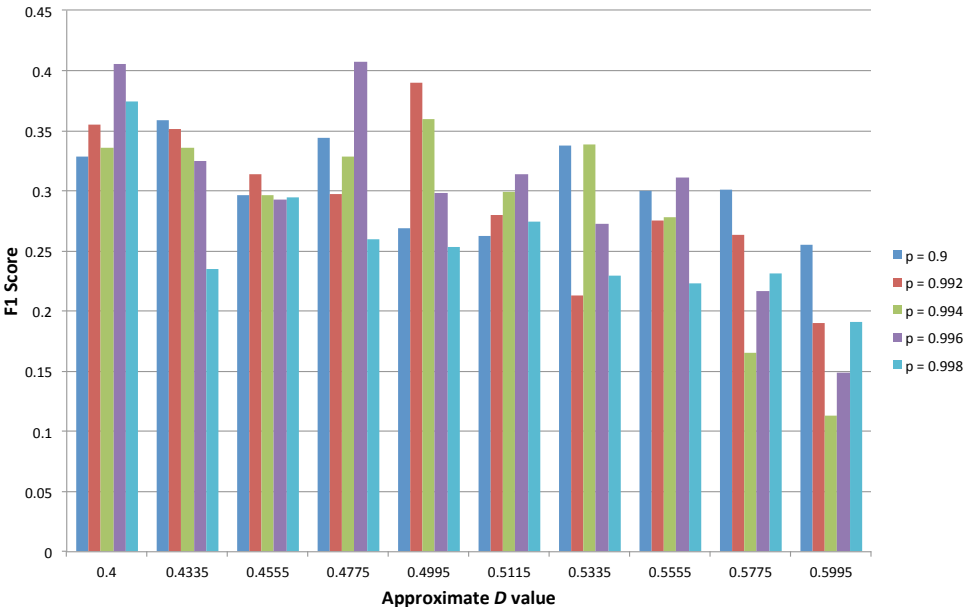

Figure 8.5: The F1 score for the publishers property dump with different values for D and p.

Sub-Experiment #2 – Setting Approximate D Automatically

The same dumps where used in this experiment, where an approximate D value was calculated first using the *simple* reservoir sampler and then using the *type-selective* reservoir sampler. A single host was chosen randomly from these reservoir samplers, together with a starting host location. The choice of a random data object will not affect the precision of the algorithm, as all data objects will be compared and mapped in suitable cells. From Figure 8.6 and Figure 8.7 we observe that the *type-selective* sampler outperforms its simpler counterpart for all p values with regard to the precision. One possible reason is due to the low approximate D values identified by the simple reservoir sampler. Low approximate D values mean that less data objects get mapped together in cells, since the approximate distance becomes smaller and data objects will be dispersed throughout the whole 2D space. This means that since less data objects are mapped in the same cell or surrounding cells, it would be more difficult to reach the $M + 1$ quota, and thus more cells will be marked as outliers. Therefore, whilst a low approximate D could lead for a decrease of *false positives* in non-outlier cells, it can also increase of *false negatives* (thus decreasing *true positives*), as objects that should not be marked as outliers could end up in outlier-marked cells. The main factors that affect the approximate D value are (1) the choice of the semantic similarity measure, and (2) the underlying schema (cf. limitations in Section 8.2.5). Furthermore, this approximate D value and the user-defined sensitivity threshold value (p) affect the precision and recall.

Following these experiments, in Figure 8.8 and Figure 8.9 we compared the *type-selective* precision and recall results for every p against the manual approach. For this comparison we used the manual scores that got the highest F1 measure for each p value, thus having a balance between the precision and recall. Figure 8.8 shows that the manual approach performed overall better than the automatic one in terms of the F1 measure. Nevertheless, in most cases, there are no large discrepancies between the two. Whilst the manual approach provided better recall on both property dumps, the automatic approach fared better with regard to the precision of the *publishers* property dump. In both cases, the automatic approach resulted into a higher approximate D value than the manual approach. For the authors use case, the automatic approach resulted into an approximation D value of 0.482147, which was approximately

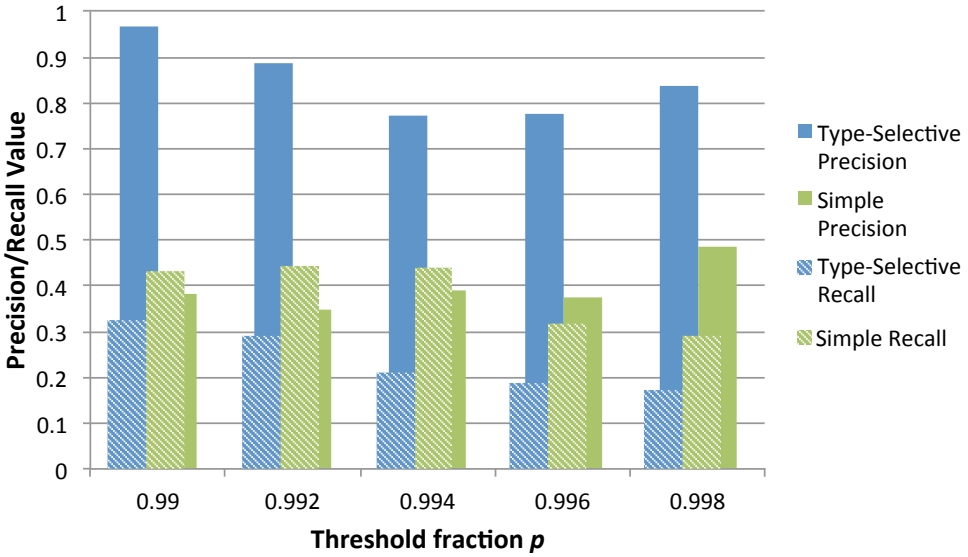

Figure 8.6: The precision and recall values for the authors property dump with different values for p and a generated D value.

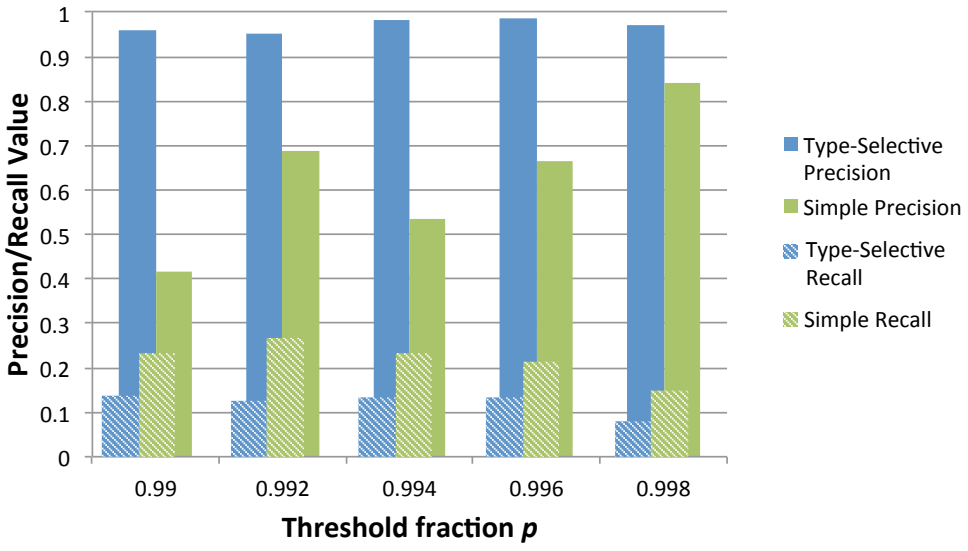

Figure 8.7: The precision and recall values for the publishers property dump with different values for p and a generated D value.

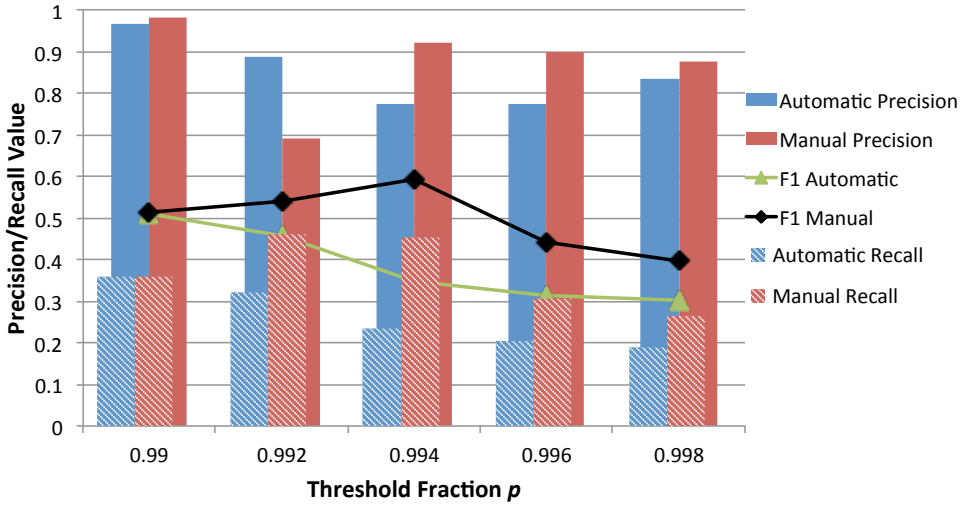

Figure 8.8: Precision and recall values for the authors property dump comparing the manual results against the automatic results for multiple values of the fraction p.

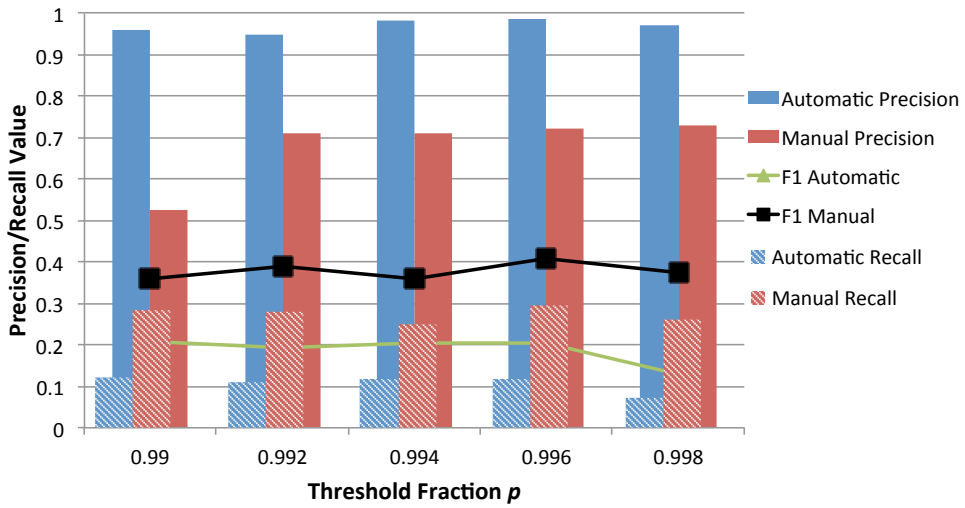

Figure 8.9: Precision and recall values for the publishers property dump comparing the manual results against the automatic results for multiple values of the fraction p.

18% more than the given manual approximation D value for the highest F1 value (i.e 0.3995 for threshold fraction p). Similarly, for the publishers use case, the approximation D value for the automatic approach was higher that that of the manual by approximately 16%.

8.2.4 Approximating Quality for Incorrect RDF Statements

The rationale of this experiment is to evaluate and see the approximate number of iterations required until the proposed outlier detection algorithm produces **no more** true positive outliers. This means that the evaluation will iterate until the *real quality value* is 1. Triples identified as outliers are removed from the assessed data dump after each iteration. Using the tagged pseudo triples we simulated the *real quality*

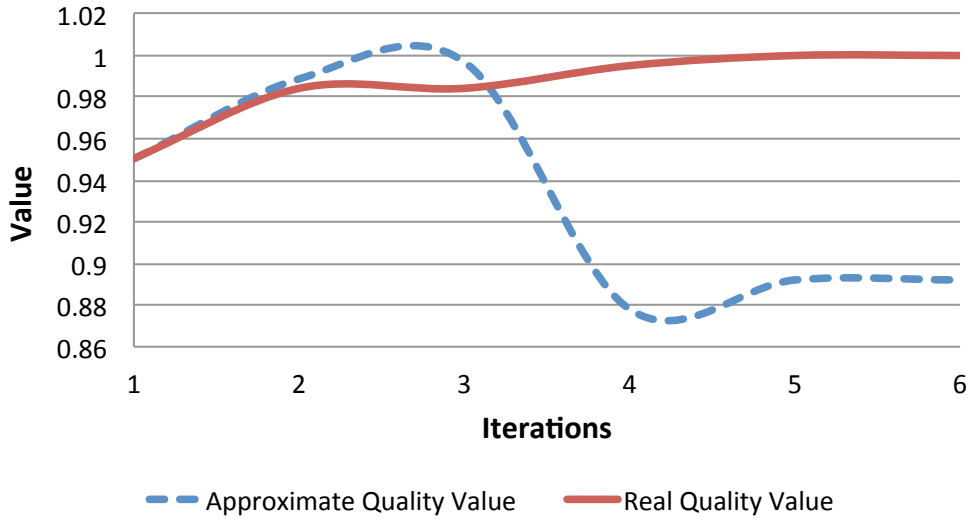

Figure 8.10: Iterations and Values (Approximate and Real) for the Quality Assessment of Incorrect RDF Triples – Authors.

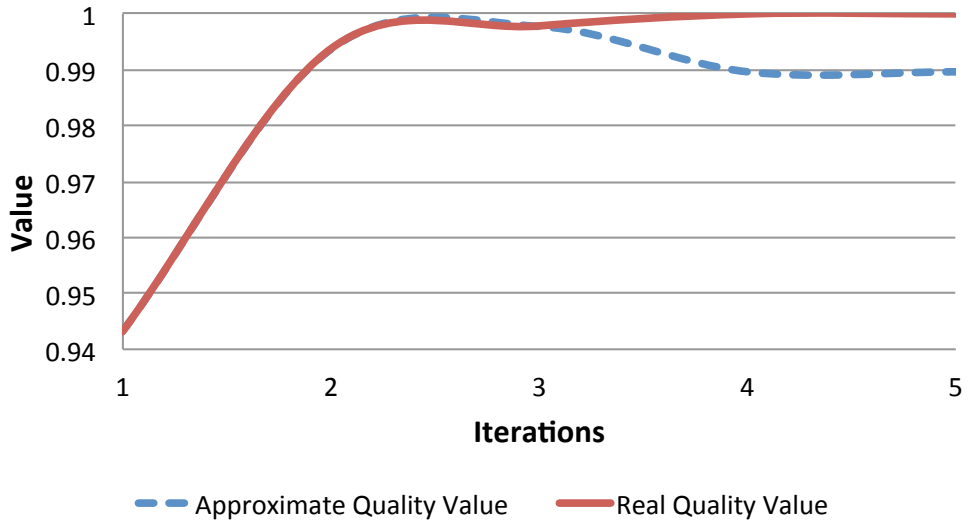

Figure 8.11: Iterations and Values (Approximate and Real) for the Quality Assessment of Incorrect RDF Triples – Publishers.

value. This is calculated as:

$$qv_{real} = 1 - \left(\frac{\#truePositives}{\#totalTriples} \right) \tag{8.4}$$

On the other hand, given that our algorithm works in an unsupervised environment, the *approximate quality value* should take into consideration all outliers detected by the algorithm. This value, which gives an approximate percentage of incorrect RDF triples, is calculated as follows:

$$qv_{approximate} = 1 - \left(\frac{\#allOutliers}{\#totalTriples} \right) \tag{8.5}$$

Figure 8.10 and Figure 8.11 depict two graphs with iterations and their corresponding quality values.

Both graphs show that in this case the algorithm needs between 5 and 6 iterations until a steady approximate quality value can be achieved. Even though the ideal value 1 was not achieved, we have confidence that the reported approximate values are sufficient, as *false positives* are unavoidable when using unsupervised clustering techniques.

We also noticed that in Figure 8.10 there was a remarkable difference between the third and the fourth iteration, which then propagated to the other two iterations. In the first two iterations, the algorithm identified 552 possible incorrect statements, which could have included a number of statements that were incorrectly detected. In the third iteration no statements were detected as outliers, and thus the threshold fraction value p was decreased. Following this change in the p value, more outliers (152) where detected in the fourth iteration, thus this change in threshold might have triggered more false positives.

8.2.5 Discussion

In this section we evaluated different aspects of our proposed approach. The led evaluation is as yet not conclusive, since we only evaluated our approach with two properties. Nevertheless, with this preliminary evaluation we can already show that the results regarding the precision are promising and comparable with the state-of-the-art, in particular [127]. Furthermore, our approach is not yet positioned against other state-of-the-art approaches. Positioning our work would not only lead to better conclusions, but also help us identifying the strong and weak points of our approach. Unfortunately, there is a lack of evidence (i.e. available implementation or comparison results like precision and recall) provided by related work that would enable us to compare it to our approach. Ideally, benchmarks and gold standards should be created that allow researchers to evaluate their tool/approach against it, similarly to what is done in the fields of information retrieval or question answering.

This evaluation also showed that our approach produces a low recall value and thus a low F1 measure. A higher recall, without compromising the precision, would have been ideal, as with low recall we are missing relevant data objects that should have been marked as outliers. In our evaluation we also validate our earlier assumptions (i.e. Hyp 1 and Hyp 2) related to the choice of the reservoir sampler. With regard to Hyp 1, the simple reservoir sampler value for the approximated D was on average 0.168, whilst for the *type-selective* sampler the approximated D was 0.482. The approximate values affect the precision results, as having a low approximate D means that similarly typed statements might not be mapped together in the same cell. The overall prevision improved by around around 40% for different p fractions when using the *type-selective* reservoir sampling technique, thus validating Hyp 2.

One must also note that the choice of a semantic similarity measure will also affect the precision and recall values of such an approach, in a way that its results are the deciding factor where a data object is mapped. Therefore, a chosen similarity measure should be appropriate with regard to the characteristics of the schemas underlying the assessed knowledge base.

Nevertheless, our approach has a number of known limitations:

1. The proposed approach does not fully exploit the semantics of typed annotations in linked datasets, since our approach assumes that an instance is a member of only one type (and its subclasses), in particular the most specific type assigned to the resource;

2. The evaluated semantic similarity measures are limited to hierarchical '*is-a*' relations, which might be more fitting to ontologies having deep hierarchies;

3. The sampled population (used to identify the approximate value D) might not reflect the actual diverse population of the data objects that have to be clustered in both sampler implementations.

Thus, with both implementations we will not achieve the best representative sample. One possible solution is to modify the *type-selective* algorithm to use a stratified sampling approach (cf. Section 2.3.1);

4. Whilst with the *simple* sampler outliers might occur in the sample population, with the *type-selective* sampler there is a 100% certainty that outlier data objects are present in the sample that determines the approximate D. Knorr et al. [97] had foreseen this problem and whilst suggesting that sampling provides a reasonable starting value for D, it cannot provide a high degree of confidence for D because of the unpredictable occurrence of outliers in the sample.

8.3 Concluding Remarks

Improving the quality of linked datasets has a significant impact on the Web of Data and its users, including producers as well as consumers. Having good quality datasets ensures their re-usability and thus helps in decreasing the number of duplicate and redundant resources on the Web. The semantic lifting process can produce a number of incorrect and inconsistent triples that affect the quality and thus re-usability of the Web of Data resources.

In this chapter we perform a preliminary study looking at the possibility of detecting potentially incorrect RDF statements in a dataset using a time and space efficient approach. More specifically, we applied a distance-based outlier detection technique [97] to identify outliers in a Linked Data scenario. In light of the challenge 1 defined in Section 1.1, in this chapter we propose a scalable approach to approximately assess the number of outliers in a dataset (classified under the semantic accuracy dimension in [160]). In this chapter we also provide another solution for the defined research question 3 (cf. Section 1.3). While providing preliminary results, our approach has a number of limitations, which we described in Section 8.2.5. The main advantage of our method is that the usage barrier is very low, since no initial supervision, configuration or training is required. Our method reports potential quality problems immediately and the high precision values make the manual review of potential quality problems relatively efficient. However, the recall we achieved still leaves room for further improvements in subsequent work.

Most existing work focuses on detecting potentially incorrect triples through, statistical analysis [127, 156], crowdsourcing [2, 153] or using background knowledge for schema enrichment [148, 161]. Hence, a particular innovation of our approach is to take into consideration the underlying schema of a knowledge base and the semantic topology of types in order to create clusters to identify outliers.

Part V

Large Scale Experiments and Conclusions

In the first three parts of this thesis we described a methodology and implemented a quality assessment framework (cf. Part II), defined a meta-model to describe the quality of a dataset (cf. Part III) and finally we discussed a number of techniques that can be used to ensure scalability and efficiency during assessment (cf. Part IV). In Chapter 9 we implemented a number of quality metrics (27 metrics across 4 different categories) based on the survey presented in [160], in order to perform a large empirical evaluation of linked datasets. The purpose of this evaluation is two-fold; (1) understand the quality of a number of datasets (130 datasets) present in the latest Linked Open Data Cloud, therefore tackling research question 4 (cf. Section 1.3), and (2) the overall application of Luzzu helps us to tackle the main research question of the thesis. Using the quality assessment results, we use the Principal Component Analysis (PCA) in order to identify the non-informative quality metrics. Furthermore, quality metadata following this evaluation is available online as Linked Data resources, ready for consumption.

We conclude this thesis in Chapter 10 by re-visiting and answering the research questions defined in Section 1.3. Finally, we look into the next steps towards Linked Data Quality assessment.

CHAPTER 9

Assessing the Linked Open Data Cloud's Quality

Since its inception, the *Linked Open Data (LOD) Cloud* [112] has been a point of reference to the Linked Data community, comprising a number of linked datasets crawled on the Web of Data or added to the *LODCloud* group in *datahub.io* registry[1]. The maintainers provide a set of criteria for the inclusion of a dataset within the LOD Cloud; more specifically, datasets should be published according to the Linked Data principles as defined in [22]. The Linked Data principles, closely related to the five star scheme for publishing open data, can be summarised into *the publishing of open, linked, structured data, in non-proprietary formats, using URIs*.

Linked Data resources are usually a complex structures encompassing some existing thing (an object in the real world), giving it semantics (i.e. meaning) and possibly linking to other resources, that both machines and humans can understand. According to the editors of the W3C Data on the Web Best Practices document,

> *"data quality can affect the potentiality of the application that uses data, as a consequence, its inclusion in the data publishing and consumption pipelines is of primary importance."*
> –[105, §9.5]

Making data quality more transparent and easy-to-access is a key factor for the wider penetration of Linked Data and semantic technologies. In this study, we perform a large scale evaluation of Linked Data quality in terms of data size, domain and quality indicator coverage. We assess and quantify the quality of a number of datasets in the Linked Open Data Cloud over a number of quality indicators. This is done to understand better the quality of the datasets being published on the Web (cf. research question 4 in Section 1.3). Furthermore, such an investigation may lead to other insights, such as identifying which of the assessed metrics are the most informative to describe the quality of a linked dataset (cf. Section 9.4.2)

Using Luzzu (cf. Chapter 4) and a number of quality metrics (including a number of probabilistic approximation metrics), this study produces a quality metadata graph for each assessed dataset (publicly available for consumption as Linked Data resources), represented in daQ (cf. Chapter 6). The benefits of these metadata graphs are twofold: (1) humans can understand the quality of a dataset better, using ranking or visualisation tools, thus making more informed decisions prior to using a dataset; and (2) machines can automatically process the quality metadata of a dataset.

The remainder of this chapter is structured as follows. In Section 9.1 we perform a primary investigation towards the *openness* of the Linked Open Data, followed by the dataset acquisition description in

[1] https://datahub.io/group/lodcloud

Section 9.2. Following the data acquisition process, in Section 9.3 we assess and discuss the quality of these datasets against twenty-seven metrics related to four different quality categories as described in [160]. We then use the assessment results in order to identify the non-informative quality metrics in Section 9.4.2, followed by the conclusions in Section 9.5.

9.1 'O'penness in the Linked Open Data Cloud

Open Data, in terms of the *Open Definition* should be possible to

> *... be freely used, modified, and shared by anyone for any purpose* [58].

More specifically, open data should [58, §1]:

1. have a defined open license or status - having a license is the only way to define boundaries between the publisher and the consumer (who can also re-publish the data without worrying about using the data improperly);
2. be accessible, i.e. in the case of Linked Open Data a dataset should have some entry point such as a data dump or SPARQL endpoint (preferably referred to in dataset metadata defined by standard vocabularies);
3. be machine readable, if possible interoperable i.e. using for example RDF;
4. have an open format.

Drawing parallels with Linked Open Data, Berners-Lee proposed the five-star open data principles, in which the first three stars are similar to the principles defined in the Open Definition, whilst the last two are more related to the Linked Data principles, i.e. (4th star) the use of URIs to identify things, and (5th star) linking between the published data and external data [22].

Having metadata as part of a published dataset is the first step in putting a dataset on the open data map (thus encouraging discoverability [135]), as it is generally the first access point for consumers who wish to use the published data. Metadata ensures that it complies with best practices by making it self-descriptive [74, §5.5]. Therefore, 'doing metadata right' is a must for any kind of published open data. In a holistic assessment of open government data initiatives, Attard et al. [11] describe a number of initiatives that had the aim to assess the quality of metadata. This shows further the importance metadata is given in open data.

Heath and Bizer provide a checklist for Linked Data publishing, which includes the provision of provenance metadata, licensing metadata, and dataset level metadata in terms of standard vocabularies such as VoID [7] and DCAT [108]. Schemas like DCAT and VoID enable metadata description in a semantically interoperable format and can be exchanged between various agents. Currently, there are other schema initiatives such as daQ (cf. Chapter 6) and DQV [5] to represent quality metadata for datasets, and the DUV [106] to describe various factors of a dataset such as citation and feedback from a human consumer perspective.

The current LOD Cloud snapshot was taken in 2014, containing about 188 million crawled triples[2]. Metadata description of these datasets can be easily retrievable from the Linked Data catalog published together with the latest snapshot. In a recent study, Assaf et al. [10] gave an insight towards the metadata available in the Linked Open Data Cloud. The authors concluded that the quality regarding the available

[2] This number was taken from http://data.dws.informatik.uni-mannheim.de/lodcloud/2014/ISWC-RDB/, although the actual number of triples in the referred datasets is larger

metadata information is in a bad condition. More specifically, licensing and accessibility metadata contained noisy data, thus resulting in incorrect information.

Whilst the CKAN API includes a metadata export functionality in terms of DCAT [108], metadata of new datasets imported to the catalog is generally manually added as textual description, thus it is prone to errors such as inconsistency and duplication. For example, in the *formats* tags, we find a variety of tags referring to the same format (the number in brackets refer to the number of datasets tagged):

- application/rdf+xml (17); application/rdf xml (4);
- api/sparql (368); sparql (4);
- text/turtle (75); ttl (10); rdf/turtle (7); turtle (2);

We find also a number of tags that we could not match with an appropriate format or else tags with formats of a proprietary nature, for example:

- RDF (187) [possibly application/rdf+xml, but this had to be verified manually];
- xhtml, rdf/xml, turtle (2) [this is one tag with three possible formats];
- example/* (2);
- mapping/twc-conversion (5)

Having a variety of formats in such metadata would hinder the potential re-use of datasets by automated agents as they would not be able to decipher the type of data in question automatically. In order to follow the best Linked Open Data practices, such metadata should be standardised and interoperable between different machines, for example, the use of ontologies such as the *Media Types as Linked Data ontology* [129] should be considered in order to standardise the metadata effort between datasets within a catalog.

9.1.1 LOD Cloud Datasets' Accessibility

In order to identify which datasets had some kind of access points, an initial experiment was performed on the latest LOD Cloud snapshot[3,4]. The LOD Cloud snapshot has a total of 569 datasets. Based on the metadata provided in the datahub, only around 42% (239 datasets) had a possible[5] Linked Data access point, i.e a data dump URI, SPARQL endpoint, or a VoID dataset description. From the 239 datasets, 50% of the datasets had multiple access points, 33 datasets only had a data dump defined, 74 had a SPARQL endpoint, whilst 13 datasets had just a VoID description URI defined. Figure 9.1 depicts datasets from the LOD Cloud snapshot that are actually accessible.

9.1.2 LOD Cloud Datasets' Licenses and Rights

Licences are the heart of Open Data. It is the mechanism that defines whether third parties can re-use or otherwise, and to what extent. In Linked Open Data, one would expect that such licenses are machine readable using predicates such as `dct:license`, `dct:rights` and `cc:licence`, and possibly also in a human readable format (e.g. within `dc:description`). Such license specification should

[3] http://lod-cloud.net/versions/2014-08-30/lod-cloud.svg

[4] These initial experiments were performed in December 2015, prior to the actual quality assessments. This was part of the data acquisition process which is described in Section 9.2

[5] We added a validation stage which is described in Section 9.2

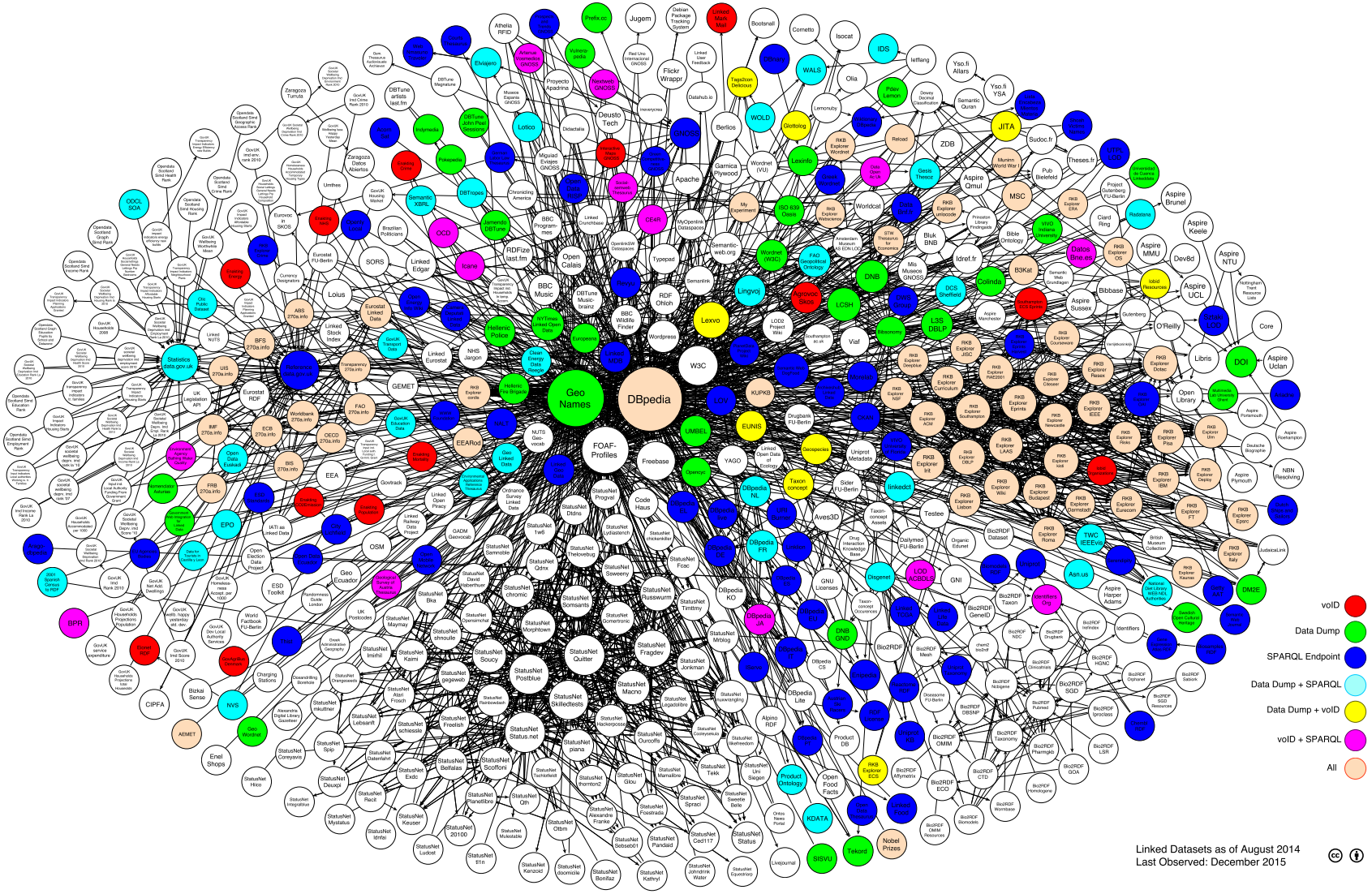

Figure 9.1: Coloring the LOD Cloud Datasets with various Access Methods (Data Dump, voID, SPARQL Endpoint, or a combination)

also be included in a dataset's metadata. Another initial experiment was performed on the LOD Cloud snapshot to check how many datasets provide some kind of machine readable license on the datahub provided metadata. In total, only 40.42% (230 in total) of all datasets represented in the current LOD Cloud snapshot have some kind of license (or rights) defined in a semantic manner. In Table 9.1, we list the licenses used within the LOD Cloud snapshot, with the Creative Commons Attribution License (*cc-by*) being used the most (93 instances), followed by the Creative Commons Attribution Share-Alike License (*cc-by-sa*; 47 instances) and the Creative Commons Attribution Non-Commercial V2.0 License (*cc-by-nc 2.0*; 31 instances). In spirit of the Open Data definition described in the introduction, the *cc-by-nc 2.0* license is deemed as a non-conformant[6] license since it does not support some of the definition's principles, more specifically the principle that Open Data could be re-used for any purpose, including commercial purposes [58, §2.1.8]. It was noted that 7 out of 9 licenses used in the dataset's metadata were non-semantic resources (i.e. cannot be dereferenced to an RDF description). In Linked Data, publishers of such metadata should re-use RDF resources, such as Creative Commons[7] [1] and RDF License[8] [136]

A number of data publishers declared the datasets' license (and subsequent rights description) in a human readable manner in the textual description, for example `https://datahub.io/dataset/uniprot`. A regular expression[9] that captures *license* or *copyright* and one of *under*, *grant*, or *right* was performed on all metadata descriptions in order to identify possible license definitions on a dataset. 13 datasets had this kind of human readable license declaration (results displayed in brackets in Table 9.1). This second experiment identified 5 new licenses used in the LOD Cloud snapshot, two of which (Creative Commons Attribution-NonCommercial-ShareAlike V3.0 and Project Gutenberg License) are non-conformant to open data. Figure 9.2 shows the datasets with a declared license.

9.1.3 The LOD Cloud Snapshot and its Future

From our preliminary investigation on the available metadata, we have identified that approximately less than half of the datasets should be part of the Linked **Open** Data cloud, as they do not satisfy the properties of *Open Data*. Furthermore, the Web of Data, unlike the LOD Cloud snapshot, is volatile. Datasets on the web, although undesirable, are unpredictable, and thus features, more specifically access points, might not be available on the cloud at all times. Changes in documents themselves could also change the shape of the LOD Cloud as we know it. Such dynamics of the Web of Data are described further in [91]. Käfer et al. [91] presented the Dynamic Linked Data Observatory[10], from which a comprehensive analysis over 29 weeks was conducted. Their study show that around 60% of the data(sets) did not change, 5% went offline, whilst the rest had changes in the document itself. SPARQLES[11], a tool monitoring the availability of public SPARQL endpoints (amongst other tests), shows that only around 45% were available (from a total of 549 publicly available endpoints monitored) at the time of study[12]. Whilst this percentage is low, we noticed a small (insignificant) average change in the uptime of 0.002% between 24th February 2016 and 2nd March 2016. Downtime can be caused by various issues, such as network failures or high server load. Availability statistics, provided by SPARQLES, show that as at November 2015, 181 endpoints (around 32% from 549 endpoints) have a $\geq$ 99% uptime. In April 2015, this number stood 242, therefore over a period of 6 months, 12% of these endpoints became less reliable. Overall,

[6] `http://opendefinition.org/licenses/nonconformant/`. Date Accessed 10th October 2016
[7] `https://creativecommons.org/ns`. Date Accessed 10th October 2016
[8] `http://purl.org/NET/rdflicense/`. Date Accessed 10th October 2016
[9] `.*(licensed?|copyrighte?d?).*(under|grante?d?|rights?).*`
[10] `http://swse.deri.org/dyldo/`. Date Accessed 10th October 2016
[11] `http://sparqles.ai.wu.ac.at/`. Date Accessed 10th October 2016
[12] As of 2nd March 2016

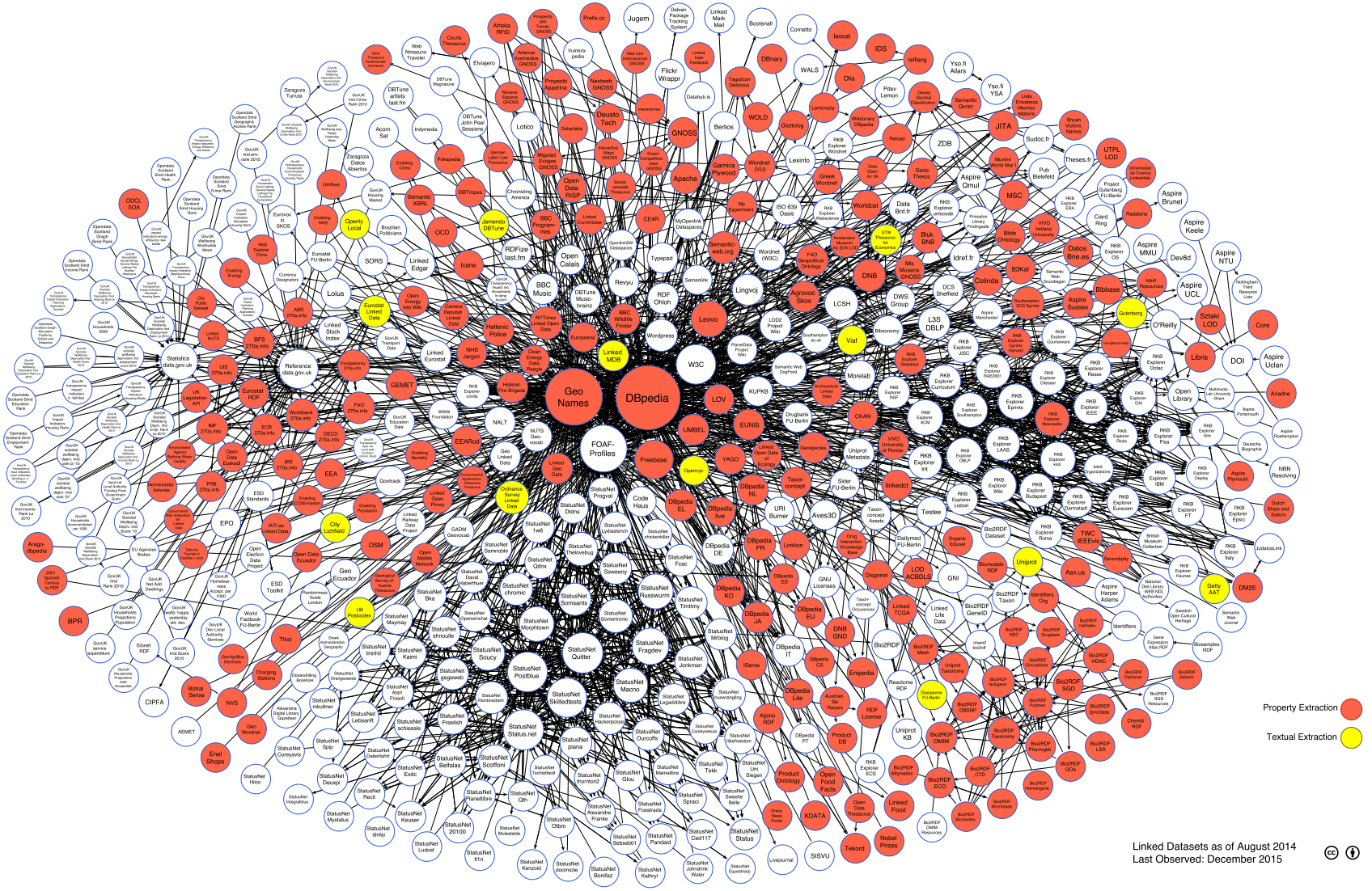

Figure 9.2: Coloring the LOD Cloud Datasets with Licence Availability extracted either via machine readable properties or using regular expressions from textual descriptions.

License Used	Type of License	URL Used	Semantic Resource	Frequency
Creative Commons Attribution License	Requires Attribution	http://www.opendefinition.org/licenses/cc-by	✗	93 (+2)
Creative Commons Attribution Share-Alike License	Requires Attribution and Share Alike	http://www.opendefinition.org/licenses/cc-by-sa	✗	47 (+1)
Creative Commons Attribution Non-Commercial V2.0 License	Requires Attribution but dataset cannot be used for commercial purposes. This license is a non-conformant license for open data.	http://creativecommons.org/licenses/by-nc/2.0/	✓	31 (+1)
Creative Commons CC Zero License	Public domain waiving all rights on the data	http://www.opendefinition.org/licenses/cc-zero	✗	30 (+1)
Open Database License	Requires Attribution and Share Alike	http://www.opendefinition.org/licenses/odc-odbl	✗	9
Open Government License for Public Sector Information	Requires Attribution. License can only be used by third parties licensed by the UK Government	http://reference.data.gov.uk/id/open-government-licence	✓	6
Open Data Commons Public Domain Dedication and Licence	Public domain waiving all rights on the data	http://www.opendefinition.org/licenses/odc-pddl	✗	5 (+1)
Open Data Commons Attribution License	Requires Attribution	http://www.opendefinition.org/licenses/odc-by	✗	5
GNU Free Documentation License	Share Alike	http://www.opendefinition.org/licenses/gfdl	✗	4
Creative Commons Attribution-NonCommercial-ShareAlike V3.0	Requires Attribution and Share Alike but dataset cannot be used for commercial purposes.	-	-	(+2)
OS Open Data License	Requires Attribution and Share Alike	-	-	(+2)
Eurostat Policy	Requires Attribution	-	-	(+1)
Project Gutenberg License	Restricts Commercial Use	-	-	(+1)
Creative Commons Attribution-NoDerivs License	Does not allow work to be re-used in derivative works	-	-	(+1)

Table 9.1: List of licenses used in the metadata, extracted by machine readable properties and from human readable descriptions (values in brackets).

239 endpoints (around 44% - as at November 2015) are the least reliable, having an uptime of < 5%. In the future, if the LOD Cloud snapshot is to represent the state of the Web of Data, these dynamics should also be considered. Thus, ideally the LOD Cloud snapshot is dynamically updated as datasets are added, die and change.

9.2 Dataset Acquisition Process

In this section we detail the process for defining possible datasets that are used for the empirical study. Our main goal was to automate the whole process, whilst retrieving as many datasets as possible. The metadata of the 2014 LOD Cloud was taken as the primary corpus for this study. Each dataset in the LOD Cloud, grouped by their fully qualified domain name (FQDN)[13], has a corresponding generated DCAT metadata entry in the datahub.io portal. Metadata descriptions of these datasets can be easily retrieved from the catalogs Linked Data interface.

9.2.1 Identifying Datasets' Access Points

For this initial experiment we retrieved the distribution resources (from the property `dcat:distribution`) defined in the dataset metadata (`dcat:Dataset`), in order to identify the media types and corresponding URLs where the dataset is made available for consumption. We aimed to identify the **data dump** (containing all triples of the dataset), a **SPARQL endpoint** description, and a

[13] A fully qualified domain name (FQDN) is the complete name for a specific host, for example `de.dbpedia.org` is the FQDN for the German chapter of DBpedia, whilst `pt.dbpedia.org` is the Portuguese version of DBpedia.

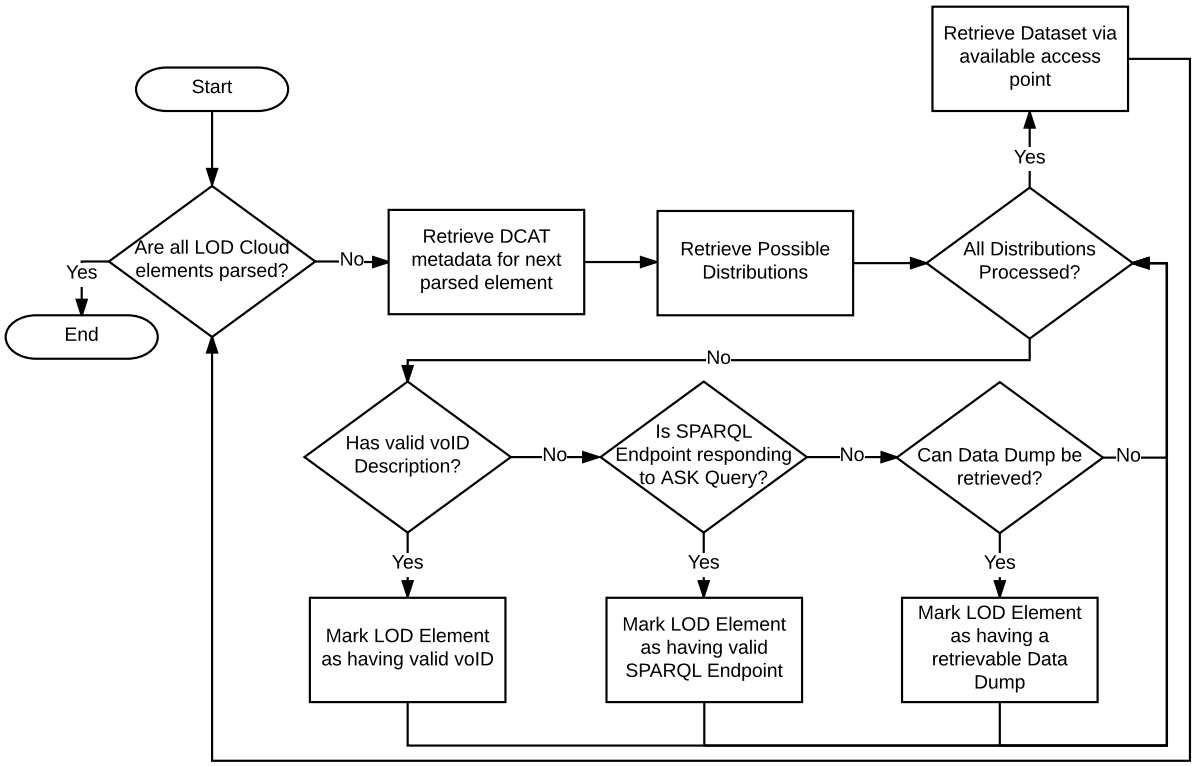

Figure 9.3: A high-level flowchart depicting the marking and retrieval process of datasets from the LOD Cloud.

VoID description for each dataset. Ideally, a dataset description provides all three resources. Figure 9.1 shows the LOD Cloud indicating the retrieved datasets and their respective (meta) data access methods, whilst Figure 9.3 shows an overview of the marking and subsequent retrieval process of the LOD datasets used for the assessment.

With regard to data dumps, we looked for media types that are generally associated with the Semantic Web, such as `application/rdf+xml` (which is the minimal requirement for any linked dataset [74, §5.1]) and `text/turtle`. In pursuance of acquiring the largest possible linked dataset coverage, we identified other possible wrongly tagged media types (e.g. `rdf`) and added them to our script[14].

Similarly, for SPARQL endpoints we looked at those distribution resources with a `api/sparql` media type. If the dataset had no SPARQL distribution defined, we probed for availability of a SPARQL endpoint by accessing the path `/sparql` at the fully qualified domain name. Having such a canonical endpoint path is a common practice. In fact, 69.58% of endpoints registered in SPARQLES end with the path `/sparql`. If a SPARQL endpoint is available, we perform a simple *ASK* query to check whether the endpoint responds to queries.

VoID descriptions were retrieved from media types containing `void` in their value. Typical media types included `meta/void`. Similar to SPARQL endpoints, if a VoID description is not available as part of the distribution, we look for the metadata in the `/.well-known/void` path of the FQDN, as recommended in [7, §7.2], following the RFC 5785 [123] practices. The VoID metadata is checked for a `void:Dataset`, in order to retrieve possible data dumps (via the `void:dataDump` property) or access the SPARQL endpoint (via the `void:sparqlEndpoint` property).

[14] All experiments can be replicated by downloading the scripts available on GitHub: `https://github.com/jerdeb/lodqa`

Following this methodology the acquired dataset collection has a number of known bias factors:

- the harvesting of datasets from the LOD Cloud was performed in December 2015 and the download of the data dumps between December 2015 and February 2016, thus the quality assessment of these datasets reflects the dumps available at the time of download (this does not apply to SPARQL endpoints);
- the downloaded data dumps cover a wide range of tagged media types (also considering incorrect tags), but our assessment is limited to the following: `application/rdf+xml`, `text/turtle`, `application/x-ntriples`, `application/x-nquads`, `text/n3`, `rdf`, `text/rdf+n3`, `rdf/turtle`;
- distributions with `example` in their title were ignored even though they had a correct media type, as we are only interested in having complete datasets (where possible) for our large-scale quality assessment;
- SPARQL endpoints that did not respond to the *ASK* query were considered unavailable and thus not included in the follow-up assessment.

The downloaded data dumps require some data preparation prior to assessment. Each dataset might have multiple distributions, some defining different sub-datasets, others defining the same dataset with different media types (for different serialisations). All dumps in these distributions are downloaded, and then converted to n-quads, merged, sorted, and cleaned by removing duplicate quads. All datasets are identified using their fully qualified domain name.

9.2.2 Datasets' Summary

During the acquisition process, we identified 239 accessible datasets – i.e. data dump, SPARQL endpoint, or VoID dataset:

- 13 datasets having only accessible VoID metadata;
- 34 datasets with only a data dump corresponding to the pre-defined media-types;
- 73 datasets with only a SPARQL endpoint;
- 34 datasets having both a data dump and a SPARQL endpoint;
- 9 datasets having both VoID metadata and a SPARQL endpoint;
- 14 datasets having both VoID metadata and a data dump;
- 62 datasets having all three possible access points.

Figure 9.4 illustrates this summery in a Venn diagram. The data dumps and SPARQL endpoints overall comprised approximately 5 billion quads.

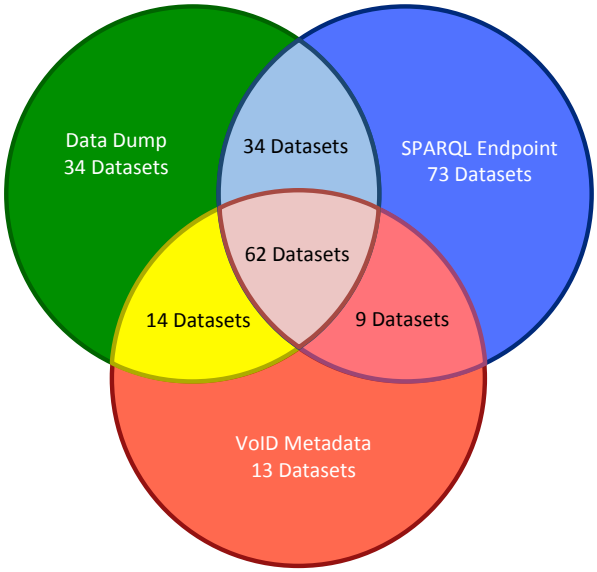

Figure 9.4: A Venn Diagram illustrating a summery of the datasets' access points.

9.3 Quality Assessment

In this study we are mainly interested in understanding the persistent quality issues within the LOD datasets, rather than the performance of the quality metrics. Furthermore, this study complements the work undertaken in the survey by Zaveri et al. [160] and the work that survey refers to, by analysing the quality of a collection of LOD Cloud datasets against a number of metrics classified in the mentioned survey. In general, the assessment is done locally, meaning that no inferencing or dereferenceability of external resources is done, unless required by the quality metric. For each data quality metric we plot a box and whiskers chart to summarise metric values and display them on a single graph. Furthermore, with the box and whiskers plot, we describe the *sample*'s spread of quality values amongst the LOD Cloud datasets. During the assessment we also collect a sample of the quality problems found during assessment, in order to describe typical problems found in LOD datasets.

9.3.1 Choice of Data Quality Metrics

In this empirical study we assess the datasets against 27 quality metrics described in [160], and two additional quality metrics describing provenance information. The majority of the 27 metrics are objective metrics, that is, the metrics' results will not be influenced by the assessor's opinion. For the only subjective metric in this study (re-use of existing terms, cf. Metric IO1), we used the LOD Cloud category classification as the basis of our classification in order to limit any bias. Furthermore, with this subjective metric we show that the Luzzu framework can handle both subjective and objective metrics, as described in Pipino et al. in [131] (cf. Section 2.1.1).

Since the assessed datasets come from a variety of domains, a certain quality metric might not be relevant, hence some datasets might fare poorly for these particular metrics. Following the overall quality assessment, users can then use the generated quality metadata to rank and filter datasets based on their choice of metrics.

The choice of generic quality metrics was based solely on the classification in [160]. Nonetheless, there is no study confirming the usefulness of such metrics, and whether or not these quality metrics are

informative in a generic assessment such as in this study. In order to examine this phenomenon, following the quality assessment of the datasets, we statistically analyse the assessment results in order to determine which of the chosen quality metrics are key quality indicators.

9.3.2 Representational Category

In this section we look at metrics related to the design of data, or in other words: how well the data is represented in terms of common best practices and guidelines. Zaveri et al. [160] categorised a number of metrics in this category within the four dimensions *Representational Conciseness*, *Interoperability*, *Interpretability* and *Versatility*.

(RC1) Keeping URIs short

Classified in the representational-conciseness dimension, this metric observes the length of URIs. In the Cool URIs document [142], the editors remarked that apart from providing descriptions for people and machines, the best URIs are *simple*, *stable*, and *manageable*.

This metric focuses on the *simplicity* aspect of this definition, where by *simplicity* the editors of the same document mean that having short and mnemonic URIs are easier for humans to remember (e.g. `http://danbri.org/foaf` vs. `https://w3id.org/lodquator/resource/826514e9-e34a-40a2-bc8d-9e6b8bd54770`), whilst serving the purpose of being machine processable. Hogan et al. [80] remarked that short URIs have other benefits such as allowing for smaller sized datasets and indexes.

Metric Computation: The metric computation is based on the W3C best practices for URIs, where the editor suggests that a URI should not be longer than 80 characters [147, §1.1]. Furthermore, URIs with appended parameters are considered as bad, irrelevant of their length. The metric can be quantified as follows:

$$RC1(D) := \frac{size(\bar{u} = \{u \mid ((len(u) \leqslant 80) \wedge ('?' \notin u))\})}{size(dlc(D))}$$

where $\bar{u}$ is a set of URIs having a length (defined by *len*) of 80 or less and are not parameterised (URI contains no '?') and $dlc(D)$ is the set of possible data-level constants on dataset D (i.e. the dataset being assessed). A data-level constant is defined by [80] as the subject or the object of a quad, when the predicate is not `rdf:type`. Therefore, this metric value measures the ratio of short URIs.

Discussion: A box plot with the quality values is illustrated in Figure 9.5. The box plot for this metric (RC1) is comparatively tall, suggesting that publishers tend to have quite different inclinations on how long the URI identifiers should be. The sample over the LOD Cloud is centred on 97.92% with a sample standard deviation (σ_s) value of 24.90%. A number of outliers (around 13% of the datasets), were detected. These are datasets that scored lower than 54.55% (i.e the lower whisker). The range of quality values, including outliers, is 99.75%, whilst the range between quartile group 4 and quartile group 1 is 45.45%. We also notice that the population is skewed to the left (i.e the median is closer to the third quartile), whilst the bottom whisker is longer than the top whisker (since we cannot have a value greater than 100%), suggesting that most quality values are large with some smaller values. The average quality value across the assessed datasets is around 84.07%, with 69% of the datasets scoring more than 90%, and 28% of the datasets scoring 100%. From the sample problem report we extracted during the assessment, 14.65% of the URIs were parameterised whilst the rest where URIs longer than 80 characters.

This metric has two drawbacks. First, our metric takes into consideration external URIs, however, we acknowledge that the length of such external URIs cannot be influenced by the datasets' publisher. A solution for this is that the metric looks only at locally minted URIs. The second drawback of this metric

is the lack of discriminative power, since URIs with 80 characters are fine, while longer ones are deemed to be "bad". There might be various reasons for publishers to use longer URIs. For example, URIs can comprise some structure, such as a directory scheme. In order to avoid the discriminatory power problem, Hogan et al. [80, §5.1 – Issue IV] calculate the metric value based on the average length of the URIs in a dataset, promoting those datasets that have short URIs. In our case the metric is more flexible with regard to the typical length of URIs used in datasets. Furthermore, we deem that publishers who use domain names with various levels (e.g. typical university URIs) and still adhere to the recommendation should not be given a lower quality value.

(RC2) Minimal Usage of RDF Data Structures

The usage of RDF data structure features, more specifically reification, containers, and collections, is discouraged due to their syntactic/semantic complexity. Despite the fact that a number of efforts were made in order to facilitate the use of such data structures (e.g. the introduction of property paths[15] in SPARQL 1.1 allows the retrieval of all members in an `rdf:List` with one graph pattern: `{?s rdf:rest*/rdf:first ?o .}` - this was not possible in SPARQL 1.0), these are still more complicated to handle. In [74, §2.4.1.2], the author discourage the use of RDF reification since they "are rather cumbersome to query with the SPARQL query language". Furthermore, the authors argue that if set ordering is not required, collections and containers are best avoided. In RDF, these data structures are typically described using blank nodes, which is another discouraged practice (cf. Metric IN4). In [80, §5.3 - Issue VIII], Hogan et al. explain the various issues, such as scalability and lack of semantics, that these features bring about.

Metric Computation: This metric detects the use of standard RDF data structure features. More specifically, this metric checks quads as suggested in [80, §5.3 - Issue VIII]:

- if the predicate is `rdf:type` and the object is one of `rdf:Statement`, `rdf:Alt`, `rdf:Bag`, `rdf:Seq`, `rdf:Container`, or `rdf:List`;
- if the predicate is one of `rdf:subject`, `rdf:predicate`, `rdf:object`, `rdfs:member`, `rdf:first`, `rdf:rest`, or `rdf:_'[0-9]+'`.

The value of this metric can be quantified as follows:

$$RC2(D) := 1.0 - \frac{size(RCC(D))}{size(quads(D))}$$

where $RCC(D)$ is the set of quads from dataset D that satisfy the above conditions, and $quads(D)$ is the set of all quads in dataset D. Therefore, the metric value is a ratio of quads in a dataset with and without discouraged RDF data structures.

Discussion: Similar to the findings of Hogan et al. [80], most publishers do not use RDF data structures. In our assessment 87.2% of the publishers use none, compared to the 78.7% reported by Hogan et al. This is reflected in the short box plot illustration for this metric (RC2 - Figure 9.5), with the interquartile ranges and whiskers all being close to 100%. The average quality value of this metric is 99.44% and the calculated σ_s 2.86% (median value is 100). The σ_s value confirms our findings that most publishers try to minimise the use of such undesired RDF features, with 97% of the datasets ranking within 1 σ_s (i.e. having a quality value of at least 96.369%). Similar to the Short URIs metric (RC1), a relatively small number of outliers (around 12% of the datasets) were detected. Nonetheless, the dataset with the lowest

[15] http://www.w3.org/TR/sparql11-query/#propertypaths. Date Accessed 10th October 2016

quality value for this metric (http://bibsonomy.org) is 70.25%. Upon further inspection of this dataset, we found that the publisher used rdf:Seq and rdf:Bag in order to list information such as editors and authors of some publication. In general, the RDF collections were the most common issue (95.23%), followed by RDF containers (3.09%) and RDF reification (1.67%).

(IO1) Re-use of Existing Terms

Vocabulary re-use is widely advocated. For instance, Bizer and Heath [27] argues that re-using terms from known vocabularies makes it easier for applications to process Linked Data, thus increasing interoperability between agents. Schemas for different domains are meanwhile publicly available; also via registries such as the *Linked Open Vocabulary* (LOV) portal[16]. Together with W3C recommendation vocabularies such as *SKOS*, schemas such as *FOAF*, *Dublin Core*, and *SIOC*, amongst others, have become de-facto standards with more than 15% of the LOD datasets using at least one of these vocabularies [143]. Furthermore, the W3C is striving to create standardised cross-domain vocabularies, such as *DCAT* and *PROV-O* amongst others. Zaveri et al. [160] classify this metric under the interoperability dimension, and focus on the overlap between the dataset in question and its overlap with recommended vocabularies [80, §5.3 - Issue IX].

Metric Computation: This metric assesses if a dataset re-uses relevant terms in a particular domain. More specifically, each dataset is tagged with the domain as classified by the LOD Cloud, for example, the Lexvo dataset is tagged as *linguistics*. The LOV API is then queried with 'linguistics' and the schemas given by the service are used. In particular, this metric checks if a property or a class (in case the predicate is rdf:type) used in a triple refers to an existing term in another vocabulary. Since the metric depends on the domain of the dataset, for this experiment all LOD Cloud datasets were tagged according to their identified domain in the cloud itself (e.g. DBTropes is tagged with the label *media*). During the initialisation of the metric, the LOV API[17] is invoked to obtain the vocabularies available with the respective tag. Furthermore, based on the usage study conducted in [143], we included the following vocabularies by default for all datasets: RDF, RDFS, FOAF, DCTerms, OWL, GEO, SIOC, SKOS, VOID, DCAT.

We identify overlapping classes and properties in the same manner as defined in [80, §5.3 - Issue IX], with the set of known vocabularies generated from LOV. The metric counts the number of external classes and properties (from external vocabularies identified by LOV) for a particular domain:

$$IO1(D) := \frac{size(\{x \mid x \in \overline{v_c} \wedge x \in class(D)\}) + size(\{y \mid y \in \overline{v_p} \wedge y \in prop(D)\})}{size(class(D)) + size(prop(D))}$$

where $class(D)$ is the set of classes in the assessed dataset D, appearing in the object position with predicate rdf:type excluding blank nodes. The set $prop(D)$ defines the set of terms appearing at the predicate position of the quads in the dataset D, excluding rdf:type. $\overline{v_c}$ and $\overline{v_p}$ are the sets of **all** classes and properties respectively, gathered from the identified external vocabularies for the particular dataset. Therefore, the metric value is a ratio of the number of external terms (classes and properties) vs. the number of terms used in the dataset.

Discussion: The box plot for this metric (IO1 - Figure 9.5) is comparatively long and skewed to the left, suggesting that most values are small with some larger values. This also suggests that there is a lack of conformity on the principle of re-use; only few publishers rely actively on the re-using vocabularies ($\approx$ 10% of datasets have a quality value of > 90%), with 8.8% of the datasets being outliers in this case

[16] http://lov.okfn.org/. Date Accessed 10th October 2016
[17] http://lov.okfn.org/dataset/lov/api/v2/vocabulary/search. Date Accessed 10th October 2016

as they have a quality value larger than 92.32% (i.e. the upper whisker value). The sample is centred on a value of 24.00% with a sample standard deviation (σ_s) value of 29.10%. More concerning is the mean value of 34.01%, indicating the low overall re-use. One possibility is the fact that publishers (such as DBpedia) use local terms and properties with few external properties (e.g. `rdfs:label`). Our values are comparable to those in Hogan et al. [80, §5.3 – Issue IX], where the authors also suggest that the amount of re-used terms and properties in their sample is widely distributed (the σ value in their experiment is 29.05).

With the pre-defined tags associated to each dataset, we ensured that each dataset is assessed solely based on its domain, relying on the LOV service to provide us with relevant public vocabularies. This means that our assessment might have either missed some vocabularies, or expected datasets to use terms from a vocabulary which has been overlooked by the publishers. This metric does not consider user-defined terms with links to *existing* terms using predicates such as `owl:sameAs`, `owl:equivalentClass`, or `owl:equivalentProperty`, as being a valid re-used existing term as described in this metric.

In order to improve schema re-use, services such as LOV and Swoogle[18] should be used to find suitable schemas. On the other side, vocabulary curators should maintain and promote their schemas, for example, by making sure that vocabularies are properly dereferenceable.

(IN3) Usage of Undefined Classes and Properties

The invalid usage of undefined classes and properties metric is classified under the interpretability dimension [160], which targets the technical representation of the data itself. Using classes and properties without a formal definition (i.e. not defined in a schema) is undesirable, as agents would not be able to understand how the data should be interpreted, for example, during reasoning. Errors that lead to such invalid usage includes: capitalization errors (e.g. `foaf:person` vs. `foaf:Person`), syntactic errors (e.g. `foaf:img` vs. `foaf:image`), and dereferencability issues (cf. Section 7.1.1) with external schemas (e.g. schema not available anymore, or not in machine-readable format).

Metric Computation: This metric measures the number of undefined classes and properties in the assessed dataset:

$$IN3(D) := 1.0 - \frac{size(\{x \in V_c \mid \exists V \cdot ns(x) \mapsto V \wedge x \in class(D)\}) + size(\{y \in V_p \mid \exists V \cdot ns(y) \mapsto V \wedge y \in prop(D)\})}{size(class(D)) + size(prop(D))}$$

where V_c is the set of classes (where a class is defined as being of type `rdfs:Class` or `owl:Class`) in a vocabulary V which is resolved by an agent using the namespace of the term[19] x ($ns(x)$). Similarly, V_p is the set of properties (where a property is defined as being of type `rdf:Property`, `owl:ObjectProperty`, `owl:DatatypeProperty`, `owl:AnnotationProperty`, or `owl:OntologyProperty`) in vocabulary V. Therefore, the metric value shows how much of a dataset uses classes and properties that are formally defined. In order to check if a class or property (term) is defined, the term is dereferenced for its semantic description and queried for properties and classes. If a term is non-dereferenceble, it is considered undefined.

Discussion: The box plot for this quality metric (IN3 - Figure 9.5) is relatively tall, covering a range of 99.58%. This suggests that data publishers are using a wide range of defined and undefined classes and properties. Furthermore, the quality value is centred (median) at 53.33% with a σ_s value of 32.18%, whilst the average quality value is 54.48%. Although the box plot might seem symmetrical, the values

[18] http://swoogle.umbc.edu. Date Accessed 10th October 2016

[19] In cases where slash URIs are used, the namespace does not necessarily resolve the schema, therefore the term is used to resolve the term's description.

Dataset	V1(D)
zbw.eu/stw	4
linkedmarkmail.wikier.org	3
nhs.psi.enakting.org	3
population.psi.enakting.org	3
crime.psi.enakting.org	3
...	
vocab.nerc.ac.uk	0
wals.info	0
www.productontology.org	0
bfs.270a.info	0
cordis.rkbexplorer.com	0

Table 9.2: Top and Bottom 5 ranked datasets for the different serialisation formats metric.

Dataset	V2(D)
nhs.psi.enakting.org	15
population.psi.enakting.org	15
crime.psi.enakting.org	15
co2emission.psi.enakting.org	15
rdfdata.eionet.europa.eu	13
...	
education.data.gov.uk	1
vocab.nerc.ac.uk	1
wals.info	1
www.productontology.org	1
cordis.rkbexplorer.com	1

Table 9.3: Top and bottom 5 ranked datasets for the multiple language Usage metric.

are skewed to the right by a small margin ($\approx 5\%$).

A higher value means that less undefined terms were used in the dataset. From our assessment, 30.80% of properties used were undefined. Some of the undefined terms were possibly previously defined. For example, for the rkbexplorer datasets, the publishers use terms from the `aktors.org` namespace, which now resolves to a personal blog. We noticed that apart from undefined terms, publishers use terms that were wrongly defined, for example, `rdfs:Property` as opposed to `rdf:Property`. Other datasets had schemas that were unavailable during the assessment, thus resulting in undefined terms.

(IN4) Usage of Blank Nodes

Blank nodes are undesirable in Linked Data because they cannot be externally referenced, which conflicts with the two Linked Data best practices interlinking and re-using. In simple terms, the scope of blank nodes is *"limited to the document in which they appear"* [74].

Moreover, the existence of blank nodes can cause a number of problems during Linked Data consumption and when performing certain tasks, such as deciding whether two RDF graphs are isomorphic. In SPARQL, blank nodes behaviour is unpredictable in *RDF equivalent graphs*, whilst they cannot be referenced during querying [111].

Metric Computation: This metric assesses the usage of blank nodes within the subjects and objects. The metric value is assessed as suggested in [80, §5.1 – Issue I]:

$$IN4(D) := \frac{size(dlc(D))}{size(dlc(D) \cap bn(D))}$$

where $dlc(D)$ is the set of data-level constants in dataset D and $bn(D)$ is the set of blank nodes in D. The value represents the degree of **avoiding** the usage of blank nodes.

Discussion: The box plot (IN4) illustrated in Figure 9.5, is relatively short, suggesting that most data publishers agree to avoid blank nodes. Furthermore, the box plot range is 5.07%, with the upper whisker and third quartile at the 100% mark. The sample centrality is 100% and the σ_s is 12.15%. The higher the value, the less blank nodes are used in a dataset. The average quality metric value 96.01% confirms the generally high conformance with this metric.

Whilst the majority of data publishers use blank nodes sparsely or not at all (around 85% of the datasets score higher than 94.93%, which is the lower whisker limit), there are a number of datasets marked as outliers consequently stretching the σ_s value. In particular, the `prefix.cc` dataset uses blank nodes in almost every triple. This dataset affected the σ_s value significantly, which otherwise would be considerably lower than in [80, §5.1 – Issue I]. One should note that the corpus in [80] contained FOAF profiles, which traditionally contain many blank nodes. In certain situations, the usage of blank

nodes is complementary to RDF data structure features and OWL axioms, as these structures and axioms use blank nodes as the encoding, though in general avoiding them means that resources in a dataset are more likely to be re-used for linking.

(V1) Different Serialisation Formats

An RDF data model can be serialised using a variety of formats, including RDF/XML, RDFa, Turtle, N-Triples, Quads, and JSON-LD. For example, Web applications prefer the JSON-LD format, rather than having to use some parser, as the JavaScript environment handles JSON data internally. The different characteristics of each serialisation brings about different pros and cons, as described in [74, §2.4.2]. The rationale of this metric is to assess whether various consumption methods are supported. Ensuring that a dataset is available in multiple serialisation formats facilitates its use. The metric is classified under the versatility dimension [160].

Metric Computation: This metric checks whether a dataset has multiple serialisation formats defined in its metadata, by verifying that multiple quads having `void:feature` as a predicate exist in the assessed dataset. The `void:feature` predicate is used to express the technical features of a dataset, such as the serialisation formats the dataset is available in.

According to the VoID W3C recommendation, the `void:feature` *"can be used for expressing certain technical features of a dataset, such as its supported RDF serialisation formats"*. [7, §2.6] Data publishers can serialise their data in up to 23 different formats[20]. The metric can be quantified as follows:

$$V1(D) := size(features(D))$$

where $features(D)$ is the set of dataset features identified by the object in a triple **subject** × **void:feature** × **object**. Therefore, this metric returns a value indicating the number of supported serialisation formats.

Discussion: Table 9.2 shows the five top and bottom ranked datasets[21], according to the number of serialisation formats defined. In most cases, the publishers did not define any serialisation format in the metadata of their datasets. Only nine datasets had a serialisation format following our guideline. The σ_s value is 0.71 whilst the mean value is 0.18.

A dataset, serialised in different formats, widens possible uses in different scenarios. In order to encourage multiple format serialisation, tools such as *Raptor*[22] or *Serd*[23] provide command line functions that transform (bulk) data into various serialisations. One drawback is that using different serialisations takes up more storage resources. Regarding the generation of VoID metadata, generators such as [32], help publishers to create VoID descriptions.

(V2) Usage of Multiple Languages

Catering for multiple languages ensures that the dataset reaches a wider global audience. For example, a dataset with literals having only a Maltese language tag is not suitable for Chinese speaking users. On the other hand, if the dataset has literals in both Maltese and Chinese, then the dataset is likely to be used more often. A plain (textual) literal string can be combined with a language tag (e.g. `@mt`) Furthermore, the Data on the Web Best Practices document suggests that locale parameters should be provided in metadata:

[20] `http://www.w3.org/ns/formats/`. Date Accessed 10th October 2016
[21] No ties were resolved.
[22] `http://librdf.org/raptor/`. Date Accessed 10th October 2016
[23] `https://drobilla.net/software/serd/`. Date Accessed 10th October 2016

Dataset	$v(C, 1.0)$	RC1	RC2	IO1	IN3	IN4	V1	V2
http://co2emission.psi.enakting.org/	97.24%	91.43%	100.00%	97.06%	97.06%	94.59%	3	15
http://crime.psi.enakting.org/	97.06%	91.43%	100.00%	97.06%	97.06%	94.59%	3	15
http://nhs.psi.enakting.org/	96.88%	91.43%	100.00%	97.06%	97.06%	94.59%	3	15
http://population.psi.enakting.org/	96.60%	91.43%	100.00%	97.06%	97.06%	94.59%	3	15
http://thesaurus.iia.cnr.it/	95.53%	100.00%	100.00%	100.00%	100.00%	100.00%	0	2
...								
http://lod.taxonconcept.org/	43.22%	67.46%	100.00%	3.42%	9.99%	99.44%	0	1
http://wals.info/	42.66%	90.83%	100.00%	5.14%	5.14%	100.00%	0	1
http://vocabulary.semantic-web.at/PoolParty/wiki/semweb	42.49%	93.96%	100.00%	10.00%	15.30%	98.97%	0	1
http://sw.opencyc.org/	37.14%	85.10%	99.39%	0.37%	0.42%	98.36%	0	1
http://minsky.gsi.dit.upm.es/	33.69%	11.98%	100.00%	2.88%	4.33%	100.00%	0	1

Table 9.4: Overall ranking of datasets for the representational category.

> *"making the language explicit allows users to determine how readily they can work with the data and may enable automated translation services."* – [105]

The usage of multiple languages metric is also classified by Zaveri et al. under the versatility dimension [160].

Metric Computation: This metric checks the number of languages a dataset supports. Specifically, the metric checks whether the data (in this case string literals) is evenly available in different languages:

$$V2(D) := round\left(\frac{size(lt = \{o \in lit(D) \mid hasLangTag(o)\})}{size(\overline{lt})}\right)$$

where lt is a set of literals with a language tag in dataset D, and $\overline{lt}$ is the set of unique literals with a language tag. This metric value will return a natural rounded number of languages that characterise the assessed dataset.

Discussion: Table 9.3 shows the datasets that have a high number of multi-lingual textual labels. In most cases, publishers describe their textual literals using only one language ($\approx$ 83%). One possible reason is that publishers target a particular audience, or do not have the resources to create multilingual datasets. The mean value for this metric is 1.72 languages, whilst the standard deviation (σ_s) is 2.71 (median value: 1). The σ_s value shows that publishers are inclined towards supporting a lower number of languages.

Language tags allow agents to express linguistic or text-based information better, for example, providing better localisation. There are publishers who refrain from adding a language tag to the textual literals. Tools such as *Apache Tika*[24] detect the language of literals and can help publishers to add the correct tags.

Aggregated Results

Table 9.4 shows the aggregated ranking (top and bottom five datasets)[25] of datasets in the representational dimension, as described in Section 9.3.6. Figure 9.5 shows a box plot illustration of the aggregated quality value compared with the category's metrics (V1 and V2 are missing as the quality value are integers, whilst the rest are float values). The overall aggregated box plot shows a population which is slightly skewed to the left, close to symmetrical (since the mean and median values are close), with a centrality median of 60.70%. This suggest that there is more variety amongst higher quality values (i.e. more than the median) amongst the sample. Nevertheless, the deviation value (σ_s) is 14.50%, which suggests a moderate distribution, whilst the average score is 63.60%.

[24] http://tika.apache.org/. Date Accessed 10th October 2016

[25] For a full list visit http://jerdeb.github.io/lodqa/ranking.html

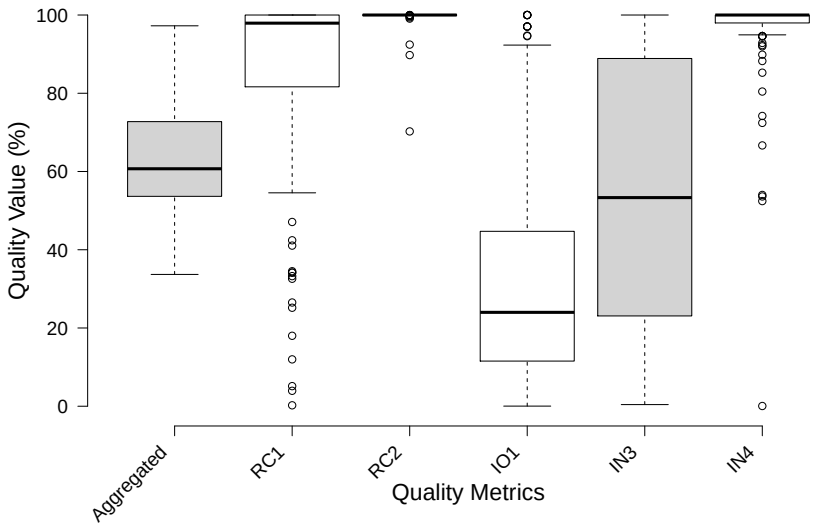

Figure 9.5: Center lines show the medians; box limits indicate the 25th and 75th percentiles; whiskers extend 1.5 times the interquartile range from the 25th and 75th percentiles, outliers are represented by dots. Different Serialisation Formats and Usage of Multiple Languages metric are excluded, but included in the aggregated result box plot.

9.3.3 Contextual Category

According to Zaveri et al. [160], the contextual category groups those dimensions and metrics that are highly dependent on the task at hand. The dimensions classified in this category deal with (i) *relevancy* of a dataset vis-à-vis the task at hand, (ii) degree of the data correctness and credibility, i.e. the *trustworthiness* of the dataset, (iii) *understandability* of the data in terms of human comprehensibility and ambiguity, and (iv) *timeliness* of data. In this chapter, we introduce a new dimension, *provenance*, which for quality purposes we define as *the provision of information regarding the origin of the dataset and the resources within the dataset itself*. The provenance metrics can be seen similar to those classified under the *trustworthiness* dimension. Furthermore, in this category, we only tackle three metrics related to *understandability*, whilst no metrics classified under the *relevancy* and *timeliness* metrics are assessed.

(P1) Provision of Basic Provenance Information

Data provenance is considered as one of the main assets in a Linked Data.

> *"Data provenance becomes particularly important when data is shared between collaborators who might not have direct contact with one another either due to proximity or because the published data outlives the lifespan of the data provider projects or organisations."* – [105, §9.4]

The importance of data provenance lies in the fact that consumers need to understand where the data

comes from and by whom it was produced. In this way, consumers can identify whether for example they could trust the integrity and credibility of the dataset.

Metric Computation: At the very least, a dataset should have a `dc:creator` or `dc:publisher` within their VoID or DCAT metadata. We focus on searching for triples with the predicates `dc:creator` or `dc:publisher` in every resource pertaining to `void:Dataset` or `dcat:Dataset`. The metric can be formally defined as follows:

$$P1(D) := \frac{\sum_{d \in \overline{ds}(D)} basic(d)}{\overline{ds}(D)}$$

where $\overline{ds}(D)$ is the set of resources having a type of `void:Dataset` or `dcat:Dataset` in the assessed dataset D, whilst $basic(d)$ is a function that returns '1' if $d \in \overline{ds}$ has a triple corresponding to **subject** $\times$ (**dc:creator** || **dc:publisher**) $\times$ **object**.

Results Overview: A box plot with the quality values for the contextual dimension metric is given in Figure 9.6. The box plot for this metric (P1) is, qualitative speaking, very negatively short, suggesting that most of the sampled datasets contain no basic provenance information in their VoID or DCAT metadata (when available). The median value is 0%. Nonetheless, this metric has a number of outliers, amounting to around 16.27% of the sample population. From this 16.27%, 71% of the datasets have a quality value of 100%. The σ_s value stands around 32.89%, whilst the mean is 12.78%.

Publishers might add basic provenance triples directly in a dataset rather than in the metadata, which is a drawback in terms of *"understand(ing) the meaning of data"* [105], as the provenance will be unknown to an automated agent looking for this information within the metadata before consuming the actual data. For example, `europeana.eu` attaches a `dc:creator` to every resource rather to some metadata. Hence, we encourage publishers to use dataset profiles for VoID and DCAT, such as DCAT-AP [26] and VoID Editor[27].

(P2) Traceability of the Data

In the Data on the Web Best Practices document, the editors note that

> *"consumers need to know the origin or history of the published data, [...], data published should include or link to provenance information"* – [105, §9.4]

Different publishers might contribute to the same dataset, by publishing within the same namespace. Therefore, it is important that consumers can track the origin of each piece of data/resource in a dataset. This provenance metadata can be described using the PROV-O ontology [73]. PROV-O allows the identification of agents, entities and activities. An agent represents the owner, or the responsible person for an activity or entity. An entity represents some aspect which is being modelled in form of linked data, for example weather information from Malta. An activity describes the process of creating Linked Data resources.

Metric Computation: This metric checks whether each resource has provenance information related to the origin of data. With regard to the quality metric survey in [160], this metric can be related to the "trustworthiness of statements (T1)". More specifically, this metric checks for entities with the following characteristics:

[26] `https://joinup.ec.europa.eu/asset/dcat_application_profile/description`. Date Accessed 10th October 2016

[27] `http://voideditor.cs.man.ac.uk`. List of other VoID editors and generators: `http://semanticweb.org/wiki/VoID.html#Generators_.26_Editors`. Date Accessed 10th October 2016

- Identification of an *agent* of an *entity* (quads having a predicate `prov:wasAttributedTo`);
- Identification of *activities* in an *entity* (quads having a predicate `prov:wasGeneratedBy`);
  1. Identification of a *data source* in an *activity* (quads having a predicate `prov:used`);
  2. Identification of an *agent* in an *activity* (quads having a predicate `prov:wasAssociatedWith` and/or `prov:actedOnBehalfOf`);

In order to avoid bias, an agent and an activity in an entity are both given a weight of 0.5. Similarly, data source and agent (in an activity) are also given a weight of 0.5. Then, the metric can be computed as follows:

$$P2(D) := \frac{\sum_{e \in prov(D)} val(e)}{size(prov(D))}$$

where $prov(D)$, is the set of entities as described above, whilst $val(e)$ is the quantified weighted value of the entity. The metric's value represents the ratio of the dataset's resources conformity to this metric.

Results Overview: Similar to Metric P1, this metric (P2 - Figure 9.6) is also very negatively short. Unlike Metric P1, the granularity level of the metadata in this case can even reach a triple level. This means that the size of the overall dataset can grow very large, therefore publishers might not be willing to trade-off size for better metadata coverage. In fact, we noticed that there is only one publisher (`270a.info` datasets) who creates such metadata to enable users to identify the origin of data. The overall median value is 0%. The σ_s value stands around 10.06%, whilst the mean is 2.17%.

The practice of tracking the origin of data is often ignored by data publishers, possibly for a myriad for reasons, such as the inflating the size of the dataset, or modelling issues. We suggest that publishers add provenance information on the activities undertaken when creating resources in their dataset, and possibly separating this metadata from the data itself by using named graphs.

(U1) Human Readable Labelling and Comments

Data on the Web is meant to be exposed to both humans and machines. Therefore, a human information consumer should be able to comprehend and understand the ambiguity of a Linked Data resource. Apart from human understandability, labels and comments can be used in various applications, such as keyword-based and natural-language based search [54]. A Linked Data application is dependent on labels and comments provided with each resource, as the application itself is not yet intelligent enough to try to map a resource to its real-world description. Labels can possibly be extracted from a human readable URI, e.g. extracting the fragment 'Malta' from `http://dbpedia.org/resource/Malta`.

Heath and Bizer suggest that predicates such as `rdfs:label`, `foaf:name`, `skos:prefLabel`, `dcterms:title`, should be used to label resources as they are widely supported by Linked Data applications, whilst `dcterms:description` and `rdfs:comment` should be used for a textual description of a resource [74]. Nevertheless, there are a number of vocabularies having terms to describe human readable labels and comments[28]. The authors in [54] study the usage of labels in the Web of Data[29], and reported the occurrence of the various predicates used for resource labelling. In terms of classification, according to [160] this metric is classified under the *understandability* dimension.

Metric Computation: The aim of this metric is to calculate a dataset completeness in terms of human-readable labels and descriptions. The metric measures the percentage of local entities that have a label or

[28] A simple search on LOV resulted into 346 terms for labels (12 of which tagged as W3C recommendations) and 150 terms for comments (1 being tagged as a W3C recommendation).

[29] The corpus used was the BTC2010 (`http://challenge.semanticweb.org/`)

a description. More specifically, each resource should have one (or more) of the following predicates, extracted from the top 50 vocabularies used in the LOD Cloud [143]:

- `rdfs:label`;
- `rdfs:comment`;
- `dcterms:title`;
- `dcterms:description`;
- `dcterms:alternative`;
- `skos:altLabel`;
- `skos:prefLabel`;
- `skos:note`;
- `powder-s:text`;
- `skosxl:altLabel`;
- `skosxl:hiddenLabel`;
- `skosxl:prefLabel`;
- `skosxl:literalForm`;
- `schema:name`;
- `schema:description`;
- `schema:alternateName`;
- `foaf:name` (for FOAF profiles).

A Linked Data resource is a *thing of interest*, or in a more practical sense, a set of triples that have the same subject URI. The metric can be computed as follows:

$$U1 := \frac{size(\{t \mid (\forall t \in \overline{ent} \cdot t.predicate \in desc)\})}{size(\overline{ent})}$$

where $\overline{ent}$ is the set of resources (i.e. triples with the same subject URI) in the assessed dataset D, t is a triple in $\overline{ent}$, and $desc$ is the set of predefined predicates that define a label or description. The metric's value represents the level of completeness of a dataset with regard to human-readable labels and descriptions.

Results Overview: The box plot for this quality metric (U1 - Figure 9.6) is relatively tall, covering the whole range of values, i.e. 100%. This suggests that data publishers follow varying practices with regard to human-readable labels and comments. The quality value is centred on 33.33% with a σ_s value of 40.93%, whilst the average quality value is 43.76%. The quality values of this metric provides the biggest variance against the rest of the contextual metrics. Moreover, around 29.29% of the assessed datasets have a completeness value of more than 90%, whilst in total around 43% of the datasets have a value of more than 50%. This metric is similar to the one presented in Hogan et al. [80, §5.3 – Issue XI]. Our assessment shows larger variation (σ_s value in [80, §5.3 – Issue XI] was 14.99%) in the quality result, the average value in our study increased by 28.76% when compared with the previous study conducted in 2012.

Whilst most of the publishers tend to attach labels and descriptions, other publishers might use other non de-facto schemas to describe resources in a human readable fashion. Overall, we can draw parallels between our assessment results and the results presented in [54], as both assessments show that the community needs to work harder to ensure the completeness of human readable labels and descriptions in Linked (Open) Datasets.

(U3) Presence of URI Regular Expression

One of the main purposes of the Web of Data is to be queried and explored. Structural metadata enables consumers to understand the underlying structure of a dataset. Having a regular expression defining the URI structure of a dataset enables agents to interpret resources better, for example, extracting fragments from URI resource such as local name, or query a dataset retrieving local resources according to the specified URI structure. The presence of URI regular expression metric is classified under the *understandability* dimension [160].

Metric Computation: This metric checks for the identification of a URI regular expression in the dataset's metadata, and can be quantified as follows:

$$U3(D) := \begin{cases} 1.0 \; if \; has \; pattern \\ 0.0 \; otherwise \end{cases}$$

where by *has pattern*, the metric is looking for a triple **subject** × **void:uriRegexPattern** × **object** in the assessed dataset

Results Overview: This metric reports 100% if the assessed dataset has a URI regular expression pattern defined. Our assessment showed that only 10 of the datasets had such an expression, giving a mean value of 7.75%, and a σ_s value of 26.84%. The box plot for the metric U3 in Figure 9.6, illustrates this negatively short quality indicator.

(U5) Indication of Used Vocabularies

Vocabularies play an important role in the structure of a dataset, since one or more of these vocabularies describe the dataset's resources. Similar to Metric U3, indicating the vocabularies used is part of the structural metadata of a dataset. Knowing the vocabularies used in a dataset, a human consumer can query the data. This metric is also classified under the *understandability* dimension [160].

Metric Computation: This metric checks whether vocabularies used in the datasets, either in the predicate position or in the object position if the predicate is `rdf:type`, are indicated in the dataset's metadata, specifically using the `void:vocabulary` predicate. The RDF, RDFS and OWL vocabularies are not taken into account in this metric. This metric value can be computed as follows:

$$U5 := \frac{size(vocabularies(D))}{size(\{ns(v) \mid v \in class(D) \cup prop(D)\})}$$

where $vocabularies(D)$ is the set of vocabularies, identified by the object in a triple **subject** × **void:vocabulary** × **object**. The metric's value represents the ratio of the defined vocabularies in the dataset's VoID description vs. the actual vocabularies used in a dataset, identified by the unique namespaces of the classes ($class(D)$) and properties ($prop(D)$).

Results Overview: Similar to most of the contextual metrics, the box plot for this metric (U5) is, also negatively very short, suggesting that most of the population datasets have no indication of the vocabularies used. Despite having a median value is 0%, this metric has a number of outliers, amounting to around 11% of the population dataset. These outliers pushed the σ_s value to 10.62%, whilst the mean is 2.71%.

From our assessment, around 2,800 different (not unique) vocabularies were used throughout the assessed dataset, whilst only 128 (around 4%) vocabularies where identified by the `void:vocabulary` predicate. Moreover, only 63 of those 128 defined vocabularies (around 63%, and around 2% of the total number of vocabularies used) where actually used in the dataset. This means that around 37% of the

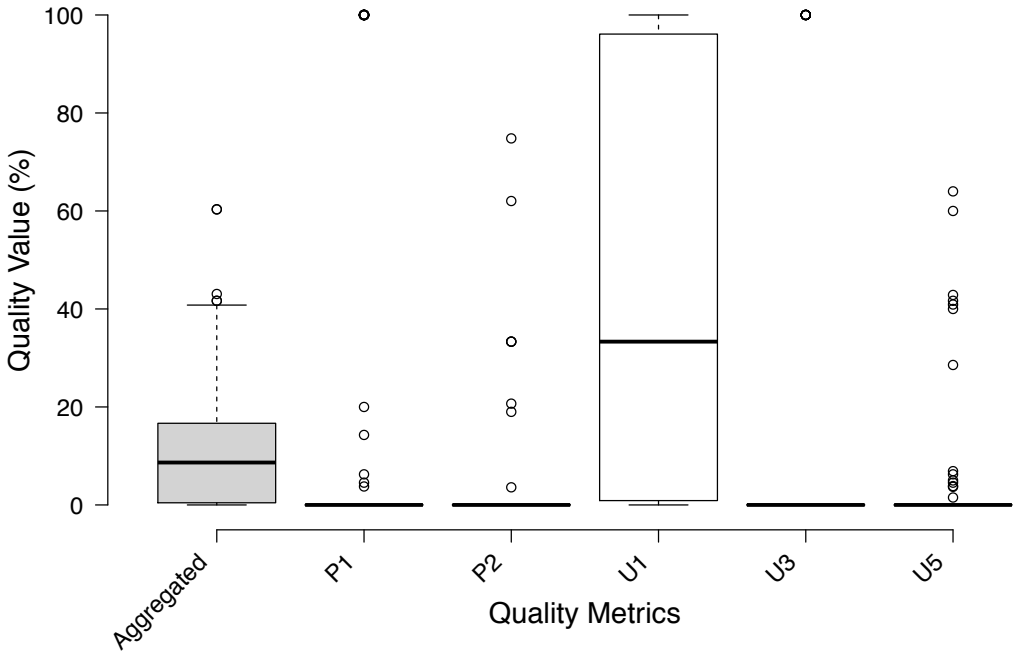

Figure 9.6: Center lines show the medians; box limits indicate the 25th and 75th percentiles; whiskers extend 1.5 times the interquartile range from the 25th and 75th percentiles, outliers are represented by dots.

defined vocabularies were not used in their respective datasets. Using VoID generators as part of their publishing methods (mentioned in Metric P1), such issues can be easily rectified by the publishers.

Aggregated Results

Table 9.5 shows the aggregated ranking of the five top and bottom datasets per category. Figure 9.6 shows a box plot illustration of the aggregated quality value compared with the category's metrics. The overall aggregated box plot shows a population population that is symmetrical, with a centrality median of 8.66%. The deviation value (σ_s) is 13.84%, which suggests a moderate distribution, whilst the average score is 13.04%. Five datasets from the whole population are "positive outliers" (since their overall quality value in this category is superior to rest of the population). These quality scores shed light on the real problems related to the contextual category. More worryingly is the fact that provenance information is not given the same importance as other quality metrics. Data consumers might look at such provenance information to make informed decisions on whether to trust a particular dataset or data publisher prior to using a dataset. Lacking such information might make it hard for data consumers to re-use and adopt some dataset.

9.3.4 Intrinsic Category

Defined as "independent of the user's context" [160], the intrinsic category quality indicators are related to assess *correctness* and *coherence* of the data. Zaveri et al. [160] classified metrics according to the following dimensions:

1. *syntactic validity* – the conformance of an RDF document vis-à-vis the standard specification;
2. *semantic accuracy* – the correctness degree of the represented values with regard to the real world;

Dataset	$v(C, 1.0)$	P1	P2	U1	U3	U5
http://bfs.270a.info/	60.35%	100%	74.80%	99.91%	0%	0%
http://lod.geospecies.org	60.29%	99.98%	0%	47.82%	100%	64%
http://statistics.data.gov.uk/	43.05%	0%	0%	98.33%	100%	60%
http://www.kupkb.org/	41.66%	100%	0%	100%	0%	0%
http://rod.eionet.europa.eu/	41.66%	100%	0%	100%	0%	0%
	...					
http://curriculum.rkbexplorer.com	0%	0%	0%	0%	0%	0%
http://extbi.lab.aau.dk/resource/Dataset	0%	0%	0%	0%	0%	0%
http://data.dcs.shef.ac.uk/	0%	0%	0%	0%	0%	0%
http://prefix.cc/	0%	0%	0%	0%	0%	0%
http://id.ndl.go.jp/auth/ndla	0%	0%	0%	0%	0%	0%

Table 9.5: Overall ranking of datasets for the contextual category.

3. *consistency* – the level of coherence in a dataset with respect to the knowledge it represents and inference mechanisms;
4. *conciseness* – the degree of redundancy in a dataset; and
5. *completeness* – the extent to which data is complete with respect to the real world.

In this section we assess metrics related to the *conciseness* dimension (the extensional conciseness metric), the *consistency* dimension metrics (seven in total), and one metric from the *syntactic validity* dimension. No metrics were assessed for the other two dimensions mentioned in [160], as they would have required a different experiment set up. For example, for the *completeness* dimensions, we would require to assess the datasets according to their domain.

(CN2) Extensional Conciseness Metric

In [30], Bleiholder and Naumann define a conciseness metric as *"measure(ing) the uniqueness of object representations"*. Undoubtedly, from a database point of view, data redundancy causes a dataset to be large. This issue might not be that significant anymore because of large storage devices, or distributed storage. However, data redundancy can be challenging in terms of data curation. For example, a data curator has to ensuring that all "replicated" resources are updated accordingly. However, data redundancy is not always a bad thing. For example, such redundancies can lead to improvements in query rewriting in Ontology-based Data Access, although it should be avoided if the publisher does not understand how to maximise its utility [157].

At the Linked Data level, a linked dataset is concise if there are no redundant instances [114]. By redundancy, Mendes et al. [114] explains that there are no two instances (locally) with different identifiers but with the same set of properties and corresponding data values. The extensional conciseness metric is classified under the conciseness dimension in [160].

Metric Computation: The extensional conciseness metric checks for redundant resources in the assessed dataset, and thus measures the number of unique instances found in the dataset. In Section 8.2.5, we showed that a naïve implementation of this metric results in large computational time, therefore we suggested the use of Bloom Filters [19] as an approximation technique. Using the bloom filter for identifying possible duplicate instances during the assessment process, we quantify this metric as:

$$CN2(D) := 1.0 - \frac{size(\{r \mid \forall r \in \overline{ent} \cdot isSet(hash(r)) == true\})}{size(\overline{ent})}$$

where, *hash* is a function that hashes the resource and *isSet* is the function that checks if the produced hash is already contained in the filter. *r* is a resource whose hash bits might have been set before, thus indicating a possible duplicate resource. In simple terms, the value returned by this quality metric describes the dataset's level of non-redundant entities. Further discussion on Bloom Filters and this metric can be found in Section 7.2.2.

Results Overview: Our assessment estimated that the assessed datasets had an average of 7.6% redundancy (the mean value is 92.40%) in total. Nevertheless, this does not mean that there is low redundancy on the whole Web of Data, since the sample standard deviation (σ_s) stands at 13.22% (median 99.34%), which suggests a moderately varied quality value overall. Although the box plot (see Figure 9.7) for this metric (CN3) is comparatively short, the outliers stretch the σ_s value. Around 13% of the datasets had a quality value less than the lower whisker, i.e. 78.55%. The range of quality values, including outliers, is 62.31%.

For this estimate value, we used 13 filters with a size of 5,500,000, ensuring efficient runtime with a low loss in precision (cf. Section 8.2.5). Around 76% of the datasets scored a value of 90% or more, meaning that the level of redundancy in these datasets is on the low side. Publishers should keep redundancy at a low level, and ensure that identical resources are not recurrent throughout the dataset. This can be done by creating `owl:sameAs` links between identical resources, without repeating property-value triples.

(CS1) Entities as Members of Disjoint Classes

The Web Ontology Language (OWL) extends the RDFS expressivity by modelling primitives that are otherwise difficult to express in the traditional RDFS. Generally, the OWL axioms deal with restrictions that can be placed on an otherwise open world assumption. On the other hand, incorrect usage of OWL features results in inconsistencies and thus jeopardizes reasoning.

The `owl:disjointWith` property is used to *"guarantee(s) that an individual that is a member of one class cannot simultaneously be an instance of a specified other class"* [144, §5.3]. One of the most popular examples of disjoint classes can be found in the FOAF vocabulary, where `foaf:Person` and `foaf:Document` are defined disjoint, which means that the resource `John` (as an example) cannot be both a person and a document. This metric is classified under the consistency dimension in [160].

Metric Computation: Metric CS1 checks for disjointness between types in multi-typed resources. Moreover, each assessed explicit type is inferred in order to check disjointness also between parent classes. Along these lines we quantify this metric as follows:

$$CS\,1(D) := 1.0 - \frac{\sum_{r \in \overline{ent}} hasDisjointTypes(r)}{size(\overline{ent})}$$

$$hasDisjointTypes(r) := \begin{cases} 1.0 \; size(\{(pInf(t_i) \setminus pInf(t_j)) \mid \forall t \in types(r)\}) > 0 \\ 0.0 \; otherwise \end{cases}$$

where $pInf(t)$ is the set containing the disjoint members of t ($\mathbf{t} \times$ **owl:disjointWith** $\times \bar{\mathbf{t}}$) and the disjoint members of the parent members of t ($\mathbf{t} \times$ **rdfs:subClassOf*** $\times \dot{\mathbf{t}}$), and $types(r)$ is the set of the types a resource is a member of ($\mathbf{r} \times$ **rdf:type** $\times \mathbf{t}$). The metric value indicates the degree of disjoint entities used within resources in the assessed dataset.

Results Overview: The assessment shows that almost all of the assessed datasets observe the `owl:disjointWith` property and their entities do not violate this property's restriction. In total around 98% of the datasets score a quality value of 100 for this metric, whilst the other two datasets

score a value of more than 99.9%, therefore still considered as of high quality. The average quality value for this metric is 100%, whilst the standard sample deviation (σ_s) is 0% (median is 100). The box plot CS1 in Figure 9.7 shows that there is no variation in the quality value of this metric, with the quartile ranges having the same value. Such low values in OWL inconsistencies were also reported in [82], were the authors attribute inconsistency problems caused by various incompatible exporters, such as FOAF exporters.

(CS2) Misplaced Classes or Properties Metric

RDF Schema provides property-centric mechanisms for defining classes (`rdfs:Class`) and properties (`rdf:Property`) in vocabularies [33]. This means that:

> *"instead of defining a class in terms of the properties its instances may have, RDF Schema describes properties in terms of the classes of resource to which they apply."* - [33, §2]

OWL has its own class axiom (`owl:Class`), which implicitly is a subclass of its RDF Schema counterpart. The schema has specialised property axioms (`owl:DatatypeProperty` and `owl:ObjectProperty` amongst others) that extend the `rdf:Property`, in order to (1) distinguish between the supposed values of the property, (2) enforce property-value constraints, and (3) describe logical characteristics of a property (cf. Metric CS3).

The RDF data model is represented by a *triple form* (**subject**×**predicate**×**object**), where the predicate is expected to be a property that describes a resource in the subject position and its value in the object position. On the other hand, a class URI defining a resource is usually in the object position when `rdf:type` is in the predicate position. The RDF data model is flexible allowing *any* resource URI to be in the predicate position. Therefore, whilst in OWL this practice it is prohibited (unless OWL 2 punning is used), the data model does not prohibit publishers to have a defined class in the *predicate* position and a property in the *object* position, but this could cause problems when agents are interpreting the data. Nonetheless, there are two OWL axioms, `owl:equivalentProperty` and `owl:inverseOf` that require a property to be in the *object* position. Therefore, triples with these two properties as predicates should be excluded from the assessment. This metric is classified under the consistency dimension in [160].

Metric Computation: The misplaced classes or properties metric assesses the datasets' statements in order to check the correct usage of classes and properties. More specifically, this quality indicator checks if the assessed dataset has defined classes placed in the triple's predicate and defined properties in the object position. We quantify this metric as follows:

$$CS2(D) := 1.0 - \frac{size(\{c \mid \forall c \in class(D) \cdot c \in V_p\}) + size(\{p \mid \forall p \in prop(D) \cdot p \in V_c\})}{size(quads(D))}$$

In other terms, this metric is checking the existence of class c in the set of property V_p (as defined in Metric IN3), which would mean that c is wrongly placed as a resource type, and similarly for property p. A high value of this metric is interpreted as conformance to usage of classes and properties in a dataset.

Results Overview: The usage of classes as properties and vice-versa are not common in the assessed datasets. Overall, 83% of the datasets score a value of 100% whilst the rest score 99.99%. The σ_s value for this metric is 0.01% (median 100%), which shows a very low deviation, whilst the average is 99.99%. The range value is 0.09%. Upon further inspection, we saw that no properties were used in the object position of an `rdf:type` triple, although classes such as `http://creativecommons.org/ns#License` were used infrequently (two instances in this case) as properties. Figure 9.7 shows the box plot for this metric CS2.

(CS3) Misused OWL Datatype or Object Properties Metric

OWL differentiates between properties referring to individuals (`owl:ObjectProperty`) and properties referring to data values (`owl:DatatypeProperty`). Incorrect usage of properties in this regard might lead to inapt functioning of an agent, for example, if a Linked Data viewer is using `owl:ObjectProperty` and `owl:DatatypeProperty` characteristics in order to hyperlink properties or not. Zaveri et al. [160] classify this metric under consistency.

Metric Computation: This quality indicator assesses a dataset's statements for the correct usage of the predicate in terms the `owl:DatatypeProperty` and `owl:ObjectProperty` axioms. Therefore, this metric detects "erroneous" triples where a data value (literal) object is attached to an `owl:ObjectProperty`, and an entity (individual) to an `owl:DatatypeProperty`. Following this description, the metric can be formalised as follows:

$$CS\,3(D) := 1.0 - \frac{size(\{t \mid \forall t \in D \ \cdot \ misusedOWL(t)\})}{size(quads(D))}$$

$$misusedOWL(t) :=$$
$$(isLiteral(t.object) \ \wedge \ isOP(t.predicate)) \vee$$
$$(isIndividual(t.object) \wedge isDP(t.predicate))$$

where *isLiteral* is a function that returns *true* if the assessed triple's object is a literal (i.e data value), *isIndividual* is a function that returns *true* if the assessed triple's object is a URI or a blank node, *isOP* and *isDP* are functions that check if the assessed triple's predicate is an `owl:ObjectProperty` or `owl:DatatypeProperty` respectively. A high value of this metric indicates a low amount of (or no) misused properties.

Results Overview: Figure 9.7 shows the box plot for this metric CS3. Similar to the previously discussed metrics for this dimension, the datasets adhere to a high quality score (average 98.88%) and a considerably low deviation (σ_s) value of 5.17% (median 100%). Overall, around 87% of the datasets scored 100% whilst in total 95% of the datasets scored 90% or higher. Nonetheless, the box plot shows that around 12% of the assessed datasets are outliers, having a value lower than 100%, which is less than the box plot lower whisker.

From our assessment the following datatype properties (top five) were used with resources:

- http://swrc.ontoware.org/ontology#series (28,269 times)
- http://swrc.ontoware.org/ontology#journal (21,731 times)
- http://reegle.info/schema#sector (1,876 times)
- http://rdf.myexperiment.org/ontologies/components/link-datatype (502 times)
- http://eunis.eea.europa.eu/rdf/species-schema.rdf#sameSpeciesRedlist (4 times)

whilst the following are object properties (top five) with literals:

- http://www.europeana.eu/schemas/edm/collectionName (50,000 times)
- http://lexvo.org/ontology#represents (49,966 times)
- http://xmlns.com/foaf/0.1/based_near (45,233 times)
- http://vivoweb.org/ontology/core#dateTime (25,538 times)
- http://purl.org/NET/c4dm/event.owl#place (7,952 times)

(CS4) Usage of Deprecated Classes or Properties Metric

Removing classes and properties from schemas renders data using these incoherent. OWL introduces the two classes `owl:DeprecatedClass` and `owl:DeprecatedProperty` for such situations. These two properties indicate that a class or property that belongs to these, are no longer recommended to be used in published data. This metric is classified under the consistency dimension in [160].

Metric Computation: This metric assesses a dataset to check if depreciated terms are used. More specifically, all used classes and properties are checked if they are members of `owl:DeprecatedClass` or `owl:DeprecatedProperty` respectively:

$$CS4(D) := 1.0 - \frac{size(\{c \mid \forall c \in class(D) \cdot c \in V_c \wedge dc(c)\} \cup \{p \mid \forall p \in prop(D) \cdot p \in V_p \wedge dp(p)\})}{size(class(D) \cup prop(D))}$$

where $dc(c)$ is a function that returns true if the class c member of (**c** × **rdf:type** × **owl:DeprecatedClass**), whilst $dp(p)$ is a function that returns true if the property p member of (**c** × **rdf:type** × **owl:DeprecatedProperty**). The metric's value calculates the ratio of used deprecated classes and properties against all used classes and properties.

Results Overview: With around 97% of the datasets scoring a quality value of 100%, data publishers tend to avoid using deprecated classes and properties. The LOD Cloud sample that was assessed used the minimal deprecated terms in most cases, with the lowest quality score of 97.41% marked as an outlier in the box plot (CS4) in Figure 9.7. The deviation (σ_s), as in the other consistency metrics, is very low (0.23%) with the median being 100. The overall average is 99.97%.

(CS5) Valid Usage of the Inverse Functional Property Metric

In the real world, an encryption public key is unique to every individual. If we want to represent this public key in a Linked Data document, then there should be one exactly one resource (possibly and individual of the type `foaf:Agent`) describing this public key, in order to represent this uniqueness between the key and the individual. Such properties are termed as *inverse functional*, meaning that if two different resources share the same value for that property, during reasoning or smushing[30] these two resources are treated as the same. The OWL schema provides a term `owl:InverseFunctionalProperty`, in which a vocabulary property with the above described semantics should be member of. Common examples of such properties include `foaf:mbox` and `foaf:homepage`. This metric is classified under the consistency dimension in [160].

Metric Computation: This quality indicator checks for incoherent values within the assessed dataset's values. More specifically, this metric checks if a value attached to a property member of `owl:InverseFunctionalProperty` (IFP) is shared by two or more **different** resources. In this metric, we only consider those statements with an inverse functional property. We quantify this metric as follows:

$$CS5(D) := 1.0 - \frac{size(\{t, \bar{t} \mid \forall t \in quads(D) \cdot ifp(t.predicate) \wedge \varphi(t, \bar{t})\})}{size(\{t.predicate \mid \forall t \in quads(D) \cdot ifp(t.predicate)\})}$$

$$\varphi(t, \bar{t}) := \begin{cases} true\ if\ (t.sb \neq \bar{t}.sb) \wedge (t.pr = \bar{t}.pr) \wedge (t.ob = \bar{t}.ob) \\ false\ otherwise \end{cases}$$

where $ifp(t.predicate)$ is a function that checks if a term is a member of (**t.predicate** × **rdf:type** ×

[30] This term is often used to name the process of aggregating resources based on inverse functional properties (https://www.w3.org/wiki/RdfSmushing).

owl:InverseFunctionalProperty), $\varphi(t, \bar{t})$ is a function that returns true if a triple t and a previously seen triple $\bar{t}$ are violating the IFP functionality. Therefore, the metric value is a ratio between the number of violating IFP triples, against the number of statements having an IFP predicate.

Results Overview: The box plot for this metric (CS5) in Figure 9.7 shows the trend in this metric where a large part of the assessed datasets are have no varying quality, bar a few number of datasets that are considered as outliers. These outliers, around 18% of the assessed datasets, increased the σ_s value to 12.29%, whilst the calculated median is 100%.

One should keep in mind that not all datasets assessed made use of inverse functional properties and were given a 100% score (since there was no triple breaking the IFP constraint), nevertheless, these where included in the assessment. From the assessment, around 3% of the datasets got a quality score of less than 50%.

Triples with the following IFP properties (top 5) where singled out in the assessment:

- http://xmlns.com/foaf/0.1/homepage (violated in 2861 triples)
- http://rdf.myexperiment.org/ontologies/base/has-friendship (violated in 635 triples)
- http://eunis.eea.europa.eu/rdf/species-schema.rdf#sameSynonymGBIF (violated in 380 triples)
- http://eunis.eea.europa.eu/rdf/species-schema.rdf#sameSynonymITIS (violated in 328 triples)
- http://eunis.eea.europa.eu/rdf/species-schema.rdf#sameSynonymFaEu (violated in 215 triples)

Since each dataset is assessed individually, our assessment did not point out possible IFP violations across the assessed datasets. In order to ensure that the IFP constraint is not violated, data publishers should ensure that data values (such a email address, homepage) are validated for uniqueness before publishing, possibly across the Web of Data and not just locally in the dataset.

(CS6) Ontology Hijacking Metric

In [79], the term ontology hijacking was described as the *"re-definition or extension of a definition of a legacy concept [. . .] in a non-authoritative source"*. An authoritative source s for concept c means that the namespace of c coincides with that of s. In simple terms, `http://xmlns.com/foaf/0.1/` is the authoritative source for the concept `foaf:Person`. Nevertheless, ontology hijacking can be seen as restricting the Linked Data idea of open world assumption, in a sense that such terms restricts what one can say about some concept. On the other hand, ontology hijacking may lead to incorrect inferencing throughout the data [79]. Zaveri et al. [160] classifies this metric under consistency.

Metric Computation: This metric assesses a dataset for its redefinition of third party external classes and properties. More specifically, this metric identifies if a dataset is the authoritative document for defining a class or property, following the axioms identified in [79]. The hijacking rules (axioms - triple position for authoritative document) are:

- `rdfs:subClassOf` - subject;
- `owl:equivalentClass` - subject or object;
- `rdfs:subPropertyOf` - subject;
- `owl:equivalentProperty` - subject or object;
- `owl:inverseOf` - subject or object;
- `rdfs:domain` - subject;
- `rdfs:range` - subject;
- `owl:SymmetricProperty` - subject;
- `owl:onProperty` - object;

- `owl:hasValue` - subject;
- `owl:unionOf` - object;
- `owl:intersectionOf` - subject or object;
- `owl:FunctionalProperty` - subject;
- `owl:InverseFunctionalProperty` - subject;
- `owl:TransitiveProperty` - subject.

This metric analyse defined classes and properties in a dataset by checking if these definitions are violating the hijacking rules. Along these lines, we quantify the metric as follows:

$$CS6(D) := 1.0 - \frac{size(\{t \mid \forall t \in tdef(D) \cdot \mathcal{H}(t)\}}{size(tdef(D))}$$

where $tdef(D)$ is the set of triples in dataset D having one of the hijacking rules axioms in its predicate or object position, and $\mathcal{H}(t)$ is a function that checks if triple t is violating one of the hijacking rules. Therefore, the value of this metric illustrates the percentage of triples that have some form of ontology hijacking, against all possible ontology hijacking triples.

Results Overview: Similar to the Metric CS5, the variation in quality within most of the assessed datasets ($\approx$ 86% of the datasets) is very low, though due to a number of outliers (shown in Figure 9.7 Metric CS6), the standard deviation value (σ_s) stands around 19.99% (median is 100%). Furthermore, the mean value is 93.64%. Overall, publishers tend to avoid redefining terms that they are not authoritative to do so, with around 85% scoring a quality value of 100%. In general, publishers should try to avoid redefining terms but instead they should extend existing terms (if needed), thus avoiding the confusion that can be caused by term cross-definition.

(CS9) Usage of Incorrect Domain or Range Datatypes Metric

In a schema, a property can optionally have a domain and range types defined. These definitions determine in what class type a resource should be used (the domain) and what is the expected type for its value. Similar to the most metrics defined in this section, using incorrect domain and range datatypes would not break the RDF data model. Nevertheless, it makes the data incoherent, as consumers who know the underlying schemas could query the data without looking at it, making it harder to retrieve the right or all results. Zaveri et al. [160] classifies this metric under consistency

Metric Computation: This metric assesses a dataset for the type validity of the domain and range of its statements, according to the schema of the predicate used. In particular, the predicate of each triple is dereferenced where the domain and range types were extracted, together with the types' inferred parent types. Following that, the subject and the object resource types are checked against the domain and range types for the particular property. We quantify this metric as follows:

$$CS9(D) := 1.0 - \frac{size(\overline{dom}(D)) + size(\overline{ran}(D))}{size(\mathcal{R}) \times 2}$$
$$\overline{dom}(D) := \{t \mid \forall t \in \mathcal{R} \cdot ((\mathcal{T}(t.s) \cap dom(t.p)) = 0)\}$$
$$\overline{ran}(D) := \{t \mid \forall t \in \mathcal{R} \cdot ((\mathcal{T}(t.o) \cap ran(t.p)) = 0)\}$$

where $\mathcal{R}$ is a set of sampled (which sample can be as big as the dataset under assessment) triples from the assessed dataset D (i.e. $\mathcal{R} \subseteq D$), $\mathcal{T}(r)$ is a function that returns the type of the local resource[31] r,

[31] External resources are ignored as we assume a closed world during the assessment. Thus, only resources with locally defined

the functions *ran*(*p*) and *dom*(*p*) return a set of range and domain types respectively for the predicate *p* together with their inferred parents. The metric value is a ratio between the total number of incorrect domain/range datatypes in statements and the total number of items in the reservoir $\mathcal{R}$ multiplied by 2 - since we are assessing the predicate of a triple twice (once for its domain, and another for its range).

Results Overview: This metric is implemented as a probabilistic metric using the reservoir sampling, in a similar manner as explained in Section 2.3.1. Our assessment shows that data publishers tend to use the incorrect domain and range types in the triples. Around 4% of the assessed datasets had a quality score of 90% or more, with the highest score being 99.51%. On the other hand, around 13% of the datasets scored less than 50%. The average score for this metric is 60.11% whilst the standard deviation (σ_s) is around 13.43%. The box plot for Metric CS9 in Figure 9.7 is symmetrical with the median standing at 57.14%. It also depicts a set of outliers over the top whisker and one dataset marked as outlier under the bottom whisker. It is also lower than the rest of the consistency metrics (Metric CS1 to Metric CS6), suggesting that Linked Data publishers might be more laid-back with using the right datatypes when creating resource triples. Linked Data publishers should be aware of the domain and ranges of the properties used in their datasets by consulting with the relevant vocabularies. Furthermore, simple on-the-fly type checking scripts can be created and used throughout the publishing activities, inspecting for such schema-to-data inconsistencies.

Since this metric is an estimate metric, the bias of these results lie within the reservoir sampler data objects being assessed, which can be under-represented. On the other hand, in Chapter 7 we have shown that with the right parameters probabilistic approximation techniques can provide a good estimate quality value.

(SV3) Compatible Datatype Metric

Ranges with a data value (i.e literal) are usually constrained to be of a certain datatype, for example, a property `ex:age` would have an `xsd:integer`. Being an important component in the RDF data model, literals can represent infinitely anything, whilst the datatype attached to the value can be used to interpret the data concisely. In [17], the authors describe four benefits of having good quality literals including *efficient computation*. This means that having a canonical representation of the datatype ensures a unique representation of a literal across the Web of Data, and thus actions such as comparing two literals of the same type would be easy [17]. It is recommended that publishers add the datatype to the literals. This metric is classified under the syntactic validity dimension [160].

Metric Computation: This quality indicator assesses the lexical form of the data values against the data type attached with the literal itself. Consider `"10"^^xsd:integer`, the value `10` is what is known as the lexical form, whilst `xsd:integer` (translated to `http://www.w3.org/2001/XMLSchema#integer`) is its datatype. Along these lines we quantify this metric as follows:

$$SV3(D) := \frac{size(\{v \mid v \in lit_t(D) \ \wedge \ \vartheta(v_{lf}, v_{dt})\})}{size(lit_t(D))}$$

where $lit_t(D)$ is the set of all **typed** literals, $\vartheta(v_{lf}, v_{dt})$ is a function that checks the validity of the value's lexical form v_{lf} against the value's datatype v_{dt}. Untyped literals are ignored in this metric as they cannot be validated against an unknown datatype. Therefore, the value of the metric is a ratio between the number of correctly typed literals and the total number of typed literals in the assessed dataset *D*.

Results Overview: The box plot for metric SV4 in Figure 9.7 shows that most of the datasets assessed adhere to a 100% quality value, though there were also a number of datasets that scored less and thus are

types types are included.

Dataset	$v(C, 1.0)$	CN3	CS1	CS2	CS3	CS4	CS5	CS6	CS9	SV3
http://extbi.lab.aau.dk/resource/	99.55%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	90.63%	100.00%
http://fao.270a.info/	98.74%	99.97%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	73.83%	100.00%
http://worldbank.270a.info/	98.40%	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	66.75%	100.00%
http://uis.270a.info/	98.39%	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	66.19%	100.00%
http://imf.270a.info/	98.31%	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	64.68%	100.00%
				...						
http://citeseer.rkbexplorer.com/	57.69%	80.20%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	50.00%	N/A
http://lingweb.eva.mpg.de/ids/	57.55%	78.60%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	58.35%	N/A
http://acm.rkbexplorer.com/	56.09%	75.40%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	50.00%	N/A
http://jisc.rkbexplorer.com	54.92%	78.55%	100.00%	100.00%	52.60%	100.00%	99.90%	100.00%	50.96%	N/A
http://www.productontology.org	48.68%	49.90%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	73.07%	N/A

Table 9.6: Overall ranking of datasets for the Intrinsic category.

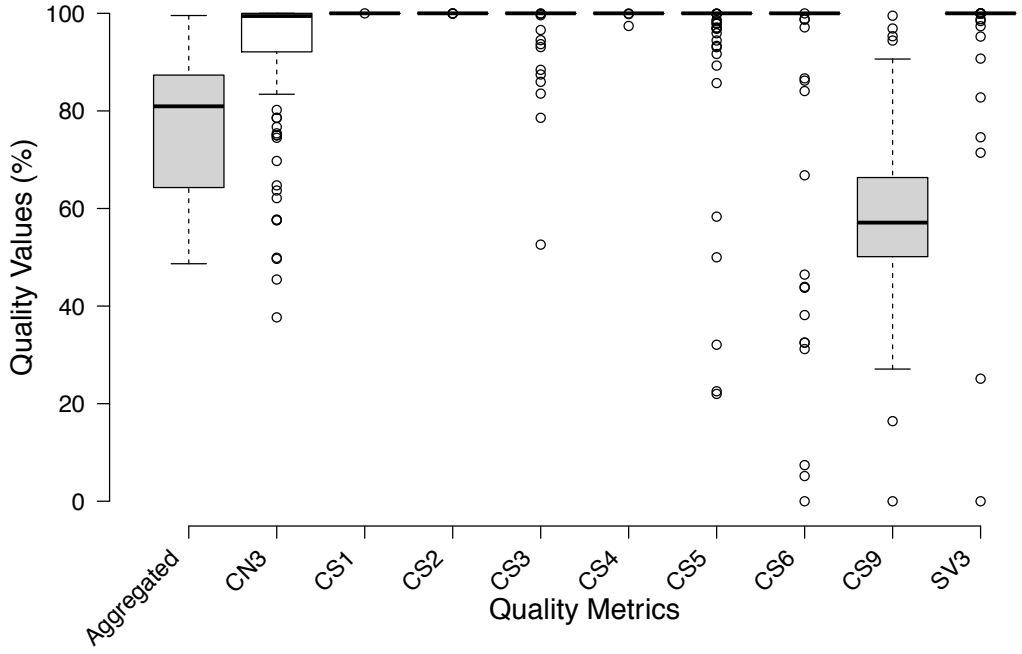

Figure 9.7: Center lines show the medians; box limits indicate the 25th and 75th percentiles; whiskers extend 1.5 times the interquartile range from the 25th and 75th percentiles, outliers are represented by dots.

marked as outliers. On average, the quality score of the assessed dataset is around 96.80% whilst the σ_s value is a high 14.16% (median 100%). Datasets that had no literal values were omitted from this assessment. In order to reduce incompatible datatypes vis-a-vis the lexical form of a data value, publishers could publish and serialise their data using the latest Turtle 1.1 parser, as it relaxes and simplifies the serialisation of such literals.

Aggregated Results

Table 9.6 shows the aggregated ranking of the top and bottom 5 datasets from the intrinsic category point of view. Figure 9.7 shows a box plot illustration of the aggregated quality value compared with the category's metrics. The overall aggregated box plot shows a population that is slightly varied having a σ_s value of 12.89% and a median of 80.94%. The majority of the metrics shows that a relative high quality (mean value of 77.36%) is adhered to by Linked Data publishers.

9.3.5 Accessibility Category

The accessibility category groups quality indicators related to the proper access functions of the Linked Data resources. The dimensions in this category deals with the ease of using Linked Data resources. In [160], Zaveri et al. classify metrics under the following dimensions: (i) *availability* - dealing with the access methods of the data; (ii) *licensing* - what are the permissions (if defined) to re-use a dataset; (iii) *interlinking* - the degree of internal and external interlinks between data sources; (iv) *security* - deals with the security and authenticity of datasets; (v) *performance* - how does the hosting servers affect the efficiency of a data consumer. In this section we assess metrics related to the *availability* dimension (2 metrics), *licensing* dimension (2 metrics), *interlinking* dimension (1 metric), and *performance* (2 metrics).

(A3) Dereferenceability of the URI

Dereferenceability is one of the main principles of Linked Data. HTTP URIs should be dereferenceable, i.e. HTTP clients should be able to retrieve the resources identified by the URI. We discuss this metric in more depth in Section 7.1.1.

Metric Computation: The aim of this metric is to check the number of valid deferencable URIs used (according to these LOD principles) in a data source. More specifically, an `HTTP GET` request is performed on a URI defining a concept, together with a header accepting a variety of Linked Data valid mime-types (e.g. application/rdf+xml, text/n3, text/turtle, etc. . .). A correct server-side dereferencing mechanism, should identify that the requested resource is an *abstract concept* and thus replies with a `303 See Other` and a redirect location where the *real-world object* (of the desired format) is. Heath and Bizer explain that *"where URIs identify real-world objects, it is essential to not confuse the objects themselves with the Web documents that describe them"* [74, §2.3.1].

This `303` redirection is handled automatically by the client, with the server responding with a `200 OK` together with the semantically described object in the requested format. This metric checks all local and non-local URIs for dereferencability. For this metric we use a stratified sampling approach as described in Section 7.2.1. Along these lines we adapt the metric from Hogan et al. [80, §5.1, Issue III]:

$$A3 := \frac{size(\{u \in \mathcal{R} \cap (dlc(D) \cap \mathcal{U}) \cdot deref(u) = true\})}{size(\mathcal{R})}$$

where $\mathcal{R}$ is the set of sampled URIs in the dataset, $\mathcal{U}$ is the set of URIs in the dataset D, and $deref(u)$ is a function returns *true* if the URI u being examined follows the dereferencability rules as described above. The metric's value represents the percentage of valid dereferenceable URIs in a dataset.

Results Overview: This metric was assessed using the stratified sampling technique described in Section 7.2.1 (Approach 2). The parameters used were 5000 as the global reservoir size (i.e. the number of possible different pay-level domains (PLD) in a dataset), and a PLD size of 10000. However, one must keep in mind that these parameters introduce a bias in our results in a way that the sample might be under-represented.

The box plot for this metric (A3) in Figure 9.8 shows a large varying quality with the box plot ranging all values from 100% to 0%. The average quality value of this metric is 36.86%, which is 33.44% lower than the average recorded in [80, §5.1 – Issue III]. There are two reasons for this difference. First, in our study we do not just study local dereferenceable URIs, but we also take into consideration the dereferenceability of external resources the publishers use. Secondly, we noticed that certain hosts blacklisted our IP address during this assessment following numerous HTTP requests. The box plot for metric A3 in Figure 9.8 is right skewed, meaning that the assessment shows a high concentration of low quality values. Similar to [80, §5.1 – Issue III], our assessment shows a high variability between

data producers on the dereferencability of resources. We report a σ_s value of 36.54%, with a median of 31.11%. In total our assessment attempted to dereference a total of 709,356 resources, out of which only 233,127 where valid dereferenceable resources. The rest of the resources resulted in the following problems:

- Hash URIs without parsable content - 5 resources;
- Status Code 200 - 61,922 resources;
- Status Code 301 - 7,281 resources;
- Status Code 302 - 13,878 resources;
- Status Code 303 without parsable content - 1,293 resources;
- Status Code 307 - 1 resource
- Status Code 4XX - 104,379 resources;
- Status Code 5XX - 5,444 resources;
- Failed Connection (either due to blacklisting or resource not online anymore) - 289,289 resources.

Surprisingly, not a lot of publishers abide by the dereferenceability guideline. Our assessment shows that only 33% of the assessed datasets have a dereferenceability value of 50% or more. Whilst this guideline is an one of the Linked Data principles, one should understand the extra costs this mechanism requires, including the maintenance of content-negotiation and re-direction schemes. However, one must investigate if the need of the dereferenceability mechanism is a must in Linked Data, or if agents can be adapted to understand Linked Data URIs automatically. In the meantime, a possible solution is that data publishers make use of Linked Data-based content management systems that handles such mechanisms automatically.

Licensing

> *"It is a common assumption that content and data made publicly available on the Web can be re-used at will. However, the absence of a licensing statement does not grant consumers the automatic right to use that content/data."* - [74, §4.3.3]

Licences, as defined by the Open Definition [58], are the heart of open data. It is the mechanism that defines whether third parties can re-use or otherwise, and to what extent. In Linked Open Data, one would expect that such licences are either machine-readable using predicates such as `dct:license`, `dct:rights` and `cc:licence`, or at most human-readable (e.g. within `dc:description`). Such license specification should also be included in a dataset's metadata.

(L1) Machine-Readable License

Having machine-readable license definitions (such as `http://purl.org/NET/rdflicense` [136]), agents would be able to consume (for example to visualise) different parts of the license, such as the jurisdiction and duties (e.g. share-alike, attribution, etc ...). Furthermore, agents would be able to understand the limitations of a license, and make informed decisions (e.g. if resources can be used within paid services) with less human interaction.

Metric Computation: The aim of this metric is to check if a dataset has a valid machine-readable license. By valid we mean that a license can be retrieved from a semantic resource (e.g. `http://purl.org/NET/rdflicense/.*`) with an `owl:sameAs` link to one of the following URLs:

- `http://(www.)?opendatacommons.org/licenses/odbl.*`

- http://(www.)?opendatacommons.org/licenses/pddl/.*
- http://(www.)?opendatacommons.org/licenses/by/.*
- http://creativecommons.org/publicdomain/zero/.*
- http://creativecommons.org/licenses/by/.*
- http://(www.)?gnu.org/licenses/.*
- http://creativecommons.org/licenses/by-sa/.*
- http://(www.)?.gnu.org/copyleft/.*
- http://creativecommons.org/licenses/by-nc/.*
- http://purl.org/NET/rdflicense/.*

These should be attached to a "license" predicate:

- dct:license;
- dct:rights;
- dc:rights;
- xhtml:license;
- cc:license;
- dc:licence;
- doap:license;
- schema:license.

We quantify this metric as follows:

$$L1(D) := \begin{cases} true\ if\ (lpr(t_p) \wedge lvld(t_o)) \\ false\ otherwise \end{cases}$$

where, $lpr(t_p)$ is a function that checks the triple's predicate against the set of defined license predicates, and $lvld(t_o)$ is a function that checks if the triple's object is a valid machine-readable license. This metric returns true if the assessed dataset has a valid machine-readable license.

Results Overview: In Section 9.1.2 we discuss the licences and rights in the LOD Cloud datasets' metadata. We show how around 41% of the whole LOD Cloud datasets have license or rights metadata, using the predicates dct:license, and dct:rights. In this metric we assessed the acquired data dumps and SPARQL endpoints for machine-readable licenses. However, our assessment resulted in just 17 datasets (≈ 13%) that contained at least one machine-readable license. Whilst we have to acknowledge that our data acquisition process did not take into consideration sources other than the LOD Cloud metadata (CKAN metadata was not included in the assessment), such **open** datasets should make this information explicit, as not all agents will have access to the LOD Cloud metadata. For example, dataset metadata can easily add machine-readable license statements by using other linked open datasets such as [136].

(L2) Human-Readable License

In contrast to Metric L1, a human-readable license enables human agents to read and understand a license in textual format, rather than in terms of triple statements.

Metric Computation: The aim of this metric is to verify whether a human-readable license text, stating the licensing model attributed to the dataset, has been provided as part of the dataset itself. The difference

from Metric L1 is that this metric looks for objects containing literal values and analyses the text searching licensing related terms. More specifically, we check for the following:

1. A license **description** triple, identified by a triple with a predicate `dct:description`, `rdfs:comment`, `rdfs:label`, or `schema:description` and a literal matching the following regular expression: `.*(licensed?|copyrighte?d?).*(under|grante?d?|rights?).*`;

2. A license triple, identified by a triple with a license predicate described in Metric L1, and a URI pointing to a human-readable documents (also defined in Metric L1).

We quantify this metric as follows:

$$L2(D) := \begin{cases} true\ if\ (t_p \in p_{hrdesc} \wedge lregex(t_o)) \\ false\ otherwise \end{cases}$$

where, p_{hrdesc} is the set of predicates representing human-readable descriptions, and *lregex* is a function that checks a literal against the defined license regular expression. This metric returns true if the assessed dataset has a valid human-readable license.

Results Overview: Similar to Metric L1, the assessment shows a low overall level of conformance to this metric. We detected human-readable licenses in 11 ($\approx$ 8.46%) datasets, 4 of which also had a machine-readable license. Whilst it is understandable that publishers are less inclined to have statements with large textual literals containing licensing data, we suggest that publishers should at least define the license name in the datasets' metadata. Licenses are of utmost importance to open data [58, §1], therefore, publishers should define the license or rights either as machine-readable (preferable) or at least human-readable.

(I1) Links to External Linked Data Providers

One of the main Linked Data principles is to *"include links to other URIs, so that they (referring to agents) can discover more things."* [22]. Furthermore, Berners-Lee states that linking your data to external sources would earn the dataset the fifth star (`http://5stardata.info`), given that the rest of the 4 guidelines are satisfied. Having external links in a dataset would enable data consumers to explore and understand better the data in question. Additionally, Heath and Bizer [74] describe the importance of external RDF links in the web of data since:

> *"they are the glue that connects data islands into a global, interconnected data space and as they enable applications to discover additional data sources in a follow-your-nose fashion."*
> – [74, §2.5]

These external outlinks is what makes the Linked Data ideology stands out from others. Well-interlinked data enables better analysis and understanding of the data. The interlinking property is often used in order to identify the importance or authority of a data source in the Web of Data. For example in [143], the interlinking degree is used to visualise the importance of datasets within the LOD Cloud. We discuss this metric in more depth in Section 7.1.2.

Metric Computation: The aim of this metric is to identify the total number of external RDF links used within the assessed dataset. An external link is identified if the object's resource URI in a triple has a PLD different than the assessed dataset's PLD. Furthermore, the external link should be a semantic resource

Dataset	I1(D)	# Unique PLDs
http://energy.psi.enakting.org	1402	1623
http://lobid.org/organisation	1395	1604
http://dbpedia.org/	32	346,708
http://vocabulary.semantic-web.at/PoolParty/wiki/semweb	13	291
http://lod.geospecies.org	11	42

Table 9.7: Top 5 ranked datasets for the links to external RDF data providers metric.

that can be dereferenced and parsed by an RDF parser. For this metric we use a reservoir sampling approach as described in Section 7.2.1. Along these lines, we quantify the metric as follows:

$$I1(D) := size(\{pld(u) \mid (u \in (dlc(D) \setminus ldlc(D)) \cap \mathcal{U}) \wedge isParseable(u) = true\})$$

where $pld(u)$ is a function that returns the pay-level domain of the resource's URI (u), $ldlc(D)$ is the set of local DLCs, and $\mathcal{U}$ is the set of URIs in dataset D. The value returned by this metric is the number of valid external RDF links the assessed dataset has.

Results Overview: Similar to Metric A3, this metric was assessed using a sampling technique. We used the reservoir sampling technique, where each external PLD has a sampler of maximum 25 items, as identified in the experiments in Section 7.3. Estimation techniques create a bias since the parameters might create an under-represented sample. In this case, we might miss out possible Linked Data documents that identify a PLD as external. Table 9.7 shows the top five assessed datasets, the number of unique dereferenceable external PLDs linked in the dataset, and the total number of unique PLDs. From the LOD Cloud dataset acquired sample, only 9 datasets had no external PLDs, whilst around 88% of the datasets had less than 50 unique external PLDs linked. In total, the number of external PLDs amounted to 977,609. Three datasets, namely `dbpedia.org`, `kent.zpr.fer.hr`, and `www.pokepedia.fr` accounted for around 97% of these PLDs. However, the actual number of dereferenceable PLDs is 3086, which is around 0.31% of the linked external PLDs.

If one considers the ratio of actual Linked Data external PLDs and the total possible external PLDs in a dataset, we found that 7 datasets resulted in 100%, whilst 36 datasets scored 50% or more. On average, the ratio of total possible external PLDs and actual Linked Data external PLDs is 27.71%, whilst the deviation (σ_s) value is 30.94% For example, the top two datasets scored 86.38% and 86.97% respectively, whilst for `dbpedia.org` the value was less than 0.01%.

Considering the Linked Data principles, one would have expected a higher ratio of external RDF links. However, there is no set number of external Linked Data PLDs each dataset should have. The assessed datasets provide a large deviation (σ_s) of 183.3 Linked Data PLDs, and an average of 27.01 Linked Data PLDs. Nevertheless, one should consider that these two statistical descriptions are highly influenced by the top two datasets. Data publishers are encouraged to use interlinking tools such as Silk [152], LIMES [121], or DHR [70], therefore ensuring that they abide by the Linked Data principles. Silk[32] is a flexible link discovery framework allowing users to define linkage heuristics using a declarative language. Similarly, LIMES[33] is a large-scale link discovery framework based on metric spaces. Unlike Silk and LIMES, in DHR [70] the links are discovered by closed frequent graphs and similarity matching.

[32] `http://silk-framework.com`. Date Accessed 10th October 2016
[33] `http://aksw.org/Projects/limes`. Date Accessed 10th October 2016

(PE2) High Throughput

Ideally, a Linked Data host can accommodate a large number of requests without affecting the consumers' productivity. That is, a consumer is not left waiting "in a queue" until other agents are served. Therefore, in an ideal situation, a host has the capacity to handle a large number of parallel requests.

Metric Computation: Adapting the metric from [57], the *high throughput* metric measures the efficiency with which a system can bind to the data source by measuring the number of HTTP requests answered by the source of the dataset per second. From the dataset we use reservoir sampling to "randomly" choose a maximum of 10 local resources (i.e. whose namespace is the same as the data source namespace) that will be used for this metric. The metric estimates the number of served requests per second, computed as the ratio between the total number of requests sent to the dataset's host. We quantify this metric, adopting [57] as follows:

$$PE2(D) := \begin{cases} 1.0 & \geq 5 \textit{ requests answered in} \leq 1s \\ \frac{servedRequestsPerSec}{200ms} & otherwise \end{cases}$$

where *servedRequestsPerSec* is number of requests that the host served per second. If five or more requests can be answered in a second or less, then the metric's value is defined as 100%, otherwise a percentage is calculated as the ratio of the number of served requests against the ideal time (200ms) taken to serve one request.

Results Overview: The box plot for this metric (PE2) in Figure 9.8 shows a large varying quality with the box plot ranging all values from 100% to 0%. The σ_s value stands around 45.60% (median value is 29.67%) whilst the mean value is 47.78%. The box plot is right skewed, suggesting that observations at the low end are concentrated. Around 38% of the assessed datasets gave a result of 100%, which means that more than 5 requests were answered in 1 second or less. Around 8.52% of the datasets scored a quality value between 50% (inclusive) and 100% (not inclusive). All quality results are dependent on the data host during the time of the assessment, therefore, such a quality metric should be performed more frequently.

(PE3) Low Latency

Latency is the amount of time an agent has to wait until the host responds with the particular request. The time taken largely depends also on how big the HTTP request is, and the number of HTTP round-trips the server has to make before serving the request. Therefore, the choice of Hash URIs and 303 redirects (i.e Slash URIs) is also an important factor for latency [57, 74]. Hash URIs would reduce the number of HTTP round-trips, as the document with the requested fragment resource description would have other resource descriptions in the same document. Therefore, the client would end up receiving unnecessary resources that would eventually increase the latency (since the document size will be larger). On the other hand, Slash URIs should have to do the whole dereferencing process, though the client will only receive the required resource. Ideally, the data source should serve resource requests with the lowest possible latency, which in turn means that data publishers should choose the right strategy for publishing data (Hash vs Slash).

Metric Computation: The *low latency* metric measures the efficiency with which a system can bind to the data source by measuring the delay between submitting a request for that very data source and receiving the respective response. Similar to Metric PE2, a reservoir sampler is used to sample a maximum of 10 local resources from the dataset under assessment. This metric is defined as the average time taken

for ten requests to respond, normalised to a percentage value between 0 and 100 by dividing by an ideal response time defined as one second [57]. Along these lines, Metric PE3 is quantified as follows:

$$PE3(D) := \begin{cases} 1.0 & \geq 1 \textit{ requests answered in} \leq 1s \\ \frac{1000ms}{averageResponseTime} & otherwise \end{cases}$$

where *averageResponseTime* is the average response time of the 10 sampled resources. A 100% low latency means that the data source can respond to a resource in a second or less, otherwise, the percentage value is calculated as a ratio of the number of possible requests served in one second.

Results Overview: Similar to Metric PE2, results of these metrics rely on the data host at time of assessment. The box plot for this metric (PE3) in Figure 9.8 confirms the large range varying quality, as in Metric PE2. The standard deviation value (σ_s) is 47.12% with a mean value of 57.55%. However, unlike PE2, the metric's values are left skewed, with a median value of 99.23%. This shows that there is a large concentration of very high quality values. Around 49.61% of the datasets have a quality value of 100%, meaning that at least 1 request is answered in 1 second or less.

Aggregated Results

Table 9.8 shows the aggregated ranking of the top and bottom 5 datasets from the accessibility category point of view. Figure 9.8 shows a box plot illustration of the aggregated quality value compared with the category's metrics. The overall aggregated box plot shows a population that is moderately varied having a σ_s value of 19.00% and a median of 29.96%. The box plot is skewed right, showing a large concentration of low quality values, with the average aggregated quality score being 33.12%, with only 19% of the assessed datasets scoring 50% or more. The aggregated value is affected by the low licenses metrics (L1 and L2), which is a concerning matter considering that the assessed datasets are part of the Linked **Open** Data cloud. Not having a defined license might make the adoption of linked dataset more difficult.

9.3.6 Ranking and Aggregation Remarks

All categories had an aggregated value $v(C, 1.0)$ calculated using the user-driven ranking function defined in Section 4.3.8, with a default weight of 1.0. In order to calculate a ranking for integer-based metrics

Dataset	$v(C, 1.0)$	A3	L1	L2	I1	PE2	PE3
http://fao.270a.info/	76.10%	66.82%	100.00%	0.00%	73.28%	100.00%	100.00%
http://frb.270a.info/	75.87%	64.29%	100.00%	0.00%	73.28%	100.00%	100.00%
http://ecb.270a.info/	75.82%	68.36%	0.00%	100.00%	73.28%	100.00%	100.00%
http://oecd.270a.info/	74.41%	51.9%	100.00%	0.00%	73.28%	100.00%	100.00%
http://uis.270a.info/	72.21%	35.26%	100.00%	0.00%	73.28%	100.00%	100.00%
...							
http://vocabulary.wolterskluwer.de/court	0.08%	0.60%	100.00%	0.00%	0.00%	11.23%	55.34%
http://www.lingvoj.org/	0.00%	-	0.00%	0.00%	0.00%	0.00%	0.00%
http://prefix.cc/	0.00%	-	0.00%	0.00%	0.00%	0.00%	0.00%
http://transport.data.gov.uk/	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
http://msc2010.org/mscwork/	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Table 9.8: Overall ranking of datasets for the accessibility category.

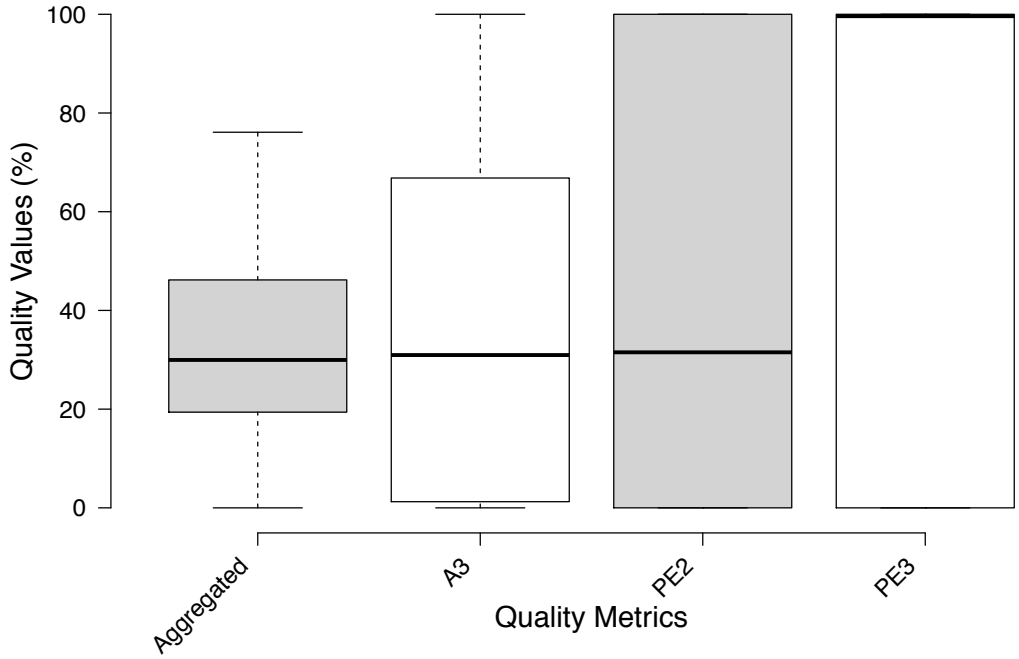

Figure 9.8: Center lines show the medians; box limits indicate the 25th and 75th percentiles; whiskers extend 1.5 times the interquartile range from the 25th and 75th percentiles, outliers are represented by dots. Machine-Readable License, Human-Readble License, and Links to External Data Providers metrics are excluded, but included in the aggregated result box plot.

(Metrics V1, V2 and I1), we followed a positional-based ranking, similar to as defined in [80]:

$$pb_x(D) := \frac{((size(\overline{D_x}) + 1) - pos_x(D)) \times 100}{size(\overline{D_x})}$$

where, x indicates the metric (e.g. V1), $\overline{D_x}$ is the set of datasets that where assessed for metric x, and pos_x is a function that returns the assigned position of dataset D following the assessment of metric x. All datasets were given a score based on a scale of 0 to 100%. In all cases 100% translates to the highest level of conformance to the quality metric being assessed, whilst 0% translates to the lowest level of conformance. The aggregated score for a dataset ($as(D)$) was calculated as follows:

$$as(D) := \frac{\sum_{m_{scr} \in \{RC1 \ldots PE3\}} m_{scr}(D)}{size(\{RC1 \ \ldots \ PE3\})}$$

where m_{scr} is the result of a dataset for a computed metric, and $\{RC1 \ \ldots \ PE3\}$ are the metrics described in this chapter. The aggregated scores only took into consideration the computed metrics. For example, in the case of the top placed dataset where all metrics were computed for the dataset, the average was taken over all 27 metrics. On the other hand, the second placed dataset was only available from a SPARQL endpoint which unfortunately did not manage to complete the evaluation (after a number of tries) due to various exception that we describe in the next subsection. Therefore, in that case, the aggregated score for those datasets was taken over 16 metrics. In Appendix D we present results for all datasets that were assessed. Nonetheless, we do not claim that lower ranked datasets are hosting poor quality data, but instead our claim is that following this study these datasets are less conformant to the quality metrics

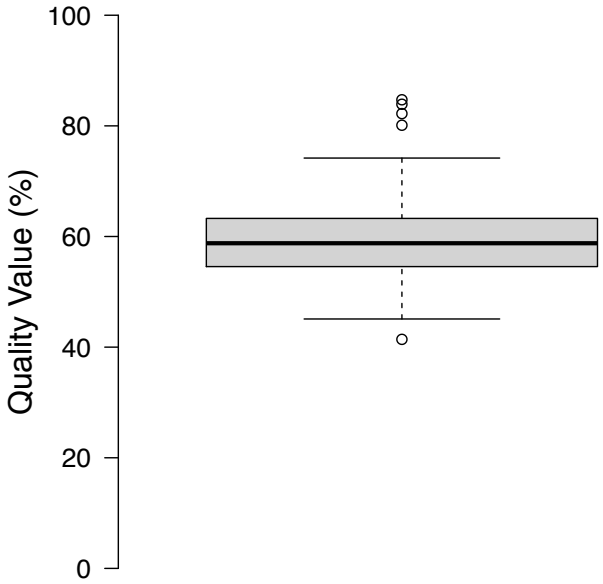

Figure 9.9: Center lines show the medians; box limits indicate the 25th and 75th percentiles; whiskers extend 1.5 times the interquartile range from the 25th and 75th percentiles, outliers are represented by dots. This box plot shows the aggregated conformance score.

assessed.

From a total of 239 datasets, only 130 datasets (totalling 3.7 billion quads were assessed. The average aggregated conformance score is 59.33% with a slight deviation (σ_s) of 7.63% (median value is 58.78%). Figure 9.9 depicts a symmetric box plot showing the spread of aggregated quality conformance scores. The box plot shows 5 outliers, four of which are "positive outliers", since their quality value is superior to the rest of the population.

Failing SPARQL endpoints

Most of the "failing" datasets are SPARQL endpoints, whilst others contained syntactic errors. In Luzzu, quality metrics are not written and executed on SPARQL endpoints, but instead triples are streamed from the endpoint[34] directly to the metric processors. In order to ensure that all triples are retrieved, the SPARQL processor makes use of the `ORDER BY` and `OFFSET` keyword, which takes much time to process especially on large knowledge bases. If the `ORDER BY` is removed, the endpoint responds faster, but since order is not guaranteed, multiple executions of the same query might result in different results. On the other hand, various endpoints have different settings, for example (i) (lack of) support of scrollable cursors – required for the query to stream triples; or (ii) different timeout settings (500 Server Error) – which might interrupt the assessment at a random point.

9.4 Is *this* Quality Metric Informative?

In this section we present a statistical analysis of the quality assessment, primarily understanding which of the quality metrics assessed can potentially give the stakeholders more information on the quality of linked datasets.

[34] This is the only query the Luzzu framework does on the endpoint, until all results are retrieved.

9.4.1 The Principal Component Analysis

The Principal Component Analysis (PCA) [128] is a statistical variable reduction technique that transforms a set of possibly correlated variables into a new set of uncorrelated components. Given some data, the PCA helps in finding the best possible characteristics to summarise the given data as well as possible. This is done by looking at the characteristics that provide the most variation across the data itself, ensuring that the data can be differentiated. On the other hand, the new set of uncorrelated components can be used to singularly describe correlated characteristics of the data. We will use the PCA in order to identify which of the assessed metrics are informative for Linked Data quality (cf. Section 9.4.2). This technique was favoured over ANOVA, which in simple terms is a technique usually used to determine whether there is significant difference between means. However, ANOVA was used in [13, 122] to identify the quality metrics that are sensitive in images, for example what are the best metric(s) that should be used for images with watermarks. Nevertheless, these statistical tests gives an indication, that ideally is sustained with a subjective test.

9.4.2 Identifying the Informative Quality Metrics for a Generic Linked Data Quality Assessment

The aim of this analysis is to study how informative are the quality metrics assessed on the Linked Open Data Cloud. Therefore, our main research question for this analysis is:

> **What are the key quality indicators that are defined in Zaveri et al. [160] and assessed during this empirical study that can give us sufficient information about a linked dataset's quality?**

Therefore, in this analysis, using PCA, we are looking at 27 different metrics in order to (1) reduce a number of quality metrics into a set of components that explain the variance of all quality values for all observations (linked datasets), and (2) possibly identify those metrics that are non-informative. The PCA will help us to find the best possible quality metrics that summarises the quality of linked datasets as well as possible, in terms of new characteristics (components). In doing so we group the quality metrics into a series of components, where each group means that the metrics in that component would have significant variance on describing the quality of Linked Data.

For this analysis we identify the following two hypotheses:

H_0: *No correlation exists among different metrics, thus each separate metric gives an informative value on the overall quality of a linked dataset.*

H_a: *Correlation exists among different metrics; therefore there are metrics that are non-informative to the overall quality value of a linked dataset.*

The null hypothesis (H_0) describes the scenario where all assessed metrics cannot be correlated and thus cannot be reduced to factors. We use the Kaiser-Meyer-Olkin Measure of Sampling Adequacy (KMO) to check whether Principal Component Analysis (PCA) is appropriate for our data, and Bartlett's Test of Sphericity to check whether the null hypothesis (H_0) can be rejected.

	Initial Eigenvalues			Extraction Sums of Squared Loadings			Rotation Sums of Squared Loadings		
Component	**Total**	**% of Variance**	**Cumulative %**	**Total**	**% of Variance**	**Cumulative %**	**Total**	**% of Variance**	**Cumulative %**
1	3.44	12.75	12.75	3.44	12.75	12.75	2.87	10.63	10.63
2	2.81	10.39	23.14	2.81	10.39	23.14	2.55	9.46	20.09
3	2.04	7.55	30.7	2.04	7.55	30.7	2.19	8.12	28.21
4	1.98	7.32	38.01	1.98	7.32	38.01	1.36	5.02	33.24
5	1.77	6.54	44.55	1.77	6.54	44.55	1.98	7.34	40.58
6	1.61	5.97	50.52	1.61	5.97	50.52	1.71	6.34	46.92
7	1.35	4.99	55.52	1.35	4.99	55.52	1.62	6	52.92
8	1.31	4.85	60.36	1.31	4.85	60.36	1.35	4.99	57.9
9	1.18	4.36	64.73	1.18	4.36	64.73	1.35	5.01	62.91
10	1.1	4.09	68.81	1.1	4.09	68.81	1.41	5.21	68.12
11	1.02	3.78	72.59	1.02	3.78	72.59	1.21	4.47	72.59
12	0.94	3.47	76.07						
13	0.88	3.27	79.34						
14	0.78	2.9	82.24						
15	0.72	2.68	84.92						
16	0.62	2.3	87.21						
17	0.58	2.14	89.35						
18	0.51	1.88	91.23						
19	0.48	1.77	92.99						
20	0.37	1.38	94.37						
21	0.34	1.26	95.63						
22	0.3	1.12	96.75						
23	0.26	0.97	97.72						
24	0.22	0.8	98.52						
25	0.16	0.61	99.13						
26	0.14	0.5	99.64						
27	0.1	0.36	100						

Table 9.10: Total variance explained.

Kaiser-Meyer-Olkin Measure of Sampling Adequacy		0.96
Bartlett's Test of Sphericity	Approx. Chi-Square	991.81
	df	351
	Sig.	0.000

Table 9.9: KMO and Bartlett's Tests.

In Table 9.9 we display the results for the KMO and Bartlett's test. The KMO results shows that our data has an adequacy of 0.96, which makes the factor analysis appropriate for our data. Kaiser recommends that values greater than 0.5 are acceptable [92]. The Bartlett's test gave a significance level of .000, that is, we can reject the null hypothesis (H_0) at $p < 0.05$, where p value is the significance level.

Following the rejection of the null hypothesis, we will use the Principal Component Analysis (PCA) in order to test the alternative hypothesis (H_a). Table 9.10 shows the total variance explained. In the Initial Eigenvalues column, the Table displays the eigenvalues associated with each component, and the total variance of the observed values for each factor. In simple terms, component 1 explains 12.75% of the total variance. Only components whose eigenvalues are greater than 1 are retained.

Therefore, the total number of factors extracted is 11. In order not to give too much importance to one component over another, a rotated component matrix (Table 9.11) is taken into consideration, in order to determine the informative quality metrics. The rotated component matrix is the main output following a Principal Component analysis. In total, these 10 factors can explain around 72.59% of the total variance. The other 16 components will only explain 27.14% of the variance.

In Table 9.11 we can see the 11 extracted components and the metrics each component represents. Each cell represents the correlation of a metric with a component. For the factor loading we use a cut-off

	Components										
	1	2	3	4	5	6	7	8	9	10	11
IO1	0.85										
IN3	0.76										
V1	0.72										
V2	0.69										
CS9											
P2		0.86									
P1		0.78									
L1		0.74									
U1		0.58									
I1											
PE3			0.92								
PE2			0.91								
A3											
CS4				0.83							
CS6				0.63							
U3					0.93						
U5					0.89						
RC2						0.85					
IN4						0.81					
L2							0.8				
CS1							-0.75				
RC1								0.68			
CS2								0.61			
CN2									0.77		
CS3									0.68		
SV3										0.79	
CS5											0.84

Table 9.11: Rotated component matrix.

point of 0.5[35] as the number of datasets is 130. This table also suggests which of the quality metrics, possibly combined (as in the case for components 1-9), are informative metrics.

By rejecting H_0, we are statistically confirming that most metrics on their own are not enough to provide an informative value on the quality of a dataset. Therefore, the PCA is used to create a descriptive summary of these metrics, which provides us with a number of components, thus proving our alternative hypothesis (H_a). Each component groups a number of quality metrics that defines an informative quality description. Recalling the main research question, the aim of this study is to highlight the key quality indicators that where classified in [160] and implemented in this empirical study. Therefore, for simplicity, we identify those metrics that are not in any of the 11 components as being metrics that describe the quality of a generic linked dataset in a non-informative manner. The PCA suggests that 3 metrics, namely Links to External Data Providers (Metric I1), Usage of Incorrect Domain or Range Datatypes (Metric CS9),and Dereferenceability (Metric A3), have values below the cut-off value for all of the 11 components.

Our initial quality assessment was generic, therefore all 130 datasets had the same 27 metrics assessed against them, irrelevantly if the metric is important to a particular dataset for a particular domain or not. Hence, the results obtained after performing the PCA are just an indication of which metrics might not be informative in a generic Linked Data quality assessment.

[35] Based on: http://imaging.mrc-cbu.cam.ac.uk/statswiki/FAQ/thresholds. Date Accessed 20th August 2016

9.5 Concluding Remarks

Quality issues in datasets have severe implications on consumers who rely on information from the Web of Data. Currently, it is difficult for a consumer to find datasets that fit their needs based on quality aspects. The semantic quality metadata produced by this empirical study fills this gap. Prospective users can now search, filter and rank datasets according to a number of quality criteria, and more easily discover the relevant, fit for use dataset according to their requirements. Nonetheless, such an assessment should not be done once, but it should be a continuous (or periodical) process to reflect the dynamic Web of Data.

Large-scale empirical studies on data quality can raise awareness on the current problems in data publishing. Such empirical analyses are important to the community as (1) they help to understand what are the current (or recurring) problems, and (2) define the future directions – in this case of Linked Data. In this chapter we quantified and analysed a number of linked datasets vis-a-vie a number of quality metrics as classified in [160]. Furthermore, in Section 9.4.2 we statistically analysed the quality scores and performed the Principal Component Analysis (PCA) test in order to identify the non-informative Linked Data quality metrics in a generic assessment. This statistical method shows that following our assessment 3 metrics were identified as non-informative to a datasets' quality. This chapter also serves as a demonstration of Luzzu's (cf. Chapter 4) applicability. This empirical survey is one of the largest (in terms of triples) evaluation of LOD data quality to date. All quality metadata produced in this empirical study is published using Linked Data principles at `https://w3id.org/lodquator`.

In Section 9.1, we explained the *Open Data* principles and using the LOD Cloud datasets metadata we performed a primary investigation in order to identify how well these abide by these principles. More specifically, we looked at the datasets' metadata in order to identify their accessibility points and licenses. We show that only around 42% had a valid Linked Data access point, whilst only 40% had a license.

In [77], Hitzler and Janowicz state that the general perception of Linked Data is that datasets are of poor quality. In line with research question 4 described in Section 1.3 we look at a number of datasets in order to understand better whether the perception label is deserved. In Section 9.3 we look at the datasets themselves in order to assess their quality against a number of metrics. We have seen that data publishers are compliant in various degrees with the different Linked Data best practices and guidelines with regard to the quality metrics. Overall, if we consider the bigger picture, that is the aggregated conformance score, we see that on average the Linked Data quality is slightly below 60% (highest value is 84.72% lowest value is 41.41%) with a low standard deviation value of 7.63%. Whilst the general perception might be derived from various different factors, the aggregated results from the generic assessment shows that this might not be the case. However, there is no known literature that scales quality scores, therefore we cannot say that the assessed linked datasets are of high or medium quality. When we talk about the aggregate conformance scores, a high performing metric compensates for a lower one. Therefore, when we look at individual metrics we see that there are certain aspects, more specifically quality metrics related to provenance and licenses, in which data publishers, collectively, should improve, as these are factors that can encourage Linked Data re-use. Nevertheless, this empirical study shows that there are still a number of problems related the Linked Data publishing and its conformance with a number of best practices and guidelines.

CHAPTER 10

Conclusions and Future Direction

In this thesis we formalise a conceptual methodology and framework for assessing the quality of linked datasets which then enables consumers to make educated decisions when searching for datasets in light of *fitness for use*. Following an introduction to the problem and discussion of the research that is to be undertaken (cf. Chapter 1), we described the proposed solution, and evaluated our conceptual methodology and framework against a number of Linked Open Datasets. In this chapter we conclude this thesis by discussing the results (cf. Section 10.1) in light of the research questions defined in Section 1.3. Finally we discuss the future work in Section 10.2.

10.1 Revisiting the Research Questions

In Section 1.3 we defined the core research question of this thesis which manifests the main goal of this document: *investigating and formalising ways for assessing Linked Data quality with the intent of allowing agents to choose fit for use datasets.* The main research question is defined as follows:

> *How can we assess (large-scale) linked datasets with regard to the fitness for a use case with the aim of enabling quality-driven selection according to various quality measures?*

In our thesis, we refined the core research question into four more specific research questions (cf. Section 1.3). The first research question (Chapters 4 and 5) explores means to assess the quality of linked datasets based on the user's choice of quality metrics. Furthermore, this question acts as the basis for the rest of the research questions. The second research question is about the representation of a dataset's quality that can be used by agents to filter and find datasets fitting their quality criteria. In Chapter 6 we investigate this question further, demonstrating the importance of having quality metadata attached to an assessed dataset. The third research question relates to the scalability of quality assessment, where together with the scalability of the framework described in Chapter 4 we investigate the possibility of having approximate quality values as a trade-off for faster assessment computation. Finally, research question four integrates the outcome of the previous three research questions to examine the state of a number of linked datasets available on the LOD Cloud.

In short, the main outcome of this thesis is a methodology and framework for assessing linked datasets, possibly using approximation techniques for faster metric computation. Furthermore, this framework

outputs interoperable quality metadata graphs for quality-based filtering and ranking. Consequently, we provide a set of graphs for different linked datasets ready to be consumed by agents.

RQ1: How can stakeholders be enabled to assess linked datasets' quality based on their choice of quality indicators?

In Part II we describe how stakeholders can be supported during the quality assessment of linked datasets. In Chapter 4 we define and formalise a conceptual methodology for assessing Linked Data quality. This methodology was based on a data quality life cycle which we re-defined to take co-evolution of linked datasets into consideration. The methodology was then implemented in a generic framework, Luzzu, that streams Linked Data triples from different sources, detects different quality problems, and presents interoperable results in form of quality problem reports and quality metadata. The Luzzu quality assessment framework was implemented keeping in mind the *extensibility*, *scalability*, *interoperability*, and *customisation* aspects. With extensibility, the Luzzu framework ensures that heterogenous linked datasets with different schemas can be assessed even if domain-specific quality metrics are required. The Luzzu framework expects that users identify the right quality measures prior to an assessment, with stakeholders either re-using existing quality metric implementations or defining their own. In this regard we developed a domain specific language (LQML), discussed in Chapter 5, to allow the definition of domain specific quality metrics, apart from allowing stakeholders to define them in traditional third generation languages. Luzzu is driven by a semantic backend where the to enhance the interoperability with other frameworks, such as cleaning tools. This enables the creation of pipelines that can support the quality life cycle in Figure 4.1. The scalability aspect is addressed by applying a streaming approach (triple by triple) to assess linked datasets. This aspect was evaluated and shows that the process scales linearly with the number of triples. Finally, the framework provides a ranking algorithm, allowing users to customise their filtering preferences to rank datasets based on quality indicators.

In Chapter 5, we tackle the challenge of defining and implementing quality metrics allowing stakeholders to choose their own quality indicators prior to an assessment. In this chapter our approach was to define a declarative domain specific language that focuses on the representation of quality metrics for linked datasets. Our DSL is aimed towards both Linked Data and non-Linked Data users, encouraging the quality assessment and hence the usage of linked datasets in a wider audience. Our heuristic assessment of the DSL, based on the cognitive dimensions of notation, outlines the strengths and weaknesses of our language. Nonetheless, we still need to assess the language's complexity with external stakeholders. The investigation of Luzzu and its components led us to a concrete open-source framework which enables stakeholders to assess linked datasets, based on their quality criteria. The framework acted as a basis to explore and unfold the next three research questions.

RQ2: How can we model and represent information about the quality of heterogeneous linked datasets to enable quality-driven dataset selection?

Having quality metadata would make dataset selection easier and hence possibly increasing the chance of re-using the dataset [105, §8.5]. A quality assessment framework should publish such metadata highlighting the quality indicators assessed, when the assessment was done (in order to check if the assessment is still relevant to the current version of the dataset), and provenance information of the assessor. In Chapter 6 (Part III) we introduce the Dataset Quality Vocabulary (daQ), a light-weight meta-model for representing the results of quality benchmarking of a linked dataset again as linked data, capturing the above mentioned requirements. The meta-model is based on three levels of abstraction: Category, Dimension and Metric, taking inspiration from the classification in [160]. The requirements are

modelled as part of the metric concept, where each dataset can be assessed over time against the same metric, each time recording a provenance-aware dated quality observation.

In this chapter we also discuss the importance of having different observations, for example in order to keep track of how quality evolved over time. Furthermore, we give a detailed example on different provenance aspects that can be attached to an observation, such as the estimate technique used and other profiling aspects that were explored during the assessment. Quality metadata can be visualised and queried on various cube dimensions and measures, in order to allow quality-driven ranking and filtering. Following our evaluation in Chapter 9, we made the quality metadata graphs available online at `http://w3id.org/lodquator`, where users can filter for assessed datasets according to their quality criteria. Moreover, the daQ has been adapted and used in a number of initiatives and development projects, including the adoption by the W3C Data on the Web Best Practices WG as a core module of their Data Quality Vocabulary [5].

RQ3: What techniques can be used to scale the assessment of large linked datasets?

As part of the solution for RQ1 we discuss the scalability of Luzzu as a solution to process large linked datasets. However, the metrics' implementation can affect the scalability of the quality assessment of these datasets. Certain metrics, albeit generally polynomial, can become intractable as the size of the dataset increases. In Part IV we look into a number of techniques, more specifically probabilistic techniques and distance-based outlier detection, to improve the running time of Linked Data quality metrics at the expense of getting an approximate quality value.

In Chapter 7 we look into sampling (reservoir and stratified), Bloom Filters, and clustering coefficient estimation and apply them to a number of quality metrics. Sampling techniques were applied to metrics that required network operations to complete. Such operations can be expensive to complete as they depend on factors such as the network's bandwidth and the data source's latency. Bloom Filters were used to possible duplicate resources in a dataset, whilst the clustering coefficient estimation was used to measure the neighbourhood density of a resource. These three techniques were applied to four different metrics; *dereferenceability* and *links to external data providers* were implemented using the two sampling approaches, the Bloom Filter technique was used for the *extensional conciseness*, and the *clustering coefficiency* metric was implemented using the clustering coefficient estimate technique. We evaluated these metrics over a number of datasets whose sizes vary from 75K up to 100M on a low-end machine. The results show that using probabilistic techniques the running time decreases considerably (for most metrics) whilst the loss of precision (the trade-off) in the quality value is not large. Considering that these experiments were executed on a machine with limited computational capabilities, in this chapter we show that quality assessment of large linked datasets can scale well with the usage of probabilistic techniques, performing the assessment within a reasonable time.

In Chapter 8 we look into distance-based outlier detection, to detect anomalies within Linked Data resources. We apply a method devised by Knorr et al. [97] to a Linked Data scenario, which creates clusters of data objects without any necessary supervision, thus reducing the user's interaction. The performed preliminary evaluation shows that the proposed approach is promising where the recorded precision is high. However, the approach records low recall, which means that the approach is not detecting a number of incorrect triples. We also show how an approximate quality value can be calculated with this techniques. This work is still in a preliminary stage and would require further investigation and experimentation before we can conclude that such technique is (1) better than the state of the art, and (2) if this technique can time-efficiently assess large linked datasets.

RQ4: What is the quality of existing Data on the Web?

After tackling the previous three research questions, we ended up with a quality assessment framework that is scalable and provides quality metadata following an assessment. In order to demonstrate this, we assessed a number of linked datasets for their quality. This assessment has a two-fold purpose: (1) demonstrate the applicability of Luzzu; and (2) tackle this research question. For this, we implemented a number of quality metrics that were classified in [160], some of which provided an approximate value.

In Chapter 9 we perform a primary investigation on the metadata of the datasets represented in the LOD Cloud. We check the openness factor of the datasets based on the metadata information available, gathering interesting insights on the access methods and the licenses of the portrayed datasets. However, this experiment was also a precursor to justify the choice of datasets that were in the end assessed for their quality. We then assessed the chosen datasets against a number of metrics across four different categories, and published the quality metadata for each dataset online following the Linked Data principles. For the representational category we saw that the quality varies moderately amongst the different datasets, whilst the average stood around 63%. The contextual category metrics fared poorly, with the most worrying metrics related to provenance. The average quality value, when all contextual metrics were combined together, stood around 13%. With regard to the intrinsic category, datasets were more compliant to principles and best practices. The deviation was similar to the other two categories, whilst the average score was around 77%. Finally, the average score for the accessibility category was around 33%, with the datasets' quality values largely varying for all metrics in this category. When we aggregated all metrics in one score for each dataset, the highest quality score was 84.72% and the lowest was 41.41%. The overall average score stands at 59.28% whilst the standard deviation (σ_s) value is 7.63%. The assessment shows that there are still a number of problems related to Linked Data publishing and its conformance with a number of best practices and guidelines. Furthermore, the quality metadata produced is published online using Linked Data principles with a two-fold aim: (1) data publishers can link their assessed datasets to the quality metadata; and (2) data consumers can filter datasets based on fitness for use.

As part of this research question we have identified a sub-question where we statistically identified the key quality indicators that describe a linked dataset's quality. Using the Principal Component Analysis (PCA) and the assessment results, we identified the metrics that possibly describe the quality of a linked dataset the best. Therefore, as part of this statistical analysis we also identified three quality metrics from all assessed metrics, namely Links to External Data Providers, Usage of Incorrect Data types, and Dereferenceability, as non-informative towards the description of a dataset's quality.

10.2 Future Work

Apart from the limitations to our approach that were discussed in the respective chapters, in this section we discuss the potential future direction and long term visions (marked as such) of this work, in an attempt to foster interest for addressing more specifically challenge 2 and 3 discussed in Section 1.1.

1. *Linked Data Quality as a Service.*
   Ideally, the assessment of Linked Data quality is not a one-off event. In this dynamic Web of Data, datasets are continuously changing. In Section 9.1.3 we discussed the dynamicity of the LOD Cloud, referring to the work by Käfer et al. [91]. They report that after a 29 week observation, 40% of the Web of Data either went offline or changed. These changes should be reflected in the quality metadata of the dataset as soon as possible. However, given the huge amount of data available, one should consider various strategies for crawling and assessing the data. Currently there are a number of crawlers available, such as the Dynamic Linked Data Observatory (DyLDO) [91] and

LOD Laundromat [18]. DyLDO collects weekly snapshots from the Web of Data and stores the crawl in quad files, where each triple would have the context from where it was crawled. Therefore, one would be able to re-create the dataset structure from a DyLDO snapshot and assign a quality metadata for a dataset. The LOD Laundromat crawls and cleans triples (removing blank nodes, syntax errors, and duplicate triples amongst others) from the Web of Data. Unlike DyLDO, a LOD laundromat crawl does not keep the context of the triples, and thus it would be more difficult to create quality metadata for a dataset from these triples. Since crawls might be large, one possible realisation of such a service could be done by using stratified sampling on DyLDO crawls to sample triples from datasets. However, if a thorough assessment is required, then one needs to figure out methods and techniques to continuously assess data quality in dynamic datasets (challenge 3);

2. *Continuous Quality Assessment of Linked Datasets.*
   Due to the co-evolution of linked datasets, there might be the need for re-assessing the quality of the dataset itself. Although changes might not be drastic (e.g. removing some triples, or some changes in literal values), the proposed methodology and framework has to re-assess the new version of the dataset from the beginning. Therefore, the proposed approach cannot just identify the dataset changes and thus increment (or decrement) the previous quality observation values as new observations. This problem is directly related to challenge 3 described in Section 1.1. Our methodology and framework can be adapted for this quality propagation by enhancing quality metadata observations with references to more information about the problems found in the current assessed version of the dataset. The new version can then be checked for changes (e.g. using diff) and then some process should check which metrics are affected by these changes. The challenge here is that a changed triple can affect multiple metrics differently.

3. *Choosing the right metrics for the right task.*
   One of the main challenges and yet one of the most important aspects of data quality is choosing the right indicators for the task at hand. For example, a data consumer finding a dataset for his question answering system might require different quality indicators than a consumer requiring a dataset for data mining. However, knowing this information requires additional analysis on the quality metrics that are used to assess a dataset, such as the Principal Component Analysis (PCA), which was used in Section 9.4 to identify informative quality metrics. In [146] we already look into a number of potential quality metrics to assess cross-domain datasets that can be used in a question answering system, though we have not yet analysed whether all of the chosen metrics are really suitable, or else if there are some metrics that are not relevant for the question answering domain. In order to adapt our framework to work as a recommender system for helping the user to choose the right metrics, a supervised algorithm should "learn" what configurations are used (during subsequent actual quality assessments) for the different tasks and use the quality results in order to build the statistical models. After some time, the system should recommend possible quality indicators for a specific task with a degree of confidence.

4. *Long-term Vision #1: Publishing Quality metadata as certificates on blockchains.*
   The quality metadata is aimed to be part of the assessed dataset as a named graph. However, one might doubt the legitimacy of such graphs or the quality values that are represented in the graph. Thus, we propose the use of blockchains as means for issuing quality metadata certificates for assessed datasets. Blockchains are secure decentralised "nodes" that discourage fraud and no node is more trustworthy than another. The current quality metadata graph, for example, can be encrypted using some technique as means for checksum. This encryption checksum number can then be used as part of the certificate that is stored in a blockchain. Concerned consumers can then

retrieve the quality metadata from the blockchain, which then are redirected to the actual quality metadata.

5. *Long-term Vision #2: Extending the framework and methodology to cover Open Data.*
The current framework is limited to Linked Data. However, there are still a number of open datasets in different formats (e.g. CSV). The framework can be extended by creating the relevant stream processors (e.g reading CSV line by line), though this has to be complemented with a number of quality metrics relevant for the different formats.

6. *Long-term Vision #3: Improving time efficiency for the Clustering Coefficient Estimation.*
In Chapter 7 we discussed the clustering coefficient estimation as a probabilistic technique to improve the running time. However, whilst we saw a significant difference against the naïve approach, this was still not enough for large datasets. We suggest that prior to the coefficient estimation technique, the assessed dataset is sampled in a similar manner as described in [102].

Bibliography

[1] H. Abelson et al., *ccREL: The Creative Commons Rights Expression Language*, Mar. 2008 (cit. on p. 129).

[2] M. Acosta et al., "Crowdsourcing Linked Data Quality Assessment", *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, ed. by H. Alani et al., vol. 8219, Lecture Notes in Computer Science, Springer, 2013 260, ISBN: 978-3-642-41337-7 (cit. on pp. 4, 30, 31, 121).

[3] H. Alani et al., eds., *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, vol. 8219, Lecture Notes in Computer Science, Springer, 2013, ISBN: 978-3-642-41337-7.

[4] R. Albertoni, M. D. Martino and P. Podestà, "A Linkset Quality Metric Measuring Multilingual Gain in SKOS Thesauri.", *LDQ@ESWC*, ed. by A. Rula et al., vol. 1376, CEUR Workshop Proceedings, CEUR-WS.org, 2015 (cit. on p. 85).

[5] R. Albertoni et al., *Data Quality Vocabulary (DQV)*, W3C Interest Group Note, World Wide Web Consortium (W3C), 13th May 2015, (visited on 03/08/2015) (cit. on pp. 8, 55, 81, 85, 126, 173).

[6] K. Alexander et al., "Describing Linked Datasets, On the Design and Usage of voiD, the Vocabulary of Interlinked Datasets", *Linked Data on the Web (LDOW)*, (Madrid, Spain, 20th Apr. 2009), ed. by C. Bizer et al., CEUR Workshop Proceedings 538, Aachen, Apr. 2009 (cit. on pp. 29, 71).

[7] K. Alexander et al., *Describing Linked Datasets with the VoID Vocabulary*, W3C Interest Group Note, World Wide Web Consortium, Mar. 2011 (cit. on pp. 29, 52, 126, 132, 140).

[8] C. B. Aranda et al., "SPARQL Web-Querying Infrastructure: Ready for Action?", *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, ed. by H. Alani et al., vol. 8219, Lecture Notes in Computer Science, Springer, 2013 277, ISBN: 978-3-642-41337-7 (cit. on p. 32).

[9] M. Arias et al., "HDT-it: Storing, Sharing and Visualizing Huge RDF Datasets", *ISWC*, 2011 23 (cit. on p. 43).

[10] A. Assaf, A. Senart and R. Troncy, "What's up LOD Cloud? Observing The State of Linked Open Data Cloud Metadata", *2nd Workshop on Linked Data Quality*, 2015 (cit. on pp. 32, 126).

[11] J. Attard et al., *A systematic review of open government data initiatives*, Government Information Quarterly **32**.4 (2015) 399, ISSN: 0740-624X (cit. on pp. 36, 126).

[12] S. Auer et al., "Managing the Life-Cycle of Linked Data with the LOD2 Stack", *The Semantic Web - ISWC 2012 - 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part II*, ed. by P. Cudré-Mauroux et al., vol. 7650, Lecture Notes in Computer Science, Springer, 2012 1, ISBN: 978-3-642-35172-3 (cit. on pp. 36, 55).

[13] I. Avcibas, B. Sankur and K. Sayood, *Statistical evaluation of image quality measures*, J. Electronic Imaging **11**.2 (2002) 206 (cit. on p. 166).

[14] D. P. Ballou, S. E. Madnick and R. Y. Wang, *Special Section: Assuring Information Quality*, J. of Management Information Systems **20**.3 (2004) 9 (cit. on p. 6).

[15] D. P. Ballou and H. L. Pazer, *Modeling Data and Process Quality in Multi-Input, Multi-Output Information Systems*, Management Science **31**.2 (1985) 150 (cit. on pp. 4, 14).

[16] C. Batini et al., "A Framework And A Methodology For Data Quality Assessment And Monitoring", *ICIQ*, ed. by M. A. Robbert et al., MIT, 12th July 2010 333 (cit. on pp. 8, 36).

[17] W. Beek et al., *Literally Better: Analyzing and Improving the Quality of Literals (Under Review)*, Semantic Web Journal (2016) (cit. on pp. 9, 155).

[18] W. Beek et al., "LOD Laundromat: A Uniform Way of Publishing Other People's Dirty Data", *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, ed. by P. Mika et al., vol. 8796, Lecture Notes in Computer Science, Springer, 2014 213, ISBN: 978-3-319-11963-2, URL: http://dx.doi.org/10.1007/978-3-319-11964-9 (cit. on pp. 3, 175).

[19] S. K. Bera et al., *Advanced Bloom Filter Based Algorithms for Efficient Approximate Data De-Duplication in Streams*, CoRR (2012) (cit. on pp. 22, 30, 93, 148).

[20] T. Berners-Lee and R. Cailliau, *World- WideWeb: Proposal for a HyperText Project*, tech. rep., CERN, 12th Nov. 1990, URL: http://www.w3.org/Proposal.html (visited on 07/08/2016) (cit. on p. 16).

[21] T. Berners-Lee, *Information Management: A Proposal*, 1989, URL: http://www.w3.org/History/1989/proposal.html (cit. on p. 16).

[22] T. Berners-Lee, *Linked Data - Design Issues*, 2006 (cit. on pp. 20, 90, 125, 126, 160).

[23] T. Berners-Lee, J. Hendler and O. Lassila, *The Semantic Web*, Scientific American **284**.5 (2001) 34, URL: http://www.sciam.com/article.cfm?id=the-semantic-web (cit. on p. 16).

[24] T. Berners-Lee et al., "Tabulator: Exploring and Analyzing linked data on the Semantic Web", English, *Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI06)*, Nov. 2006 (cit. on p. 71).

[25] C. Bizer, *Quality-Driven Information Filtering in the Context of Web-Based Information Systems*, PhD thesis: FU Berlin, 13th Mar. 2007 (cit. on p. 5).

[26] C. Bizer and R. Cyganiak, *Quality-driven Information Filtering Using the WIQA Policy Framework*, Web Semant. **7**.1 (Jan. 2009) 1, ISSN: 1570-8268 (cit. on pp. 26, 66).

[27] C. Bizer, T. Heath and T. Berners-Lee, *Linked Data - The Story So Far*, Int. J. Semantic Web Inf. Syst. **5**.3 (2009) 1 (cit. on p. 137).

[28] A. F. Blackwell and T. R. G. Green, *Notational Systems – the Cognitive Dimensions of Notations framework*, 2002, (visited on 03/08/2016) (cit. on p. 63).

[29] A. F. Blackwell et al., "Cognitive Dimensions of Notations: Design Tools for Cognitive Technology", *Cognitive Technology*, vol. 2117, Lecture Notes in Computer Science, Springer, 2001 325 (cit. on p. 63).

[30] J. Bleiholder and F. Naumann, *Data Fusion*, ACM Comput. Surv. **41**.1 (Jan. 2009) 1:1, ISSN: 0360-0300, URL: http://doi.acm.org/10.1145/1456650.1456651 (cit. on p. 148).

[31] B. H. Bloom, *Space/Time Trade-offs in Hash Coding with Allowable Errors*, Commun. ACM **13**.7 (1970) 422 (cit. on p. 22).

[32] C. Böhm, J. Lorey and F. Naumann, *Creating VoID descriptions for Web-scale data*, J. Web Sem. **9**.3 (2011) 339 (cit. on p. 140).

[33] D. Brickley, R. Guha and B. McBride, *RDF Schema 1.1*, W3C Recommendation, World Wide Web Consortium (W3C), 25th Feb. 2014, (visited on 03/08/2016) (cit. on pp. 16, 19, 150).

[34] A. Z. Broder and M. Mitzenmacher, *Survey: Network Applications of Bloom Filters: A Survey*, Internet Mathematics (2004) (cit. on p. 30).

[35] H. U. Buhl et al., *Big Data*, Business & Information Systems Engineering **5**.2 (2013) 65, ISSN: 1867-0202 (cit. on p. 15).

[36] J. J. Carroll et al., "Named Graphs, Provenance and Trust", *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, Chiba, Japan: ACM, 2005 613, ISBN: 1-59593-046-9 (cit. on pp. 18, 47, 73).

[37] M. Chortis and G. Flouris, "A Diagnosis and Repair Framework for DL-LiteA KBs", *The Semantic Web: ESWC 2015 Satellite Events*, ed. by F. Gandon et al., vol. 9341, Lecture Notes in Computer Science, Springer International Publishing, 2015 199, ISBN: 978-3-319-25638-2 (cit. on p. 37).

[38] P. B. Crosby, *Quality is Free, The Art of Making Quality Certain*, Mentor book, McGraw-Hill, 1979, ISBN: 9780070145122 (cit. on p. 4).

[39] E. Curry, A. Freitas and S. O'Riáin, "The Role of Community-Driven Data Curation for Enterprises", *Linking Enterprise Data*, ed. by D. Wood, Boston, MA: Springer US, 2010 25, ISBN: 978-1-4419-7665-9 (cit. on p. 36).

[40] R. Cyganiak, D. Reynolds and J. Tennison, *The RDF Data Cube Vocabulary*, tech. rep., World Wide Web Consortium (W3C), 16th Jan. 2014, (visited on 03/08/2016) (cit. on pp. 8, 45, 72, 195).

[41] R. Cyganiak, D. Wood and M. Lanthaler, *RDF 1.1 Concepts and Abstract Syntax*, W3C Recommendation, World Wide Web Consortium (W3C), 25th Feb. 2014, URL: http://www.w3.org/TR/rdf11-concepts/ (visited on 03/08/2016) (cit. on p. 16).

[42] A. Dasgupta, R. Kumar and T. Sarlos, "On Estimating the Average Degree", *WWW*, New York, NY, USA: ACM, 2014 795, ISBN: 978-1-4503-2744-2 (cit. on p. 30).

[43] J. Debattista, S. Auer and C. Lange, *Luzzu - A Methodology and Framework for Linked Data Quality Assessment*, ACM J. Data Inform. Quality ((To Appear)) (cit. on p. 8).

[44] J. Debattista, S. Auer and C. Lange, "Luzzu - A Framework for Linked Data Quality Assessment", *Tenth IEEE International Conference on Semantic Computing, ICSC 2016, Laguna Hills, CA, USA, February 4-6, 2016*, IEEE, 2016 124, ISBN: 978-1-5090-0662-5 (cit. on p. 8).

[45] J. Debattista, C. Lange and S. Auer, *daQ, an Ontology for Dataset Quality Information*, Linked Data on the Web (LDOW) (2014) (cit. on p. 8).

[46] J. Debattista, C. Lange and S. Auer, "Luzzu - A Framework for Linked Data Quality Assessment (Demo)", *ISWC 2015 Posters and Demonstrations Track*, ed. by S. Villata, J. Z. Pan and M. Dragoni, vol. 1486, CEUR Workshop Proceedings, CEUR-WS.org, 2015, URL: http://ceur-ws.org/Vol-1486/paper%5C_74.pdf (cit. on pp. 8, 49).

[47] J. Debattista, C. Lange and S. Auer, "Representing dataset quality metadata using multi-dimensional views", *Proceedings of the 10th International Conference on Semantic Systems - SEM '14*, New York, New York, USA: ACM Press, Sept. 2014 92, ISBN: 9781450329279 (cit. on pp. 8, 48).

[48] J. Debattista et al., "Linked 'Big' Data: Towards a Manifold Increase in Big Data Value and Veracity", *2nd IEEE/ACM International Symposium on Big Data Computing, BDC 2015, Limassol, Cyprus, December 7-10, 2015*, ed. by I. Raicu, O. F. Rana and R. Buyya, IEEE, 2015 92, ISBN: 978-0-7695-5696-3, URL: http://dx.doi.org/10.1109/BDC.2015.34 (cit. on p. 9).

[49] J. Debattista et al., "Quality Assessment of Linked Datasets Using Probabilistic Approximation", *The Semantic Web. Latest Advances and New Domains: 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 – June 4, 2015. Proceedings*, ed. by F. Gandon et al., Cham: Springer International Publishing, 2015 221, ISBN: 978-3-319-18818-8 (cit. on pp. 9, 51, 52, 87, 96, 102).

[50] J. Debattista et al., *Software prototype of the crawling, ranking and appraisal services*, Deliverable D5.2, DIACHRON: Managing the Evolution and Preservation of the Data Web, 2014 (cit. on p. 85).

[51] F. Deng and D. Rafiei, "Approximately detecting duplicates for streaming data using stable bloom filters", *SIGMOD '06: Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, Chicago, IL, USA: ACM, 2006 25, ISBN: 1-59593-434-0 (cit. on p. 30).

[52] A. V. Deursen, P. Klint and J. Visser, *Domain-Specific Languages: An Annotated Bibliography.*, Sigplan Notices **35**.6 (2000) 26 (cit. on pp. 27, 43, 57).

[53] J. Dixon, *Pentaho, Hadoop, and Data Lakes*, Pentaho, Hadoop, and Data Lakes, Oct. 2014, URL: https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/ (cit. on p. 3).

[54] B. Ell, D. Vrandecic and E. P. B. Simperl, "Labels in the Web of Data.", *International Semantic Web Conference (1)*, ed. by L. Aroyo et al., vol. 7031, Lecture Notes in Computer Science, Springer, 2011 162, ISBN: 978-3-642-25072-9 (cit. on pp. 144, 145).

[55] I. Ermilov et al., "Linked Open Data Statistics: Collection and Exploitation", *Proceedings of the 4th Conference on Knowledge Engineering and Semantic Web*, 2013 (cit. on pp. 28, 29).

[56] D. Firmani et al., *On the Meaningfulness of "Big Data Quality" (Invited Paper)*, Data Science and Engineering **1**.1 (2016) 6, ISSN: 2364-1541, URL: http://dx.doi.org/10.1007/s41019-015-0004-7 (cit. on pp. 13, 15).

[57] A. Flemming, *Quality characteristics of linked data publishing datasources*, MA thesis: Humboldt-Universität zu Berlin, Institut für Informatik, 2011 (cit. on pp. 5, 26, 162, 163).

[58] T. O. K. Foundation, *The Open Definition* (cit. on pp. 126, 129, 158, 160).

[59] Y. Gil and V. Ratnakar, "TRELLIS: An Interactive Tool for Capturing Information Analysis and Decision Making.", *EKAW*, ed. by A. Gómez-Pérez and V. R. Benjamins, vol. 2473, Lecture Notes in Computer Science, Springer, 2002 37, ISBN: 3-540-44268-5 (cit. on p. 26).

[60] T. R. Gruber, *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*, Int. J. Hum.-Comput. Stud. **43**.5-6 (Dec. 1995) 907, ISSN: 1071-5819, URL: http://dx.doi.org/10.1006/ijhc.1995.1081 (cit. on p. 19).

[61] C. Guéret et al., "Assessing Linked Data Mappings Using Network Measures.", *ESWC*, ed. by E. Simperl et al., vol. 7295, Lecture Notes in Computer Science, Springer, 2012 87, ISBN: 978-3-642-30283-1 (cit. on pp. 27, 30, 66, 90, 94).

[62] S. J. Hardiman and L. Katzir, "Estimating clustering coefficients and size of social networks via random walk.", *WWW*, ed. by D. Schwabe et al., International World Wide Web Conferences Steering Committee / ACM, 2013 539, ISBN: 978-1-4503-2035-1 (cit. on pp. 22, 30, 94).

[63] S. Harispe et al., *A Framework for Unifying Ontology-based Semantic Similarity Measures: a Study in the Biomedical Domain*, Journal of Biomedical Informatics **48** (2014) 38 (cit. on pp. 112, 113).

[64] S. Harispe et al., *Semantic Measures for the Comparison of Units of Language, Concepts or Entities from Text and Knowledge Base Analysis*, ArXiv **1310.1285** (Oct. 2013), arXiv: 1310.1285 (cit. on pp. 24, 108).

[65] S. Harispe et al., *The semantic measures library and toolkit: fast computation of semantic similarity and relatedness using biomedical ontologies*, Bioinformatics **30**.5 (2014) 740 (cit. on p. 109).

[66] S. Harris and A. Seaborne, *SPARQL 1.1 Query Language*, W3C Recommendation, World Wide Web Consortium (W3C), 21st Mar. 2013, URL: http://www.w3.org/TR/sparql11-query/ (visited on 03/08/2016) (cit. on pp. 20, 46).

[67] A. Harth, *VisiNav: A system for visual search and navigation on web data*, Web Semantics: Science, Services and Agents on the World Wide Web **8**.4 (2010) 348, Semantic Web Challenge 2009 User Interaction in Semantic Web research, ISSN: 1570-8268 (cit. on p. 71).

[68] O. Hartig, *Specification for tSPARQL*, Oct. 2008, URL: http://trdf.sourceforge.net/documents/tsparql.pdf (cit. on p. 26).

[69] O. Hartig, "Trustworthiness of Data on the Web", *STI Berlin and CSW PhD Workshop, Berlin, Germany*, 2008 (cit. on p. 26).

[70] N. Hau, R. Ichise and B. Le, "Discovering Missing Links in Large-Scale Linked Data.", *ACIIDS (2)*, ed. by A. Selamat, N. T. Nguyen and H. Haron, vol. 7803, Lecture Notes in Computer Science, Springer, 2013 468, ISBN: 978-3-642-36542-3 (cit. on p. 161).

[71] J. A. Hausman and D. A. Wise, "Stratification on Endogenous Variables and Estimation: The Gary Income Maintenance Experiment", *Structural Analysis of Discrete Data with Econometric Applications*, ed. by C. F. Manski and D. L. McFadden, Cambridge: MIT Press, 1981, chap. 10 (cit. on p. 22).

[72] P. J. Hayes and P. F. Patel-Schneider, *RDF 1.1 Semantics*, tech. rep., World Wide Web Consortium (W3C), 25th Feb. 2014, (visited on 03/08/2016) (cit. on pp. 18, 39).

[73] P. J. Hayes and P. F. Patel-Schneider, *RDF 1.1 Semantics*, W3C Recommendation, World Wide Web Consortium (W3C), 25th Feb. 2014, (visited on 03/08/2016) (cit. on p. 143).

[74] T. Heath and C. Bizer, *Linked Data: Evolving the Web into a Global Data Space*, 1st, Morgan & Claypool, 2011, ISBN: 9781608454303 (cit. on pp. 126, 132, 136, 139, 140, 144, 157, 158, 160, 162).

[75] P. Heim, J. Ziegler and S. Lohmann, "gFacet: A Browser for the Web of Data", *Proceedings of the International Workshop on Interacting with Multimedia Content in the Social Semantic Web (IMC-SSW 2008)*, vol. 417, CEUR Workshop Proceedings, Aachen, 2008 49 (cit. on p. 71).

[76] J. Helmich, J. Klímek and M. Nečaský, "Visualizing RDF Data Cubes Using the Linked Data Visualization Model", *The Semantic Web: ESWC 2014 Satellite Events: ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*, ed. by V. Presutti et al., Cham: Springer International Publishing, 2014 368, ISBN: 978-3-319-11955-7 (cit. on pp. 37, 73, 78).

[77] P. Hitzler and K. Janowicz, *Linked Data, Big Data, and the 4th Paradigm*, Semantic Web **4**.3 (2013) 233 (cit. on pp. 5, 6, 169).

[78] V. J. Hodge and J. Austin, *A Survey of Outlier Detection Methodologies.*, Artif. Intell. Rev. **22**.2 (2004) 85, URL: http://dblp.uni-trier.de/db/journals/air/air22.html#HodgeA04 (cit. on pp. 23, 24).

[79] A. Hogan, A. Harth and A. Polleres, "SAOR: Authoritative Reasoning for the Web", *ASWC*, vol. 5367, Lecture Notes in Computer Science, Springer, 2008 76 (cit. on p. 153).

[80] A. Hogan et al., *An empirical survey of Linked Data conformance*, J. Web Sem. **14** (2012) 14 (cit. on pp. 5, 32, 135–139, 145, 157, 164).

[81] A. Hogan et al.,
Searching and browsing Linked Data with SWSE: The Semantic Web Search Engine,
Web Semantics: Science, Services and Agents on the World Wide Web **9**.4 (2011) 365, JWS special issue on Semantic Search, ISSN: 1570-8268, URL: http://www.sciencedirect.com/science/article/pii/S1570826811000473 (cit. on p. 71).

[82] A. Hogan et al., "Weaving the Pedantic Web",
Linked Data on the Web Workshop (LDOW2010) at WWW'2010, 2010,
URL: http://scholar.google.com/scholar.bib?q=info:JIuMLDzUdo8J:scholar.google.com/%5C&output=citation%5C&hl=en%5C&as%5C_sdt=40000000000%5C&ct=citation%5C&cd=0 (cit. on pp. 32, 150).

[83] I. Horrocks et al., *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*,
W3C Recommendation, World Wide Web Consortium (W3C), 21st May 2004,
URL: http://www.w3.org/Submission/SWRL/ (visited on 03/08/2016) (cit. on p. 28).

[84] W. Hu and J. Feng, "Data and information quality: an information-theoretic perspective",
Proceedings of the 2nd International Conference on Information Management and Business,
Citeseer, 2006 482 (cit. on p. 15).

[85] P. Hudak, "Domain-specific languages", ed. by P. H. Salas, vol. 3,
MacMillan, Indianapolis, 1997 39 (cit. on pp. 27, 43, 57).

[86] B. Hyland, G. Atemezing and B. Villazon-Terrazas, *Best Practices for Publishing Linked Data*,
W3C Recommendation, World Wide Web Consortium (W3C), 9th Jan. 2014,
(visited on 03/08/2016) (cit. on p. 3).

[87] E. Hyvönen et al., "Linked Data Finland: A 7-star Model and Platform for Publishing and Re-using Linked Datasets", *The Semantic Web: ESWC 2014 Satellite Events - ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*,
ed. by V. Presutti et al., vol. 8798, Lecture Notes in Computer Science, Springer, 2014 226,
ISBN: 978-3-319-11954-0,
URL: http://dx.doi.org/10.1007/978-3-319-11955-7_24 (cit. on p. 70).

[88] N. Jain, M. Dahlin and R. Tewari,
"TAPER: Tiered Approach for Eliminating Redundancy in Replica Synchronization.", *FAST*,
ed. by G. Gibson, USENIX, 2005, URL:
http://dblp.uni-trier.de/db/conf/fast/fast2005.html#JainDT05
(cit. on p. 30).

[89] K. Janowicz, *Trust and Provenance You Can't Have One Without The Other*, 2009 (cit. on p. 80).

[90] J. M. Juran, *Juran's Quality Control Handbook*, 4th, Mcgraw-Hill (Tx), 1974,
ISBN: 0070331766,
URL: http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20%5C&path=ASIN/0070331766 (cit. on pp. 4, 13).

[91] T. Käfer et al., "Observing Linked Data Dynamics.", *ESWC*, ed. by P. Cimiano et al., vol. 7882,
Lecture Notes in Computer Science, Springer, 2013 213, ISBN: 978-3-642-38288-8, URL:
http://dblp.uni-trier.de/db/conf/esws/eswc2013.html#KaferAUOH13
(cit. on pp. 129, 174).

[92] H. F. Kaiser, *An index of factorial simplicity*, Psychometrika **39**.1 (1974) 31, ISSN: 1860-0980,
URL: http://dx.doi.org/10.1007/BF02291575 (cit. on p. 167).

[93] M. Kaiser, M. Klier and B. Heinrich,
"How to Measure Data Quality? - A Metric-Based Approach.", *ICIS*,
Association for Information Systems, 2007 108, URL:
http://dblp.uni-trier.de/db/conf/icis/icis2007.html#KaiserKH07
(cit. on p. 30).

[94] M. Kifer, "Rule Interchange Format: The Framework.", *RR*, ed. by D. Calvanese and G. Lausen,
vol. 5341, Lecture Notes in Computer Science, Springer, 23rd Oct. 2008 1,
ISBN: 978-3-540-88736-2 (cit. on p. 28).

[95] J. M. Kleinberg, *Hubs, authorities, and communities*, ACM Comput. Surv. **31**.4es (1999) 5
(cit. on p. 5).

[96] J. Klímek, J. Helmich and M. Nečaský, "Application of the Linked Data Visualization Model on
Real World Data from the Czech LOD Cloud.", *LDOW*, ed. by C. Bizer et al., vol. 1184,
CEUR Workshop Proceedings, CEUR-WS.org, 2014, URL:
http://dblp.uni-trier.de/db/conf/www/ldow2014.html#KlimekHN14
(cit. on p. 78).

[97] E. M. Knorr, R. T. Ng and V. Tucakov, *Distance-based Outliers: Algorithms and Applications*,
The VLDB Journal **8**.3-4 (Feb. 2000) 237, ISSN: 1066-8888,
URL: http://dx.doi.org/10.1007/s007780050006
(cit. on pp. 23, 24, 106, 107, 109–111, 121, 173).

[98] M. Knuth, J. Hercher and H. Sack, "Collaboratively Patching Linked Data", *Proc. of. 2nd Int. Workshop on Usage Analysis and the Web of Data (USEWOD 2012), co-located with the 21st International World Wide Web Conference 2012 (WWW 2012), April 17, 2012, Lyon (France)*,
2012 (cit. on p. 37).

[99] D. Kontokostas et al., "Test-driven evaluation of linked data quality", *WWW*,
ed. by C.-W. Chung et al., ACM, 2014 747, ISBN: 978-1-4503-2744-2,
URL: http://dblp.uni-
trier.de/db/conf/www/www2014.html#KontokostasWAHLCZ14
(cit. on pp. 27, 43, 51, 66).

[100] T. Lebo et al., *PROV-O: The PROV Ontology*, W3C Recommendation,
World Wide Web Consortium (W3C), 30th Apr. 2013, (visited on 02/08/2016) (cit. on pp. 8, 72).

[101] D. Lee, B. F. Lóscio and P. Archer, *Data on the Web Best Practices Use Cases & Requirements*,
W3C Working Group Note, World Wide Web Consortium, Feb. 2015,
URL: http://www.w3.org/TR/dwbp-ucr/ (visited on 08/08/2016) (cit. on p. 72).

[102] J. Leskovec and C. Faloutsos, "Sampling from Large Graphs", *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06,
Philadelphia, PA, USA: ACM, 2006 631, ISBN: 1-59593-339-5,
URL: http://doi.acm.org/10.1145/1150402.1150479 (cit. on p. 176).

[103] L. Liu and M. T. Özsu, eds., *Encyclopedia of Database Systems*, Springer US, 2009,
ISBN: 978-0-387-35544-3, 978-0-387-39940-9 (cit. on p. 25).

[104] A. Llewellyn, *NASA Tournament Lab's Big Data Challenge*, 2012,
URL: https://open.nasa.gov/blog/2012/10/03/nasa-tournament-labs-
big-data-challenge/ (visited on 15/06/2016) (cit. on p. 3).

[105] B. F. Lóscio, C. Burle and N. Calegari, *Data on the Web Best Practices, W3C Working Draft*, W3C Recommendation Candidate, World Wide Web Consortium, Feb. 2016, URL: http://www.w3.org/TR/dwbp/ (visited on 08/08/2016) (cit. on pp. 3, 69, 81, 125, 141–143, 172).

[106] B. F. Lóscio, E. G. Stephan and S. Purohit, *Data Usage Vocabulary (DUV), W3C Working Draft*, tech. rep., World Wide Web Consortium, 2016, URL: http://www.w3.org/TR/vocab-duv/ (visited on 08/08/2016) (cit. on p. 126).

[107] T. Lukoianova (Vashchilko) and V. L. Rubin, *Veracity roadmap: Is Big Data objective, truthful and credible?*, Advances in Classification Research Online **24**.1 (2014) 4 (cit. on p. 15).

[108] F. Maali, J. Erickson and P. Archer, *Data Catalog Vocabulary (DCAT)*, W3C Recommendation, World Wide Web Consortium, 2014, URL: http://www.w3.org/TR/vocab-dcat/ (visited on 08/08/2016) (cit. on pp. 29, 126, 127).

[109] C. Mader, M. Martin and C. Stadler, "Facilitating the Exploration and Visualization of Linked Data", *Linked Open Data – Creating Knowledge Out of Interlinked Data*, ed. by S. Auer, V. Bryl and S. Tramp, Lecture Notes in Computer Science, Springer International Publishing, 2014 90, ISBN: 978-3-319-09845-6 (cit. on pp. 37, 73, 78).

[110] E. Mäkelä, "Aether – Generating and Viewing Extended VoID Statistical Descriptions of RDF Datasets", *The Semantic Web: ESWC 2014 Satellite Events: ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*, ed. by V. Presutti et al., Cham: Springer International Publishing, 2014 429, ISBN: 978-3-319-11955-7, URL: http://dx.doi.org/10.1007/978-3-319-11955-7%5C_61 (cit. on p. 29).

[111] A. Mallea et al., "On Blank Nodes", *10th Int. Semantic Web Conf.* ISWC'11, Bonn, Germany: Springer, 2011 421, ISBN: 978-3-642-25072-9 (cit. on p. 139).

[112] A. J. Max Schmachtenberg Christian Bizer and R. Cyganiak, *Linking Open Data Cloud Diagram 2014*, version 0.4, 2014, URL: http://lod-cloud.net (visited on 11/10/2015) (cit. on pp. 3, 21, 125).

[113] G. K. Mazandu and N. J. Mulder, *A Topology-Based Metric for Measuring Term Similarity in the Gene Ontology*, Adv. Bioinformatics (2012) 975783:1, URL: http://dblp.uni-trier.de/db/journals/abi/abi2012.html#MazanduM12 (cit. on pp. 109, 113).

[114] P. N. Mendes, H. Mühleisen and C. Bizer, "Sieve: linked data quality assessment and fusion", *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, ed. by D. Srivastava and I. Ari, ACM, 2012 116 (cit. on pp. 27, 66, 90, 148).

[115] M. Mernik, J. Heering and A. M. Sloane, *When and How to Develop Domain-specific Languages*, ACM Comput. Surv. **37**.4 (Dec. 2005) 316, ISSN: 0360-0300 (cit. on pp. 27, 28, 43, 57, 59).

[116] A. Metwally, D. Agrawal and A. E. Abbadi, "Duplicate detection in click streams", *WWW*, ACM, 2005 12 (cit. on p. 30).

[117] P. Mika et al., eds., *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, vol. 8796, Lecture Notes in Computer Science, Springer, 2014, ISBN: 978-3-319-11963-2, URL: http://dx.doi.org/10.1007/978-3-319-11964-9.

[118] P. Missier et al., "Quality Views: Capturing and Exploiting the User Perspective on Data Quality", *VLDB*, ed. by U. Dayal et al., Seoul, Korea: ACM, Sept. 2006 977, ISBN: 1-59593-385-9, URL: http://www.vldb.org/conf/2006/p977-missier.pdf (cit. on p. 29).

[119] A. Mohaisen, A. Yun and Y. Kim, "Measuring the mixing time of social graphs", *Internet Measurement Conference*, ed. by M. Allman, ACM, 2010 383, ISBN: 978-1-4503-0483-2, URL: http://dblp.uni-trier.de/db/conf/imc/imc2010.html#MohaisenYK10 (cit. on pp. 94, 97).

[120] H. Mühleisen, *Vocabulary Usage by Pay-Level Domain, Web Data Commons Analysis Result*, tech. rep., Freie Universität Berlin, 2014, URL: http://webdatacommons.org/structureddata/vocabulary-usage-analysis/ (visited on 08/08/2016) (cit. on p. 90).

[121] A.-C. N. Ngomo and S. Auer, "LIMES: A Time-efficient Approach for Large-scale Link Discovery on the Web of Data", *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Three*, IJCAI'11, AAAI Press, 2011 2312, ISBN: 978-1-57735-515-1, URL: http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-385 (cit. on pp. 71, 161).

[122] P. B. Nguyen, M. Luong and A. Beghdadi, "Statistical Analysis of Image Quality Metrics for Watermark Transparency Assessment", *Advances in Multimedia Information Processing - PCM 2010: 11th Pacific Rim Conference on Multimedia, Shanghai, China, September 21-24, 2010, Proceedings, Part I*, ed. by G. Qiu et al., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010 685, ISBN: 978-3-642-15702-8, URL: http://dx.doi.org/10.1007/978-3-642-15702-8_63 (cit. on p. 166).

[123] M. Nottingham and E. Hammer-Lahav, *Defining Well-Known Uniform Resource Identifiers (URIs)*, RFC 5785 (Proposed Standard), Apr. 2010, URL: http://www.ietf.org/rfc/rfc5785.txt (cit. on p. 132).

[124] D. O'Brien, *The Key to Quality Big Data Analytics*, Jan. 2015, URL: http://insideanalysis.com/2015/01/the-key-to-quality-big-data-analytics/ (visited on 08/08/2016) (cit. on p. 89).

[125] T. Orgel et al., *Second Integration and Enrichment Services Prototype*, Deliverable 4.3, EEXCESS Enhancing Europe's eXchange in Cultural Educational and Scientific reSources, 2015 (cit. on p. 85).

[126] L. Page et al., *The PageRank Citation Ranking: Bringing Order to the Web.*, Technical Report 1999-66, Previous number = SIDL-WP-1999-0120: Stanford InfoLab, Nov. 1999, URL: http://ilpubs.stanford.edu:8090/422/ (cit. on p. 5).

[127] H. Paulheim and C. Bizer, *Improving the Quality of Linked Data Using Statistical Distributions*, Int. J. Semant. Web Inf. Syst. **10**.2 (Apr. 2014) 63, ISSN: 1552-6283, URL: http://dx.doi.org/10.4018/ijswis.2014040104 (cit. on pp. 4, 30, 106, 120, 121).

[128] K. Pearson, *On lines and planes of closest fit to systems of points in space*, Philosophical Magazine **2**.6 (1901) 559 (cit. on p. 166).

[129] S. Peroni, *Media type as Linked Open Data*, URL: http://www.sparontologies.net/mediatype/ (visited on 03/08/2016) (cit. on p. 127).

[130] D. Le-Phuoc et al., "Rapid Prototyping of Semantic Mash-Ups through Semantic Web Pipes", *Proceedings of the 17th WWW conference*, ed. by J. Quemada et al., ACM Press, 2009 581, ISBN: 978-1-60558-487-4 (cit. on p. 63).

[131] L. Pipino, Y. Lee and R. Wang, *Data Quality Assessment*, Communications of the ACM **45**.4 (2002) (cit. on pp. 13, 14, 134).

[132] R. M. Pirsig, *Zen and the Art of Motorcycle Maintenance: An Inquiry Into Values*, Essence Philosophy, Vintage, 1974, ISBN: 9780099786405 (cit. on p. 4).

[133] E. Prud'hommeaux and K. Coyle, *SHACL Core Abstract Syntax and Semantics*, W3C Recommendation, World Wide Web Consortium (W3C), 12th Sept. 2016, URL: https://www.w3.org/TR/shacl-abstract-syntax/ (visited on 03/08/2016) (cit. on p. 28).

[134] F. Radulovic, R. García-Castro and A. Gómez-Pérez, *SemQuaRE – An extension of the SQuaRE quality model for the evaluation of semantic technologies*, Computer Standards and Interfaces **38** (2015) 101, ISSN: 0920-5489 (cit. on p. 29).

[135] K. J. Reiche and E. Höfig, "Implementation of Metadata Quality Metrics and Application on Public Government Data", *COMPSAC Workshops*, IEEE Computer Society, 2013 236 (cit. on p. 126).

[136] V. Rodriguez-Doncel, S. Villata and A. Gomez-Perez, "A dataset of RDF licenses", *Legal Knowledge and Information Systems - JURIX 2014: The Twenty-Seventh Annual Conference, Jagiellonian University, Krakow, Poland, 10-12 December 2014*, ed. by I. O. S. Press, vol. 271, Frontiers in Artificial Intelligence and Applications, IOS Press, 2014 187, ISBN: 978-1-61499-467-1 (cit. on pp. 129, 158, 159).

[137] E. Ruckhaus, O. Baldizán and M.-E. Vidal, "Analyzing Linked Data Quality with LiQuate", *OTM Workshops*, 2013, ISBN: 978-3-642-41032-1 (cit. on pp. 26, 66).

[138] A. Rula and A. Zaveri, "Methodology for Assessment of Linked Data Quality", *Proceedings of the 1st Workshop on Linked Data Quality co-located with 10th International Conference on Semantic Systems, LDQ@SEMANTiCS 2014, Leipzig, Germany, September 2nd, 2014.* Ed. by M. Knuth, D. Kontokostas and H. Sack, vol. 1215, CEUR Workshop Proceedings, CEUR-WS.org, 2014, URL: http://dblp.uni-trier.de/db/conf/i-semantics/ldq2014.html (cit. on pp. 8, 36).

[139] B. Saberi and N. Ghadiri, *Sample-Based Approach to Data Quality Assessment in Spatial Databases with Application to Mobile Trajectory Nearest-Neighbor Search*, CoRR (2014) (cit. on p. 30).

[140] D. Sánchez, M. Batet and D. Isern, *Ontology-based Information Content Computation*, Know.-Based Syst. **24**.2 (Mar. 2011) 297, ISSN: 0950-7051, URL: http://dx.doi.org/10.1016/j.knosys.2010.10.001 (cit. on p. 113).

[141] J. Sanger et al., "Trust and Big Data: A Roadmap for Research", *Database and Expert Systems Applications (DEXA), 2014 25th International Workshop on*, Sept. 2014 278 (cit. on p. 15).

[142] L. Sauermann and R. Cyganiak, *Cool URIs for the Semantic Web*, W3C Interest Group Note, World Wide Web Consortium, 2008, URL: http://www.w3.org/TR/cooluris/ (visited on 08/08/2016) (cit. on pp. 90, 135).

[143] M. Schmachtenberg, C. Bizer and H. Paulheim, "Adoption of the Linked Data Best Practices in Different Topical Domains", *13th Int. Semantic Web Conf.* Ed. by P. Mika et al., vol. 8796, Lecture Notes in Computer Science, Springer, 2014 245, ISBN: 978-3-319-11963-2 (cit. on pp. 137, 145, 160).

[144] M. K. Smith, C. Welty and D. L. McGuinness, *OWL Web Ontology Language Guide*, W3C Recommendation, World Wide Web Consortium (W3C), 5th Sept. 2016, URL: http://www.w3.org/TR/owl-guide/ (visited on 03/08/2016) (cit. on pp. 19, 149).

[145] O. Suominen and C. Mader, *Assessing and Improving the Quality of SKOS Vocabularies*, J. Data Semantics **3**.1 (2014) 47 (cit. on pp. 4, 32).

[146] H. Thakkar et al., "Are Linked Datasets Fit for Open-domain Question Answering? A Quality Assessment", *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics*, WIMS '16, New York, NY, USA: ACM, 2016, ISBN: 978-1-4503-4056-4 (cit. on pp. 9, 85, 175).

[147] O. Théreaux, *Common HTTP Implementation Problems*, W3C Note, World Wide Web Consortium, Jan. 2003, URL: http://www.w3.org/TR/chips (cit. on p. 135).

[148] G. Töpper, M. Knuth and H. Sack, "DBpedia Ontology Enrichment for Inconsistency Detection", *Proceedings of the 8th International Conference on Semantic Systems*, I-SEMANTICS '12, New York, NY, USA: ACM, 2012 33, ISBN: 978-1-4503-1112-0, URL: http://doi.acm.org/10.1145/2362499.2362505 (cit. on pp. 30, 31, 121).

[149] S. Vale, *Classification of Types of Big Data*, ed. by U. N. E. C. for Europe, United Nations Economic Commission for Europe (UNECE), 2013, URL: http://www1.unece.org/stat/platform/display/bigdata/Classification+of+Types+of+Big+Data (visited on 06/08/2016) (cit. on p. 15).

[150] R. Verborgh et al., "Querying Datasets on the Web with High Availability", *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, ed. by P. Mika et al., vol. 8796, Lecture Notes in Computer Science, Springer, 2014 180, ISBN: 978-3-319-11963-2, URL: http://dx.doi.org/10.1007/978-3-319-11964-9 (cit. on p. 38).

[151] J. S. Vitter, *Random Sampling with a Reservoir*, ACM Trans. Math. Softw. **11**.1 (1985) 37 (cit. on pp. 21, 22, 30, 91).

[152] J. Volz et al., "Discovering and Maintaining Links on the Web of Data", *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, Berlin, Heidelberg: Springer-Verlag, 2009 650, ISBN: 978-3-642-04929-3 (cit. on pp. 71, 161).

[153] J. Waitelonis et al., *WhoKnows? - Evaluating Linked Data Heuristics with a Quiz that Cleans Up DBpedia*, International Journal of Interactive Technology and Smart Education (ITSE) **8**.3 (2011) 236, ISSN: 1741-5659 (cit. on pp. 30, 31, 121).

[154] Y. Wand and R. Y. Wang, *Anchoring Data Quality Dimensions in Ontological Foundations.*, Commun. ACM **39**.11 (1996) 86, URL: http://dblp.uni-trier.de/db/journals/cacm/cacm39.html#WandW96 (cit. on p. 13).

[155] R. Y. Wang and D. M. Strong, *Beyond accuracy: What data quality means to data consumers*, Journal of management information systems (1996) (cit. on pp. 6, 13, 14).

[156] D. Wienand and H. Paulheim, "Detecting Incorrect Numerical Data in DBpedia", *11th ESWC 2014 (ESWC2014)*, May 2014 (cit. on pp. 30, 121).

[157] H. Wu et al., "How Redundant is It? - an Empirical Analysis on Linked Datasets", *Proceedings of the 5th International Conference on Consuming Linked Data - Volume 1264*, COLD'14, Riva del Garda, Italy: CEUR-WS.org, 2014 97, URL: http://dl.acm.org/citation.cfm?id=2877789.2877798 (cit. on p. 148).

[158] H. Xie, X.-H. Tong and Z.-Q. Jiang, "The Quality Assessment and Sampling Model for the Geological Spatial Data in China", *ISPRS Archives*, vol. XXXVII Par, ISPRS, 2008 (cit. on p. 30).

[159] Y. Yang et al., "Dereferencing Semantic Web URIs: What is 200 OK on the Semantic Web?", 2011, URL: http://dl.dropbox.com/u/4138729/paper/dereference_iswc2011.pdf (visited on 08/08/2016) (cit. on p. 90).

[160] A. Zaveri et al., *Quality Assessment for Linked Data: A Survey*, Semantic Web Journal **7**, 63–93 (2015) (cit. on pp. 4, 5, 7, 9, 14, 15, 32, 35, 48, 55, 58, 65, 72, 74, 75, 78, 85, 89, 101, 121, 123, 126, 134, 135, 137, 138, 140–144, 146–155, 157, 166, 168, 169, 172, 174).

[161] A. Zaveri et al., "User-driven Quality Evaluation of DBpedia", *Proceedings of the 9th International Conference on Semantic Systems*, I-SEMANTICS '13, Graz, Austria: ACM, 2013 97, ISBN: 978-1-4503-1972-0 (cit. on pp. 26, 30, 31, 121).

[162] Z. Zhou, Y. Wang and J. Gu, "A New Model of Information Content for Semantic Similarity in WordNet", *FGCNS'08 Proceedings of the 2008 Second International Conference on Future Generation Communication and Networking Symposia - Volume 03*, IEEE Computer Society, Dec. 2008 85, ISBN: 978-1-4244-3430-5 (cit. on pp. 109, 113).

[163] H. Zhu et al., "Data and Information Quality Research: Its Evolution and Future", *Computing Handbook, 3rd ed. (2)*, CRC Press, 2014 16: 1 (cit. on p. 15).

Appendix

APPENDIX A

List of Prefixes and Namespaces Used

Prefix	Namespace
CC	http://creativecommons.org/ns#
CUBE	http://purl.org/linked-data/cube#
DAQ	http://purl.org/eis/vocab/daq#
DC	http://purl.org/dc/elements/1.1/
DCAT	http://www.w3.org/ns/dcat#
DCTerms	http://purl.org/dc/terms/
DOAP	http://usefulinc.com/ns/doap#
DUV	http://www.w3.org/ns/duv#
DQV	http://www.w3.org/ns/dqv#
FOAF	http://xmlns.com/foaf/0.1/
GEO	http://www.w3.org/2003/01/geo/wgs84_pos#
OWL	http://www.w3.org/2002/07/owl#
PROV-O	http://www.w3.org/ns/prov#
RDF	http://www.w3.org/1999/02/22-rdf-syntax-ns#
RDFS	http://www.w3.org/2000/01/rdf-schema#
SCHEMA	http://schema.org/
SIOC	http://rdfs.org/sioc/ns#
SKOS	http://www.w3.org/2004/02/skos/core#
VOID	http://rdfs.org/ns/void#

Table A.1: Most frequent prefixes and namespaces used in this thesis.

APPENDIX B

Data Cube Population Completeness Quality Metric

Population completeness is a subjective quality metric that depends very much on the use case in question. As part of our work in DIACHRON, the population completeness metric was implemented for generic RDF data cube [40] datasets, more specifically those used by the Data Publica pilot partner. The main idea is to check the completeness of observations against the original code list. For example check that there exists at least one mayor for every city in France (see Listing B.1 condition). Mayors are listed as observations (as they might change over time) and cities are found in the code list. In this metric we treat the code list as the gold standard, to which observations are compared to. We implement this metric as a complex QMP (cf. Section 4.3.5). As part of the assessment, the interested users should first create their own completeness configuration, which includes a reference to the gold standard (code list), reference to the data structure definition, and the satisfying completeness conditions. The data structure definition is where component properties in data cube are defined. Listing B.1 shows a configuration example for the metric.

```
[] a ex:PopulationCompletenessConfiguration ;
ex:goldStandard <http://example.org/dp2012.ttl> ;
ex:dsd <http://example.org/dp2012.ttl> ;
ex:conditionConfiguration [
  a ex:ConditionConfiguration ;
  ex:assessedComponentProperty <http://www.data-publica.com/lod/publication/dp#
  commune-dim> ;
  ex:conditionOperator "="^^xsd:string ;
  ex:conditionValue "1"^^xsd:integer ;
] .
```

Listing B.1: "A Population Completeness Configuration"

This configuration is used in the precursor functions which first extracts the defined conditions. Following that, we retrieve all coded properties from the data structure definition file, and for each coded property we get the respective code list from the gold standard file. We then start the actual assessment, where we check if each observation is meeting the conditions set in the configuration file, and the coverage is then calculated as the total number of observations with values from the code list, and the actual total number of entries in the code list.

APPENDIX C

Creating a Custom LQML Function

The Luzzu Quality Metric Language (cf. Chapter 5) can be easily extended by creating functions using the Java Language. Listing C.1 is a Java example of a custom function that can be used within the `match` clause.

```
public class IsDeprecatedClass implements ICustomCondition {
  public boolean compute(Object... args) throws IllegalArgumentException {
    if(args.length != 1) {
      throw new IllegalArgumentException("Illegal Number of Arguments, Required 1")
    ;
    }

    Node n = ((Node) args[0]);
    if (n.isURI()){
        Model m = RDFDataMgr.loadModel(n.getNameSpace());
        boolean isDeprecated = m.listObjectsOfProperty(m.createResource(n.getURI())
    , RDF.type).filterKeep(deprecatedfilter).hasNext();
        return isDeprecated;
    } else return false;
  }

  private Filter<RDFNode> deprecatedfilter = new Filter<RDFNode>() {
      @Override
      public boolean accept(RDFNode node) {
        return (node.equals(OWL.DeprecatedClass));
      }
  }
}
```

Listing C.1: "Defining the IsDereferencableClass as an LQML function"

APPENDIX D

LOD Cloud Quality Evaluation Results

In this appendix we display the quality metric values for the assessed datasets. The datasets are sorted by their aggregated score. For legible reasons, we remove the `http://` prefix from each namespace, and split the results into 3 different tables, the first table showing the aggregated and representational category scores, the second showing the contextual and accessibility category scores, and the last table showing the intrinsic category scores. All results and metadata is also available at https://w3id.org/lodquator/

			Representational Category						
Pos	**Namespace**	**Aggregated Score**	**RC1**	**RC2**	**IO1**	**IN3**	**IN4**	**V1**	**V2**
1	zbw.eu/stw	84.72	100.00	100.00	82.89	92.11	96.69	100.00	91.67
2	id.sgcb.mcu.es	83.91	100.00	100.00	100.00	100.00	100.00	0.00	82.58
3	kdata.kr	82.22	99.50	100.00	11.58	91.51	100.00	0.00	91.67
4	www.morelab.deusto.es	80.12	59.38	100.00	30.65	80.65	100.00	0.00	82.58
5	mapasinteractivos.didactalia.net/ comunidad/mapasflashinteractivos	74.18	81.82	100.00	94.59	94.59	100.00	93.18	82.58
6	opendata.aragon.es/	72.19	69.62	99.96	13.95	50.23	99.99	0.00	82.58
7	bfs.270a.info/	71.30	100.00	100.00	9.36	30.71	99.99	0.00	95.45
8	dblp.l3s.de/d2r/	71.12	99.30	100.00	36.59	90.24	100.00	0.00	82.58
9	bis.270a.info/	70.91	100.00	100.00	9.84	57.38	100.00	0.00	82.58
10	fao.270a.info/	70.73	100.00	100.00	24.00	70.00	100.00	0.00	82.58
11	asn.jesandco.org	70.67	100.00	100.00	38.89	100.00	100.00	0.00	82.58
12	frb.270a.info/	69.90	98.15	100.00	7.53	33.33	100.00	0.00	82.58
13	uis.270a.info/	69.75	100.00	100.00	15.00	77.50	100.00	0.00	82.58
14	www.productontology.org	69.15	100.00	100.00	72.73	86.36	100.00	0.00	82.58
15	lod.geospecies.org	68.54	97.02	99.76	53.23	64.52	99.83	0.00	82.58
16	dbpedia.org/	68.47	79.25	100.00	0.08	93.15	100.00	0.00	82.58
17	ecb.270a.info/	68.04	100.00	100.00	11.01	16.29	99.98	0.00	82.58
18	oecd.270a.info/	67.83	97.95	100.00	3.93	20.53	99.92	0.00	91.67
19	imf.270a.info/	67.79	100.00	100.00	28.99	46.97	99.98	0.00	82.58
20	www.nobelprize.org/ nobel_organizations/nobelmedia/	67.12	91.96	100.00	36.36	90.91	100.00	0.00	91.67
21	www.myexperiment.org	66.84	81.47	99.99	17.61	91.82	97.60	0.00	82.58
22	opendata.euskadi.net/w79-home/es	66.72	71.70	100.00	94.74	100.00	98.48	0.00	91.67
23	www.kupkb.org/data/kupkb/	66.44	100.00	100.00	63.33	63.33	66.67	0.00	82.58
24	id.loc.gov/authorities/	65.15	100.00	100.00	100.00	100.00	99.93	0.00	82.58
25	lod.b3kat.de	64.76	77.66	100.00	4.71	9.41	100.00	0.00	82.44
26	lexinfo.net/	64.66	99.70	92.44	44.71	90.59	72.43	0.00	82.58
27	linkedmarkmail.wikier.org/	64.66	93.48	100.00	81.40	95.35	95.83	99.24	96.18
28	psi.oasis-open.org/iso/639/	64.58	100.00	100.00	44.44	44.44	100.00	0.00	91.67
29	dbtune.org/bbc/peel/	63.91	100.00	100.00	14.71	91.18	100.00	0.00	82.58
30	vocabulary.wolterskluwer.de/ arbeitsrecht	63.90	99.06	100.00	60.94	64.06	99.44	0.00	82.58
31	transparency.270a.info/	63.52	18.01	100.00	17.65	52.94	100.00	0.00	82.58
32	sw.opencyc.org/	63.30	85.10	99.39	0.37	0.42	98.36	0.00	82.58
33	www.icane.es/semantic-web	63.27	34.51	100.00	72.73	86.36	99.93	95.45	82.58

Continued on next page . . .

Pos	Namespace	Aggregated Score	Representational Category						
			RC1	RC2	IO1	IN3	IN4	V1	V2
34	rod.eionet.europa.eu/	63.23	86.46	100.00	34.62	46.15	100.00	0.00	82.58
35	eurostat.linked-statistics.org/	63.19	94.29	100.00	35.29	70.59	100.00	0.00	82.58
36	reference.data.gov.uk/	63.07	71.65	100.00	29.71	93.48	100.00	0.00	82.58
37	dbtropes.org/	62.77	98.30	100.00	0.03	97.82	100.00	0.00	82.58
38	glottolog.org	62.69	100.00	100.00	80.00	80.00	100.00	0.00	82.58
39	geo.linkeddata.es/	62.61	97.01	100.00	38.46	38.46	100.00	0.00	82.58
40	vocab.nerc.ac.uk/	62.46	99.66	100.00	90.00	85.00	100.00	0.00	82.58
41	rdfdata.eionet.europa.eu/	61.97	99.99	100.00	19.72	20.64	100.00	0.00	96.97
42	thesaurus.iia.cnr.it/index.php/ vocabularies/earth	61.53	100.00	100.00	100.00	100.00	100.00	0.00	91.67
43	co2emission.psi.enakting.org/	61.37	91.43	100.00	97.06	97.06	94.59	99.24	100.00
44	crime.psi.enakting.org/	61.37	91.43	100.00	97.06	97.06	94.59	99.24	100.00
45	nhs.psi.enakting.org/	61.37	91.43	100.00	97.06	97.06	94.59	99.24	100.00
46	population.psi.enakting.org/	61.37	91.43	100.00	97.06	97.06	94.59	99.24	100.00
47	era.rkbexplorer.com	61.32	99.99	100.00	30.00	60.00	100.00	0.00	82.58
48	os.rkbexplorer.com	61.12	100.00	100.00	6.67	6.67	99.99	0.00	82.58
49	resource.geolba.ac.at/	60.97	99.08	100.00	44.44	45.24	98.59	0.00	82.58
50	colinda.org	60.50	100.00	100.00	43.75	68.75	100.00	0.00	82.58
51	eunis.eea.europa.eu	60.33	79.18	100.00	10.43	57.39	100.00	0.00	94.70
52	www.pokepedia.fr	60.29	59.88	100.00	0.13	97.98	100.00	0.00	82.58
53	data.nytimes.com/	60.20	88.32	100.00	36.59	80.49	100.00	0.00	94.70
54	statistics.data.gov.uk/	60.14	72.21	100.00	33.67	45.92	100.00	0.00	82.58
55	data.dcs.shef.ac.uk/	59.93	96.73	100.00	43.33	83.33	98.62	0.00	82.58
56	worldbank.270a.info/	59.58	57.68	100.00	7.41	22.22	100.00	0.00	82.58
57	webscience.rkbexplorer.com	59.57	97.56	100.00	59.46	75.68	100.00	0.00	82.58
58	lobid.org/organisation	59.54	99.97	100.00	39.55	39.55	100.00	95.42	91.60
59	dbtune.org/jamendo/	59.50	90.47	100.00	24.32	94.59	100.00	0.00	82.58
60	govwild.hpi-web.de/project/ govwild-sources	59.14	100.00	100.00	100.00	92.86	100.00	0.00	82.58
61	cordis.rkbexplorer.com	59.08	47.10	100.00	40.63	45.31	97.66	0.00	82.58
62	courseware.rkbexplorer.com	59.05	97.79	100.00	18.52	37.04	99.07	0.00	82.58
63	jisc.rkbexplorer.com	58.98	100.00	100.00	21.95	29.27	100.00	0.00	82.58
64	wiki.rkbexplorer.com	58.84	97.89	100.00	16.00	16.00	100.00	0.00	82.58
65	data.bibsys.no/data	58.81	100.00	100.00	69.23	92.31	100.00	0.00	82.58
66	aims.fao.org/standards/ agrovoc/about	58.75	99.97	100.00	48.54	96.49	100.00	95.45	91.67
67	data.reegle.info/	58.31	98.99	100.00	22.86	87.14	100.00	0.00	82.58

Continued on next page . . .

			Representational Category						
Pos	**Namespace**	**Aggregated Score**	**RC1**	**RC2**	**IO1**	**IN3**	**IN4**	**V1**	**V2**
68	la.indymedia.org/syn/	58.16	54.55	89.76	44.83	89.66	92.00	0.00	82.58
69	darmstadt.rkbexplorer.com	58.16	100.00	100.00	16.67	25.00	100.00	0.00	82.58
70	www.josemalvarez.es/web/ 2011/11/01/nomenclator-asturias-2010/	57.72	100.00	100.00	28.57	33.33	100.00	0.00	82.58
71	dblp.rkbexplorer.com	57.64	33.26	100.00	50.00	55.56	100.00	0.00	82.58
72	webenemasuno.linkeddata.es/	57.52	34.16	100.00	51.40	74.77	99.99	0.00	82.58
73	energy.psi.enakting.org	57.41	96.83	100.00	19.09	22.27	100.00	-	-
74	education.data.gov.uk/	57.28	93.20	100.00	11.92	16.30	53.62	0.00	82.58
75	unlocode.rkbexplorer.com	57.00	100.00	100.00	26.67	40.00	100.00	0.00	82.58
76	msc2010.org/mscwork/	56.79	99.99	100.00	80.49	85.37	99.91	0.00	94.70
77	lingweb.eva.mpg.de/ids/	56.61	100.00	100.00	29.63	33.33	100.00	0.00	82.58
78	laas.rkbexplorer.com	56.57	100.00	100.00	7.55	11.32	97.97	0.00	82.58
79	irit.rkbexplorer.com	56.41	100.00	100.00	8.51	12.77	96.73	0.00	82.58
80	ft.rkbexplorer.com	56.37	60.00	100.00	10.81	16.22	100.00	0.00	82.58
81	extbi.lab.aau.dk/	56.35	81.82	100.00	92.31	92.31	100.00	0.00	82.58
82	vocabulary.semantic-web.at/ AustrianSkiTeam	56.23	90.65	99.93	39.51	47.32	96.31	0.00	82.58
83	www.bibsonomy.org/	56.07	97.67	70.25	25.49	82.35	52.44	0.00	82.58
84	opendatacommunities.org/ datasets/geography	55.75	100.00	100.00	55.56	55.56	100.00	0.00	82.58
85	purl.org/NET/rdflicense	55.67	97.72	99.09	19.90	36.41	85.26	0.00	82.58
86	rhizomik.net/semanticxbrl/	55.67	100.00	100.00	100.00	100.00	100.00	0.00	82.58
87	www.lexvo.org	55.55	92.71	100.00	20.83	54.17	100.00	0.00	94.70
88	deepblue.rkbexplorer.com	55.37	77.27	100.00	10.00	15.00	92.86	0.00	82.58
89	eurecom.rkbexplorer.com	55.36	100.00	100.00	13.51	18.92	80.44	0.00	82.58
90	pisa.rkbexplorer.com	55.28	94.97	100.00	14.71	20.59	94.97	0.00	82.58
91	www.w3.org/TR/wordnet-rdf	55.26	91.49	100.00	41.27	100.00	99.99	0.00	82.58
92	vocabulary.wolterskluwer.de/court	55.23	100.00	100.00	23.53	23.53	99.38	0.00	82.58
93	curriculum.rkbexplorer.com	55.23	100.00	100.00	3.85	34.62	94.93	0.00	82.58
94	rae2001.rkbexplorer.com	54.97	5.14	100.00	38.46	61.54	100.00	0.00	82.58
95	ulm.rkbexplorer.com	54.83	89.66	100.00	11.36	15.91	89.86	0.00	82.58
96	umbel.org	54.81	99.99	99.65	2.80	2.71	97.75	0.00	82.58
97	ibm.rkbexplorer.com	54.79	100.00	100.00	11.76	17.65	96.99	0.00	82.58
98	newcastle.rkbexplorer.com	54.56	100.00	100.00	7.02	10.53	100.00	0.00	82.58
99	vocabulary.semantic-web.at/ PoolParty/wiki/OpenData	53.94	93.35	100.00	31.06	36.65	99.33	0.00	91.67
100	budapest.rkbexplorer.com/	53.57	71.95	100.00	9.09	13.64	95.58	0.00	82.58

Continued on next page . . .

Pos	Namespace	Aggregated Score	Representational Category						
			RC1	RC2	IO1	IN3	IN4	V1	V2
101	epsrc.rkbexplorer.com	53.45	100.00	100.00	13.21	18.87	74.20	0.00	82.58
102	lod.taxonconcept.org/	53.11	67.46	100.00	3.42	9.99	99.44	0.00	82.58
103	id.ndl.go.jp/auth/ndla	52.87	100.00	100.00	74.07	88.89	54.01	0.00	91.67
104	deploy.rkbexplorer.com	52.84	32.59	100.00	11.11	16.67	95.60	0.00	82.58
105	lisbon.rkbexplorer.com	52.78	100.00	100.00	14.29	21.43	100.00	0.00	82.58
106	aemet.linkeddata.es/	52.60	0.24	100.00	7.14	85.71	100.00	0.00	91.67
107	roma.rkbexplorer.com	52.34	100.00	100.00	12.20	17.07	95.35	0.00	82.58
108	datos.fundacionctic.org/en	52.21	-	-	-	-	-	-	-
109	europeana.eu	52.08	26.52	100.00	11.54	100.00	100.00	0.00	82.44
110	italy.rkbexplorer.com	52.01	100.00	100.00	16.67	25.00	100.00	0.00	82.58
111	wals.info/	51.77	90.83	100.00	5.14	5.14	100.00	0.00	82.58
112	southampton.rkbexplorer.com	51.57	100.00	100.00	14.71	20.59	100.00	0.00	82.58
113	greek-lod.math.auth.gr/fire-brigade/	51.45	99.99	100.00	26.58	34.18	100.00	0.00	82.58
114	vocabulary.semantic-web.at/ PoolParty/wiki/semweb	51.43	93.96	100.00	10.00	15.30	98.97	0.00	82.58
115	linkedct.org/	51.41	55.38	100.00	4.76	4.76	100.00	0.00	82.58
116	kaunas.rkbexplorer.com	51.33	100.00	100.00	9.09	13.64	88.24	0.00	82.58
117	ieee.rkbexplorer.com	51.25	34.15	100.00	17.39	26.09	100.00	0.00	82.58
118	vivo.iu.edu	50.83	99.97	100.00	8.35	57.60	99.44	0.00	82.58
119	acm.rkbexplorer.com/	50.78	100.00	100.00	22.73	31.82	100.00	0.00	82.58
120	risks.rkbexplorer.com	50.48	100.00	100.00	15.38	23.08	100.00	0.00	82.58
121	ieeevis.tw.rpi.edu	49.83	3.95	100.00	22.22	53.33	100.00	0.00	82.58
122	pdev.org.uk/pdevlemon/	49.56	85.63	99.99	24.14	67.82	99.98	0.00	82.58
123	greek-lod.math.auth.gr/police/	48.84	99.74	100.00	37.78	42.22	100.00	0.00	82.58
124	minsky.gsi.dit.upm.es/ semanticwiki/index.php/Main_Page	48.39	11.98	100.00	2.88	4.33	100.00	0.00	82.58
125	nsf.rkbexplorer.com	48.38	100.00	100.00	8.33	12.50	92.31	0.00	82.58
126	citeseer.rkbexplorer.com/	48.31	41.06	100.00	21.74	34.78	100.00	0.00	82.58
127	prefix.cc/	46.64	-	100.00	66.67	66.67	0.06	0.00	82.58
128	kent.zpr.fer.hr:8080/ educationalProgram/	46.61	42.42	100.00	14.86	27.03	100.00	0.00	82.58
129	transport.data.gov.uk/	45.09	25.18	100.00	28.85	30.77	100.00	0.00	82.58
130	www.lingvoj.org/	41.41	-	100.00	10.00	10.00	0.00	0.00	82.44

Results of Quality Assessment for Every Assessed Dataset - Showing Aggregated Results of all metrics and Representational Category Results

	Contextual Category					Accessibility Category					
Namespace	**P1**	**P2**	**U1**	**U3**	**U5**	**A3**	**L1**	**L2**	**I1**	**PE2**	**PE3**
http://zbw.eu/stw	0.00	0.00	100.00	100.00	41.67	99.96	100.00	0.00	87.12	100.00	100.00
http://id.sgcb.mcu.es	-	-	-	-	-	-	-	-	-	-	-
http://kdata.kr	-	-	-	-	-	-	-	-	-	-	-
http://www.morelab.deusto.es	-	-	-	-	-	-	-	-	-	-	-
http://mapasinteractivos.didactalia.net/comunidad/mapasflashinteractivos	14.29	0.00	53.85	100.00	28.57	12.12	0.00	0.00	87.12	100.00	100.00
http://opendata.aragon.es/	-	-	-	-	-	-	-	-	-	-	-
http://bfs.270a.info/	100.00	74.81	99.92	0.00	0.00	33.81	100.00	0.00	87.12	100.00	18.96
http://dblp.l3s.de/d2r/	0.00	0.00	100.00	0.00	0.00	81.17	0.00	100.00	-	100.00	100.00
http://bis.270a.info/	100.00	33.33	0.00	0.00	0.00	93.21	100.00	0.00	73.48	100.00	100.00
http://fao.270a.info/	100.00	19.00	0.00	0.00	0.00	66.83	100.00	0.00	73.48	100.00	100.00
http://asn.jesandco.org	0.00	0.00	100.00	0.00	0.00	10.61	100.00	100.00	54.55	100.00	100.00
http://frb.270a.info/	100.00	62.03	0.00	0.00	0.00	64.29	100.00	0.00	73.48	100.00	100.00
http://uis.270a.info/	100.00	33.33	0.00	0.00	0.00	35.26	100.00	0.00	73.48	100.00	100.00
http://www.productontology.org	0.00	0.00	24.93	0.00	0.00	6.99	100.00	100.00	73.48	100.00	100.00
http://lod.geospecies.org	99.98	0.00	47.82	100.00	64.00	0.26	100.00	0.00	97.73	0.00	0.00
http://dbpedia.org/	0.00	0.00	82.78	0.00	0.00	48.31	0.00	100.00	99.24	100.00	100.00
http://ecb.270a.info/	100.00	20.68	0.00	0.00	0.00	68.37	0.00	100.00	73.48	100.00	100.00
http://oecd.270a.info/	100.00	33.33	0.00	0.00	0.00	51.90	100.00	0.00	73.48	100.00	100.00
http://imf.270a.info/	100.00	3.60	0.00	0.00	0.00	30.01	100.00	0.00	73.48	100.00	100.00
http://www.nobelprize.org/nobel_organizations/nobelmedia/	0.00	0.00	98.92	0.00	0.00	48.97	0.00	0.00	90.91	100.00	100.00
http://www.myexperiment.org	0.00	0.00	57.69	0.00	0.00	25.57	100.00	100.00	0.00	100.00	100.00
http://opendata.euskadi.net/w79-home/es	100.00	0.00	68.42	0.00	0.00	3.77	0.00	0.00	80.30	100.00	100.00
http://www.kupkb.org/data/kupkb/	100.00	0.00	100.00	0.00	0.00	31.25	0.00	0.00	73.48	100.00	100.00
http://id.loc.gov/authorities/	0.00	0.00	100.00	0.00	0.00	0.55	0.00	100.00	84.09	14.81	0.00
http://lod.b3kat.de	100.00	0.00	91.94	0.00	0.00	45.13	100.00	100.00	83.97	0.00	0.00
http://lexinfo.net/	100.00	0.00	60.34	0.00	0.00	63.05	0.00	0.00	80.30	100.00	100.00
http://linkedmarkmail.wikier.org/	0.00	0.00	28.57	100.00	0.00	8.70	0.00	0.00	80.15	0.00	0.00
http://psi.oasis-open.org/iso/639/	0.00	0.00	100.00	0.00	0.00	90.15	0.00	0.00	0.00	100.00	100.00
http://dbtune.org/bbc/peel/	0.00	0.00	51.98	0.00	0.00	100.00	0.00	0.00	54.55	100.00	100.00
http://vocabulary.wolterskluwer.de/arbeitsrecht	0.00	0.00	99.43	0.00	6.25	2.44	-	-	-	-	-
http://transparency.270a.info/	0.00	0.00	96.74	0.00	0.00	99.50	0.00	0.00	73.48	100.00	100.00
http://sw.opencyc.org/	0.00	0.00	100.00	0.00	0.00	3.16	0.00	100.00	73.48	100.00	100.00
http://www.icane.es/semantic-web	0.00	0.00	99.51	100.00	40.00	-	0.00	0.00	95.45	0.00	0.00
http://rod.eionet.europa.eu/	100.00	0.00	100.00	0.00	0.00	0.22	0.00	0.00	0.00	100.00	100.00

Continued on next page . . .

	Contextual Category					Accessibility Category					
Namespace	**P1**	**P2**	**U1**	**U3**	**U5**	**A3**	**L1**	**L2**	**I1**	**PE2**	**PE3**
http://eurostat.linked-statistics.org/	0.00	0.00	7.14	0.00	0.00	85.71	0.00	0.00	73.48	100.00	100.00
http://reference.data.gov.uk/	0.00	0.00	51.77	100.00	40.91	31.41	0.00	0.00	92.42	39.70	100.00
http://dbtropes.org/	0.00	0.00	100.00	0.00	0.00	0.02	0.00	100.00	54.55	100.00	100.00
http://glottolog.org	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	100.00
http://geo.linkeddata.es/	0.00	0.00	70.06	0.00	0.00	98.20	0.00	0.00	0.00	100.00	100.00
http://vocab.nerc.ac.uk/	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	62.80
http://rdfdata.eionet.europa.eu/	3.77	0.00	17.06	0.00	0.00	33.94	100.00	0.00	90.91	100.00	100.00
http://thesaurus.iia.cnr.it/index.php/vocabularies/earth	0.00	0.00	99.98	0.00	0.00	1.78	0.00	0.00	73.48	0.00	0.00
http://co2emission.psi.enakting.org/	0.00	0.00	33.33	0.00	0.00	8.57	0.00	0.00	73.48	0.00	0.00
http://crime.psi.enakting.org/	0.00	0.00	33.33	0.00	0.00	8.57	0.00	0.00	73.48	0.00	0.00
http://nhs.psi.enakting.org/	0.00	0.00	33.33	0.00	0.00	8.57	0.00	0.00	73.48	0.00	0.00
http://population.psi.enakting.org/	0.00	0.00	33.33	0.00	0.00	8.57	0.00	0.00	73.48	0.00	0.00
http://era.rkbexplorer.com	0.00	0.00	100.00	0.00	0.00	33.17	0.00	0.00	0.00	100.00	100.00
http://os.rkbexplorer.com	0.00	0.00	100.00	0.00	0.00	100.00	0.00	0.00	0.00	100.00	100.00
http://resource.geolba.ac.at/	0.00	0.00	21.11	0.00	5.00	4.51	0.00	0.00	90.91	100.00	100.00
http://colinda.org	0.00	0.00	94.42	0.00	0.00	7.87	0.00	0.00	80.30	0.00	100.00
http://eunis.eea.europa.eu	0.00	0.00	100.00	0.00	0.00	0.48	0.00	0.00	54.55	61.26	100.00
http://www.pokepedia.fr	0.00	0.00	43.96	0.00	0.00	32.64	0.00	0.00	54.55	100.00	100.00
http://data.nytimes.com/	0.00	0.00	95.73	0.00	0.00	2.82	100.00	0.00	73.48	0.00	0.00
http://statistics.data.gov.uk/	0.00	0.00	98.33	100.00	60.00	0.59	0.00	0.00	80.30	0.00	0.00
http://data.dcs.shef.ac.uk/	0.00	0.00	0.00	0.00	0.00	41.40	0.00	0.00	0.00	100.00	100.00
http://worldbank.270a.info/	0.00	0.00	0.00	0.00	0.00	98.52	0.00	0.00	73.48	100.00	100.00
http://webscience.rkbexplorer.com	0.00	0.00	40.21	0.00	0.00	13.15	0.00	0.00	84.09	100.00	100.00
http://lobid.org/organisation	0.00	0.00	99.99	0.00	42.86	0.93	0.00	0.00	100.00	0.00	0.00
http://dbtune.org/jamendo/	0.00	0.00	17.71	0.00	0.00	34.69	0.00	0.00	54.55	100.00	100.00
http://govwild.hpi-web.de/project/govwild-sources	0.00	0.00	55.21	0.00	0.00	3.19	0.00	0.00	0.00	0.00	100.00
http://cordis.rkbexplorer.com	20.00	0.00	0.01	100.00	0.00	29.61	0.00	0.00	73.48	33.36	100.00
http://courseware.rkbexplorer.com	0.00	0.00	7.76	0.00	0.00	99.35	0.00	0.00	0.00	100.00	100.00
http://jisc.rkbexplorer.com	0.00	0.00	49.90	0.00	0.00	99.98	0.00	0.00	0.00	100.00	100.00
http://wiki.rkbexplorer.com	0.00	0.00	0.00	0.00	0.00	98.13	0.00	0.00	54.55	73.67	100.00
http://data.bibsys.no/data	0.00	0.00	100.00	0.00	0.00	1.13	0.00	0.00	73.48	0.00	0.00
http://aims.fao.org/standards/agrovoc/about	0.00	0.00	99.99	0.00	0.00	0.78	0.00	0.00	-	0.00	0.00
http://data.reegle.info/	0.00	0.00	11.26	0.00	0.00	0.03	0.00	0.00	54.55	100.00	100.00
http://la.indymedia.org/syn/	0.00	0.00	8.33	0.00	0.00	0.00	0.00	0.00	54.55	100.00	100.00
http://darmstadt.rkbexplorer.com	0.00	0.00	0.80	0.00	0.00	95.28	0.00	0.00	0.00	100.00	100.00

Continued on next page . . .

	Contextual Category					Accessibility Category					
Namespace	**P1**	**P2**	**U1**	**U3**	**U5**	**A3**	**L1**	**L2**	**I1**	**PE2**	**PE3**
http://www.josemalvarez.es/web/2011/11/01/nomenclator-asturias-2010/	100.00	0.00	94.75	0.00	0.00	0.08	0.00	0.00	73.48	0.00	0.00
http://dblp.rkbexplorer.com	6.25	0.00	0.00	100.00	0.00	7.95	0.00	0.00	80.30	63.88	99.23
http://webenemasuno.linkeddata.es/	0.00	0.00	1.66	0.00	0.00	27.97	0.00	0.00	92.42	100.00	100.00
http://energy.psi.enakting.org	-	-	-	-	-	14.87	0.00	0.00	-	0.00	0.00
http://education.data.gov.uk/	0.00	0.00	64.28	100.00	6.90	83.72	0.00	0.00	87.12	0.00	0.00
http://unlocode.rkbexplorer.com	0.00	0.00	0.01	0.00	0.00	40.60	0.00	0.00	0.00	100.00	100.00
http://msc2010.org/mscwork/	0.00	0.00	94.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
http://lingweb.eva.mpg.de/ids/	0.00	0.00	99.99	0.00	0.00	65.72	0.00	0.00	80.30	0.00	0.00
http://laas.rkbexplorer.com	0.00	0.00	1.63	0.00	0.00	99.62	0.00	0.00	0.00	78.11	100.00
http://irit.rkbexplorer.com	0.00	0.00	0.24	0.00	0.00	79.29	0.00	0.00	0.00	93.51	100.00
http://ft.rkbexplorer.com	0.00	0.00	4.29	0.00	0.00	98.57	0.00	0.00	0.00	98.88	100.00
http://extbi.lab.aau.dk/	0.00	0.00	0.00	0.00	0.00	27.27	0.00	0.00	54.55	0.00	0.00
http://vocabulary.semantic-web.at/AustrianSkiTeam	0.00	0.00	18.95	0.00	3.85	20.53	100.00	0.00	93.94	10.35	50.78
http://www.bibsonomy.org/	0.00	0.00	16.67	0.00	0.00	0.00	0.00	0.00	0.00	100.00	100.00
http://opendatacommunities.org/datasets/geography	0.00	0.00	100.00	0.00	0.00	47.13	0.00	0.00	0.00	0.00	0.00
http://purl.org/NET/rdflicense	0.00	0.00	62.92	0.00	0.00	-	0.00	100.00	97.73	0.00	0.00
http://rhizomik.net/semanticxbrl/	0.00	0.00	-	0.00	0.00	7.40	0.00	0.00	54.55	2.82	100.00
http://www.lexvo.org	0.00	0.00	49.59	0.00	0.00	45.38	0.00	0.00	84.09	0.00	0.00
http://deepblue.rkbexplorer.com	0.00	0.00	16.33	0.00	0.00	95.45	0.00	0.00	0.00	61.37	100.00
http://eurecom.rkbexplorer.com	0.00	0.00	0.62	0.00	0.00	76.15	0.00	0.00	0.00	76.50	100.00
http://pisa.rkbexplorer.com	0.00	0.00	5.32	0.00	0.00	66.48	0.00	0.00	0.00	67.61	100.00
http://www.w3.org/TR/wordnet-rdf	0.00	0.00	68.54	0.00	0.00	54.45	0.00	0.00	0.00	0.00	0.00
http://vocabulary.wolterskluwer.de/court	0.00	0.00	48.58	0.00	4.55	0.60	-	-	-	-	-
http://curriculum.rkbexplorer.com	0.00	0.00	0.00	0.00	0.00	52.56	0.00	0.00	0.00	72.70	100.00
http://rae2001.rkbexplorer.com	0.00	0.00	0.10	0.00	0.00	54.33	0.00	0.00	0.00	100.00	100.00
http://ulm.rkbexplorer.com	0.00	0.00	6.78	0.00	0.00	96.55	0.00	0.00	0.00	42.28	100.00
http://umbel.org	0.00	0.00	96.67	0.00	0.00	47.85	0.00	0.00	54.55	0.00	0.00
http://ibm.rkbexplorer.com	0.00	0.00	2.65	0.00	0.00	99.68	0.00	0.00	0.00	29.67	90.52
http://newcastle.rkbexplorer.com	0.00	0.00	0.16	0.00	0.00	99.96	0.00	0.00	0.00	35.61	87.44
http://vocabulary.semantic-web.at/PoolParty/wiki/OpenData	0.00	0.00	50.18	0.00	3.70	10.64	0.00	0.00	80.30	0.00	0.00
http://budapest.rkbexplorer.com/	0.00	0.00	4.14	0.00	0.00	99.43	0.00	0.00	0.00	24.23	98.69
http://epsrc.rkbexplorer.com	0.00	0.00	52.48	0.00	0.00	38.53	0.00	0.00	0.00	28.55	81.73
http://lod.taxonconcept.org/	100.00	0.00	40.89	0.00	0.00	0.70	100.00	0.00	95.45	0.00	100.00
http://id.ndl.go.jp/auth/ndla	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	73.48	0.00	0.00

Continued on next page . . .

	Contextual Category					Accessibility Category					
Namespace	**P1**	**P2**	**U1**	**U3**	**U5**	**A3**	**L1**	**L2**	**I1**	**PE2**	**PE3**
http://deploy.rkbexplorer.com	0.00	0.00	1.03	0.00	0.00	71.72	0.00	0.00	0.00	100.00	100.00
http://lisbon.rkbexplorer.com	0.00	0.00	20.00	0.00	0.00	88.24	0.00	0.00	0.00	20.66	26.25
http://aemet.linkeddata.es/	0.00	0.00	99.89	0.00	0.00	0.04	0.00	0.00	54.55	0.00	0.00
http://roma.rkbexplorer.com	0.00	0.00	3.31	0.00	0.00	86.97	0.00	0.00	0.00	27.10	40.67
http://datos.fundacionctic.org/en	4.55	0.00	30.47	100.00	0.00	-	0.00	0.00	96.21	0.00	0.00
http://europeana.eu	0.00	0.00	83.34	0.00	0.00	1.25	0.00	0.00	73.28	0.00	0.00
http://italy.rkbexplorer.com	0.00	0.00	28.57	0.00	0.00	68.97	0.00	0.00	0.00	11.39	19.22
http://wals.info/	0.00	0.00	5.76	0.00	0.00	65.19	0.00	0.00	80.30	0.00	0.00
http://southampton.rkbexplorer.com	0.00	0.00	0.54	0.00	0.00	30.96	0.00	0.00	0.00	20.12	76.83
http://greek-lod.math.auth.gr/fire-brigade/	0.00	0.00	100.00	0.00	0.00	1.14	0.00	0.00	90.91	26.30	0.00
http://vocabulary.semantic-web.at/ PoolParty/wiki/semweb	0.00	0.00	21.70	0.00	1.52	8.16	0.00	0.00	98.48	0.00	0.00
http://linkedct.org/	0.00	0.00	100.00	0.00	0.00	0.46	0.00	0.00	84.09	0.00	0.00
http://kaunas.rkbexplorer.com	0.00	0.00	12.50	0.00	0.00	95.00	0.00	0.00	0.00	11.40	33.19
http://ieee.rkbexplorer.com	0.00	0.00	0.06	0.00	0.00	32.24	0.00	0.00	0.00	53.38	100.00
http://vivo.iu.edu	0.00	0.00	45.86	0.00	0.00	2.77	0.00	0.00	90.91	0.00	0.00
http://acm.rkbexplorer.com/	0.00	0.00	0.00	0.00	0.00	3.44	0.00	0.00	0.00	24.61	80.42
http://risks.rkbexplorer.com	0.00	0.00	0.09	0.00	0.00	42.86	0.00	0.00	0.00	19.44	34.30
http://ieeevis.tw.rpi.edu	0.00	0.00	53.04	0.00	0.00	32.89	0.00	0.00	54.55	0.00	0.00
http://pdev.org.uk/pdevlemon/	0.00	0.00	17.97	0.00	0.00	0.03	0.00	0.00	54.55	0.00	0.00
http://greek-lod.math.auth.gr/police/	0.00	0.00	0.98	0.00	0.00	0.84	0.00	0.00	93.94	0.00	0.00
http://minsky.gsi.dit.upm.es/ semanticwiki/index.php/Main_Page	0.00	0.00	96.47	0.00	0.00	0.18	0.00	0.00	73.48	0.00	0.00
http://nsf.rkbexplorer.com	0.00	0.00	0.01	0.00	0.00	3.32	0.00	0.00	0.00	21.84	44.37
http://citeseer.rkbexplorer.com/	0.00	0.00	0.00	0.00	0.00	1.16	0.00	0.00	0.00	19.34	73.44
http://prefix.cc/	0.00	0.00	0.00	0.00	0.00	-	0.00	0.00	0.00	0.00	0.00
http://kent.zpr.fer.hr:8080/ educationalProgram/	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	54.55	0.00	0.00
http://transport.data.gov.uk/	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
http://www.lingvoj.org/	0.00	0.00	-	0.00	0.00	-	0.00	0.00	0.00	0.00	0.00

Results of Quality Assessment for Every Assessed Dataset - Showing Contextual and Accessibility Categories Results

	Intrinsic Category								
Namespace	**CN2**	**CS1**	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**	**CS9**	**SV3**
zbw.eu/stw	99.96	100.00	100.00	100.00	100.00	100.00	100.00	95.30	100.00
id.sgcb.mcu.es	99.83	100.00	100.00	100.00	100.00	100.00	43.75	16.42	100.00

Continued on next page . . .

	Intrinsic Category								
Namespace	**CN2**	**CS1**	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**	**CS9**	**SV3**
kdata.kr	99.00	100.00	100.00	100.00	100.00	96.83	100.00	27.09	98.42
www.morelab.deusto.es	100.00	100.00	100.00	98.40	100.00	98.65	100.00	31.59	100.00
mapasinteractivos.didactalia.net/ comunidad/mapasflashinteractivos	100.00	100.00	100.00	100.00	100.00	85.71	100.00	79.28	95.24
opendata.aragon.es/	90.74	100.00	100.00	100.00	100.00	100.00	5.17	42.75	100.00
bfs.270a.info/	96.31	100.00	100.00	100.00	100.00	100.00	100.00	78.68	100.00
dblp.l3s.de/d2r/	97.59	100.00	100.00	95.70	100.00	99.87	100.00	66.04	100.00
bis.270a.info/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	64.66	100.00
fao.270a.info/	99.97	100.00	100.00	100.00	100.00	100.00	100.00	73.83	100.00
asn.jesandco.org	99.97	100.00	100.00	100.00	100.00	100.00	100.00	49.98	71.43
frb.270a.info/	99.61	100.00	100.00	100.00	100.00	100.00	100.00	66.22	100.00
uis.270a.info/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	66.19	100.00
www.productontology.org	49.90	100.00	100.00	100.00	100.00	100.00	97.00	73.07	100.00
lod.geospecies.org	99.98	100.00	100.00	99.98	100.00	99.94	97.12	46.74	100.00
dbpedia.org/	97.80	99.99	100.00	100.00	100.00	93.06	100.00	72.36	100.00
ecb.270a.info/	99.86	100.00	100.00	100.00	100.00	100.00	100.00	64.72	100.00
oecd.270a.info/	97.62	100.00	100.00	100.00	100.00	100.00	100.00	61.04	100.00
imf.270a.info/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	64.68	100.00
www.nobelprize.org/ nobel_organizations/nobelmedia/	99.60	100.00	100.00	100.00	100.00	100.00	100.00	62.93	100.00
www.myexperiment.org	90.14	100.00	100.00	99.92	100.00	94.43	100.00	65.93	100.00
opendata.euskadi.net/w79-home/es	75.00	100.00	100.00	100.00	100.00	58.33	100.00	58.93	100.00
www.kupkb.org/data/kupkb/	45.45	100.00	100.00	100.00	100.00	100.00	100.00	67.65	100.00
id.loc.gov/authorities/	99.96	100.00	100.00	100.00	100.00	100.00	100.00	77.17	100.00
lod.b3kat.de	95.33	100.00	100.00	100.00	100.00	97.85	100.00	60.12	100.00
lexinfo.net/	84.06	100.00	100.00	100.00	100.00	100.00	2.44	73.29	100.00
linkedmarkmail.wikier.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	66.87	100.00
psi.oasis-open.org/iso/639/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	73.05	100.00
dbtune.org/bbc/peel/	92.88	100.00	100.00	89.21	100.00	100.00	100.00	51.06	97.36
vocabulary.wolterskluwer.de/ arbeitsrecht	99.89	-	100.00	100.00	-	100.00	100.00	-	100.00
transparency.270a.info/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	74.15	100.00
sw.opencyc.org/	92.91	100.00	100.00	100.00	100.00	100.00	99.98	73.29	100.00
www.icane.es/semantic-web	69.74	100.00	100.00	100.00	100.00	100.00	0.00	68.79	100.00
rod.eionet.europa.eu/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	57.14	100.00
eurostat.linked-statistics.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	57.07	100.00
reference.data.gov.uk/	74.49	100.00	100.00	100.00	100.00	32.08	100.00	62.68	100.00
dbtropes.org/	98.03	100.00	100.00	100.00	100.00	100.00	0.00	63.42	100.00

Continued on next page . . .

	Intrinsic Category								
Namespace	**CN2**	**CS1**	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**	**CS9**	**SV3**
glottolog.org	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
geo.linkeddata.es/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	65.66	100.00
vocab.nerc.ac.uk/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	66.47	100.00
rdfdata.eionet.europa.eu/	37.69	100.00	100.00	100.00	100.00	100.00	100.00	52.61	99.89
thesaurus.iia.cnr.it/index.php/ vocabularies/earth	99.99	100.00	100.00	100.00	100.00	100.00	100.00	94.42	100.00
co2emission.psi.enakting.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	62.14	100.00
crime.psi.enakting.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	62.14	100.00
nhs.psi.enakting.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	62.14	100.00
population.psi.enakting.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	62.14	100.00
era.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
os.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	54.28	100.00
resource.geolba.ac.at/	99.82	100.00	100.00	100.00	100.00	100.00	100.00	55.01	100.00
colinda.org	99.47	100.00	100.00	88.42	100.00	100.00	100.00	67.82	100.00
eunis.eea.europa.eu	100.00	100.00	100.00	100.00	100.00	99.90	100.00	70.98	100.00
www.pokepedia.fr	97.56	100.00	100.00	100.00	100.00	100.00	97.55	61.14	100.00
data.nytimes.com/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	53.14	100.00
statistics.data.gov.uk/	93.47	100.00	100.00	100.00	99.97	100.00	100.00	56.80	100.00
data.dcs.shef.ac.uk/	99.55	100.00	100.00	99.61	99.84	100.00	100.00	73.14	100.00
worldbank.270a.info/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	66.75	100.00
webscience.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	93.33	100.00	62.41	0.00
lobid.org/organisation	96.70	100.00	100.00	100.00	100.00	91.67	32.51	76.98	100.00
dbtune.org/jamendo/	62.13	100.00	100.00	99.93	100.00	97.11	100.00	48.54	100.00
govwild.hpi-web.de/project/ govwild-sources	75.00	100.00	100.00	100.00	100.00	100.00	100.00	88.06	100.00
cordis.rkbexplorer.com	92.69	100.00	100.00	100.00	100.00	100.00	100.00	50.00	82.76
courseware.rkbexplorer.com	99.67	100.00	100.00	100.00	100.00	100.00	100.00	52.66	100.00
jisc.rkbexplorer.com	78.55	100.00	100.00	79.48	100.00	99.90	100.00	50.96	100.00
wiki.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
data.bibsys.no/data	99.99	100.00	100.00	100.00	100.00	100.00	100.00	69.23	99.99
aims.fao.org/standards/ agrovoc/about	96.71	100.00	100.00	100.00	100.00	100.00	32.51	65.42	100.00
data.reegle.info/	57.73	100.00	100.00	76.07	100.00	98.38	100.00	84.91	100.00
la.indymedia.org/syn/	100.00	100.00	99.29	100.00	100.00	100.00	100.00	54.87	100.00
darmstadt.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.05	100.00
www.josemalvarez.es/web/ 2011/11/01/nomenclator-asturias-2010/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	54.94	90.72
dblp.rkbexplorer.com	57.60	100.00	100.00	100.00	100.00	100.00	100.00	45.17	74.60

Continued on next page . . .

	Intrinsic Category								
Namespace	**CN2**	**CS1**	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**	**CS9**	**SV3**
webenemasuno.linkeddata.es/	90.24	100.00	100.00	100.00	97.41	100.00	46.45	54.01	99.99
energy.psi.enakting.org	87.60	100.00	100.00	100.00	100.00	89.29	7.41	53.51	100.00
education.data.gov.uk/	88.07	100.00	100.00	100.00	100.00	100.00	95.45	63.31	100.00
unlocode.rkbexplorer.com	99.25	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
msc2010.org/mscwork/	94.51	100.00	100.00	100.00	100.00	100.00	100.00	84.29	100.00
lingweb.eva.mpg.de/ids/	78.60	100.00	100.00	100.00	100.00	100.00	100.00	58.35	100.00
laas.rkbexplorer.com	98.55	100.00	100.00	100.00	100.00	100.00	100.00	50.17	100.00
irit.rkbexplorer.com	99.44	100.00	100.00	100.00	100.00	100.00	100.00	50.02	100.00
ft.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.66	100.00
extbi.lab.aau.dk/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	90.63	100.00
vocabulary.semantic-web.at/ AustrianSkiTeam	49.68	100.00	100.00	100.00	100.00	96.92	43.74	73.15	100.00
www.bibsonomy.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	86.33	100.00
opendatacommunities.org/ datasets/geography	100.00	100.00	100.00	100.00	100.00	100.00	100.00	64.33	100.00
purl.org/NET/rdflicense	76.71	100.00	99.99	100.00	100.00	100.00	39.37	49.79	99.92
rhizomik.net/semanticxbrl/	100.00	0.00	100.00	100.00	100.00	100.00	100.00	0.00	100.00
www.lexvo.org	94.24	100.00	100.00	83.76	100.00	100.00	100.00	80.37	100.00
deepblue.rkbexplorer.com	92.45	100.00	100.00	100.00	100.00	100.00	100.00	51.66	100.00
eurecom.rkbexplorer.com	95.80	100.00	100.00	100.00	100.00	100.00	100.00	50.09	100.00
pisa.rkbexplorer.com	94.95	100.00	100.00	100.00	100.00	100.00	100.00	50.46	100.00
www.w3.org/TR/wordnet-rdf	92.90	100.00	100.00	100.00	100.00	100.00	100.00	60.82	100.00
vocabulary.wolterskluwer.de/court	99.83	100.00	-	100.00	100.00	-	100.00	66.84	-
curriculum.rkbexplorer.com	97.57	100.00	100.00	100.00	100.00	100.00	100.00	52.38	100.00
rae2001.rkbexplorer.com	92.10	100.00	100.00	100.00	100.00	100.00	100.00	50.02	100.00
ulm.rkbexplorer.com	94.70	100.00	100.00	100.00	100.00	100.00	100.00	50.75	100.00
umbel.org	95.71	100.00	100.00	100.00	100.00	100.00	100.00	99.51	100.00
ibm.rkbexplorer.com	97.43	100.00	100.00	100.00	100.00	100.00	100.00	50.29	100.00
newcastle.rkbexplorer.com	99.78	100.00	100.00	100.00	100.00	100.00	100.00	50.02	100.00
vocabulary.semantic-web.at/ PoolParty/wiki/OpenData	98.29	100.00	100.00	99.73	100.00	95.92	100.00	65.62	100.00
budapest.rkbexplorer.com/	96.57	100.00	100.00	100.00	100.00	100.00	100.00	50.44	100.00
epsrc.rkbexplorer.com	98.79	100.00	100.00	100.00	100.00	100.00	100.00	54.30	100.00
lod.taxonconcept.org/	64.74	100.00	100.00	100.00	100.00	100.00	-10.43	54.67	25.12
id.ndl.go.jp/auth/ndla	99.75	100.00	100.00	100.00	100.00	100.00	100.00	45.66	100.00
deploy.rkbexplorer.com	65.30	100.00	100.00	100.00	100.00	100.00	100.00	50.10	100.00
lisbon.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	51.50	100.00
aemet.linkeddata.es/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	81.09	100.00

Continued on next page . . .

	Intrinsic Category								
Namespace	**CN2**	**CS1**	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**	**CS9**	**SV3**
roma.rkbexplorer.com	97.64	100.00	100.00	100.00	100.00	100.00	100.00	50.33	100.00
datos.fundacionctic.org/en	57.60	100.00	100.00	100.00	-	100.00	98.82	-	100.00
europeana.eu	99.99	100.00	100.00	97.61	100.00	100.00	100.00	78.20	-
italy.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	51.89	100.00
wals.info/	99.89	100.00	100.00	100.00	100.00	100.00	100.00	63.01	100.00
southampton.rkbexplorer.com	100.00	100.00	100.00	100.00	100.00	100.00	100.00	45.96	100.00
greek-lod.math.auth.gr/fire-brigade/	100.00	100.00	100.00	91.05	100.00	22.54	66.62	47.40	100.00
vocabulary.semantic-web.at/ PoolParty/wiki/semweb	99.91	100.00	100.00	100.00	100.00	100.00	100.00	58.13	100.00
linkedct.org/	99.88	100.00	100.00	100.00	100.00	100.00	100.00	56.25	100.00
kaunas.rkbexplorer.com	89.13	100.00	100.00	100.00	100.00	100.00	100.00	51.16	100.00
ieee.rkbexplorer.com	87.98	100.00	100.00	100.00	100.00	100.00	100.00	50.01	100.00
vivo.iu.edu	83.41	100.00	100.00	94.34	100.00	100.00	38.16	70.25	98.79
acm.rkbexplorer.com/	75.40	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
risks.rkbexplorer.com	95.16	100.00	100.00	100.00	100.00	100.00	100.00	50.01	100.00
ieeevis.tw.rpi.edu	88.70	100.00	100.00	100.00	100.00	100.00	100.00	54.27	100.00
pdev.org.uk/pdevlemon/	63.69	100.00	100.00	100.00	100.00	100.00	86.67	55.24	99.98
greek-lod.math.auth.gr/police/	99.95	100.00	100.00	100.00	100.00	22.00	99.96	38.66	100.00
minsky.gsi.dit.upm.es/ semanticwiki/index.php/Main_Page	91.28	100.00	100.00	100.00	100.00	100.00	88.24	55.08	100.00
nsf.rkbexplorer.com	91.03	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
citeseer.rkbexplorer.com/	80.20	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
prefix.cc/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.00	100.00
kent.zpr.fer.hr:8080/ educationalProgram/	88.11	100.00	100.00	100.00	100.00	97.98	100.00	50.83	100.00
transport.data.gov.uk/	99.99	100.00	100.00	100.00	100.00	100.00	100.00	50.00	99.99
www.lingvoj.org/	100.00	100.00	100.00	100.00	100.00	100.00	100.00	50.00	-

Results of Quality Assessment for Every Assessed Dataset - Showing Intrinsic Category Results