



Enabling Self-Service Data Provisioning Through Semantic Enrichment of Data

Ahmad Assaf

A doctoral dissertation submitted to:

TELECOM ParisTech

in partial fulfillment of the requirements for the degree of:

Doctor of Philosophy

Specialty : COMPUTER SCIENCE AND MULTIMEDIA

Supervisor:

Dr. Raphaël TRONCY - EURECOM, France

Dr. Aline SENART - SAP, France

*To all those who helped me
to make this dream coming true.*

*“Things don’t have to change
the world to be important”.*

-Steve Jobs -

Acknowledgments

Working as a PhD student in EURECOM and SAP was a great experience that would not be achieved without the help and support of many people, who I would like to acknowledge here.

First and foremost, I would like to thank my supervisors Dr. Raphaël Troncy and Dr. Aline Senart for their invaluable support and great guidance throughout my studies. I would like to express my gratitude to them for providing me with the freedom to pursue my research and the valuable feedback along the way. This work would not have been possible without their scientific knowledge, constructive advice and deep compassion.

I would like to thank my committee members, the reviewers Prof. XXX and Dr. XXXX, and furthermore the examiners Dr. XX and Dr. XXX for their precious time, shared positive insight and guidance.

I owe my deepest gratitude to my parents, Dr. Abdel Mouti Assaf and Renad Al Fahoum for their unwavering encouragement, devotion and love. A great thanks goes to my sisters Malak, Dima and Noor for pushing me always to be the best. Last but not least, special thanks go to my friends and colleagues in SAP and EURECOM for their constant friendship, moral and infinite support.

Abstract

Enterprises use a wide range of heterogeneous information systems in their business activities such as Enterprise Resource Planning (ERP), Customer Relationships Management (CRM) and Supply Chain Management (SCM) systems. An enterprise distributed IT landscape contains multiple systems using different technologies and data standards. In addition to this heterogeneity, the amount of information in enterprise databases and on-line data stores expands exponentially each year. Enterprise Big Data isn't big in volume only, but in the associated file formats. The information is also often stored often in unstructured and unknown formats.

Data integration is the problem of combining data residing at different sources, and providing the user with a unified view of these data. In large enterprises, it is a time and resource costly task. Various approaches have been introduced to solve this integration challenge. These approaches were primarily based on XML as the data representation syntax, Web Services to provide the data exchange protocols and Service Oriented Architecture (SOA) as a holistic approach for distributed systems architecture and communication. However, it was found that these technologies are no sufficient to solve the integration problems in large enterprises. Recently, ontology-based data integration approaches have been suggested where ontologies are used to describe the data, queries and mappings between them. A slightly different approach is the use of the Linked Data (LD) paradigm for integrating enterprise data.

Classic Business Intelligence (BI) and even the newer Agile Visualization tools focus much of their selling features on attractive and unique visualizations, but preparing data for those visualizations still remains the far more challenging task in most BI projects large and small. self-service data provisioning aims at tackling this problem by providing intuitive datasets discovery, acquisition and integration techniques intuitively to the end user.

The goal of this thesis is to provide a framework that enables self-service data provisioning in the enterprise. The main goal is to empower users to search, inspect and reuse data through semantically enriched datasets profiles.

The increasing diversity of the datasets makes it difficult to annotate them with a fixed number of pre-defined tags. Moreover, manually entered tags are subjective and may not capture the essence and breadth of the dataset. We propose a mechanism to bootstrap the process of attaching meta information to data objects by leveraging knowledge bases like DBpedia and Freebase.

In many knowledge bases, entities are described with numerous properties. However, not all properties have the same importance. Some properties are considered as keys for performing instance matching tasks while other properties are generally chosen for quickly providing a summary of the key facts attached to an entity. We propose a method to select what properties should be used when depicting the

summary of an entity, for example when augmenting extra columns into an existing dataset or when annotating instances with semantic tags.

Linked Open Data (LOD) has emerged as one of the largest collections of inter-linked datasets on the web. In order to benefit from this mine of data, one needs to access to descriptive information about each dataset (or metadata). This metadata enables dataset discovery, understanding, integration and maintenance. Data portals, which are considered to be datasets' access points, offer metadata represented in different and heterogenous models. We first propose a Harmonized Dataset model based on a systematic literature survey. Second, we discovered that rich metadata information is currently very limited to a few data portals where they are usually provided manually, thus being often incomplete and inconsistent in terms of quality. We propose a scalable automatic approach for extracting, validating, correcting and generating descriptive linked dataset profiles. This approach applies several techniques in order to check the validity of the metadata provided and to generate descriptive and statistical information for a particular dataset or for an entire data portal.

Traditional data quality is a thoroughly researched field with several benchmarks and frameworks to grasp its dimensions. Ensuring data quality in Linked Open Data is much more complex. It consists of structured information supported by models, ontologies and vocabularies and contains queryable endpoints and links. We propose an objective assessment framework for Linked Data quality based on quality metrics that can be automatically measured. We further present an extensible quality measurement tool that helps on one hand data owners to rate the quality of their datasets and get some hints on possible improvements, and on the other hand data consumers to choose their data sources from a ranked set.

Finally, the Internet has created a paradigm shift in how we consume and disseminate information. Data nowadays is spread over heterogeneous silos of archived and live data. People willingly share data on social media by posting news, views, presentations, pictures and videos. We propose a service that combines services available on the web to aggregate social news. It brings live and archived information to the user that is directly related to his active page. The key advantage is an instantaneous access to complementary information without the need to search for it. Information appears when it is relevant enabling the user to focus on what is really important.

Contents

Acknowledgements	ii
Abstract	iii
Contents	vi
List of Figures	ix
List of Tables	xi
Acronyms	xiii
1 Introduction	1
1.1 Context and Motivation	1
2 Background	3
2.1 Conclusion	3
I Open Data Integration in the Enterprise	5
3 Data Aggregation and Modeling	9
3.1 Introduction	9
3.2 Data Portals and Dataset Models	11
3.2.1 DCAT	11
3.2.2 DCAT-AP	11
3.2.3 ADMS	11
3.2.4 VoID	12
3.2.5 CKAN	12
3.2.6 DKAN	12
3.2.7 Socrata	13
3.2.8 Schema.org	13
3.2.9 Project Open Data	13
3.3 Metadata Classification	14
3.4 Towards A Harmonized Model	16
3.5 Conclusion and Future Work	18
II Towards A complete Dataset Profile	23
4 Data Aggregation and Modeling	25
4.1 Introduction	25
4.2 Motivation	27
4.3 Related Work	27
4.4 Profiling Data Portals	30

4.4.1	Data Portal Identification	30
4.4.2	Metadata Extraction	31
4.4.3	Instance and Resource Extraction	32
4.4.4	Profile Validation	33
4.4.5	Profile and Report Generation	34
4.5	Experiments and Evaluation	36
4.5.1	Experimental Setup	36
4.5.2	Profiling Correctness	38
4.5.3	Profiling Completeness	39
4.6	Experiments and Evaluation	39
4.6.1	Experimental Setup	39
4.6.2	Results and Evaluation	40
4.6.3	General information	41
4.6.4	Access information	41
4.6.5	Ownership information	42
4.6.6	Provenance information	42
4.6.7	Enriched Profiles	42
4.7	Conclusion and Future Work	43
5	Data Aggregation and Modeling	45
5.1	Introduction	45
5.2	Data Quality Assessment	46
5.3	Objective Linked Data Quality Classification	48
5.3.1	Completeness	50
5.3.2	Availability	51
5.3.3	Correctness	51
5.3.4	Consistency	52
5.3.5	Freshness	52
5.3.6	Provenance	52
5.3.7	Licensing	52
5.3.8	Comprehensibility	52
5.3.9	Coherence	53
5.3.10	Security	53
5.4	An Extensible Objective Quality Assessment Framework	53
5.4.1	Quality Score Calculation	54
5.4.2	Experiments and Analysis	55
5.5	Linked Data Quality Tools	57
5.5.1	Information Quality	57
5.5.2	Modeling Quality	58
5.5.3	Dataset Quality	59
5.5.4	Queryable End-point Quality	63

5.6	Conclusions and Future Work	64
6	Conclusions and Future Perspectives	65
6.1	Achievements	65
6.2	Perspectives	65
A	List of Publications	67
A.1	Journals	67
A.2	Conferences and Workshops	67
	Bibliography	69

List of Figures

4.1	Processing pipeline for validating and generating dataset profiles . . .	30
4.2	Error % by section	42
4.3	Error % by information type	42
5.1	Processing pipeline for objective dataset quality assessment	54
5.2	Average Error % per quality indicator for LOD group	57

List of Tables

3.1	Data models sections mapping	16
4.1	Summary of the experiments details	37
4.2	Datasets chosen for the correctness evaluation	38
4.3	Groups chosen for the correctness evaluation	38
4.4	Top metadata fields error % by type	40
5.1	Objective Linked Data Quality Framework	49
5.2	Objective Quality Assessment Methods for CKANbased Data Portals	55

Glossary

Here are the main acronyms used in this document. The meaning of an acronym is usually indicated once, when it first appears in the text.

ERP	Enterprise Resource Planning
CRM	Customer Relationships Management
SCM	Supply Chain Management
LOD	Linked Open Data
SOA	Service Oriented Architecture
LD	Linked Data
BI	Business Intelligence
API	Application Programming Interface
FOAF	Friend of a friend
GA	Genetic Algorithm
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
IR	Information Retrieval
JSON	JavaScript Object Notation
LDA	Latent Dirichlet Allocation
NE	Named Entity
NER	Named Entity recognition
NERD	Named Entity Recognition and Disambiguation
NLP	Natural Language Processing
OWL	Web Ontology Language
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
REST	Representational State Transfer
SKOS	Simple Knowledge Organization System
SPARQL	Query Language for RDF
URI	Universal Resource Identifier
URL	Universal Resource Locator
W3C	World Wide Web Consortium
XML	Extensible Markup Language

Introduction

1.1 Context and Motivation

Chapter 6 concludes the presented work and outlines new research directions.

Background

2.1 Conclusion

Part I

Open Data Integration in the Enterprise

Overview of Part ??

In Part ??,

In Chapter 5,

In Chapter ??,

Data Aggregation and Modeling

3.1 Introduction

Open data is the data that can be easily discovered, reused and redistributed by anyone. It can include anything from statistics, geographical data, meteorological data to digitized books from libraries. Open data should have both legal and technical dimensions. It should be placed in the public domain under liberal terms of use with minimal restrictions and should be available in electronic formats that are non-proprietary and machine readable. Open Data has major benefits for citizens, businesses, society and governments: it increases transparency and enables self-empowerment by improving the visibility of previously inaccessible information; it allows citizens to be better informed about policies, public spending and activities in the law making processes. Moreover, it is still considered as a gold mine for organizations which are trying to leverage external data sources in order to produce more informed business decisions [17], despite the legal issues surrounding Linked Data licenses [52].

The Linked Data publishing best practices [20] specifies that datasets should contain metadata needed to effectively understand and use them. *Metadata* is structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource [87]. Having rich metadata helps in enabling:

- **Data discovery, exploration and reuse:** In [99], it was found that users are facing difficulties finding and reusing publicly available datasets. Metadata provides an overview of datasets making them more searchable and accessible. High quality metadata can be at times more important than the actual raw data especially when the costs of publishing and maintaining such data is high.
- **Organization and identification:** The increasing number of datasets being published makes it hard to track, organize and present them to users efficiently. Attached metadata helps in bringing similar resources together and distinguish useful links.
- **Archiving and preservation:** There is a growing concern that digital resources will not survive in usable forms to the future [87]. Metadata can ensure resources survival and continuous accessibility by providing clear provenance information to track the lineage of digital resources and detail their physical characteristics.

The value of Open Data is recognized when it is used. To ensure that, publishers need to enable people to find datasets easily. Data portals are specifically designed for this purpose. They make it easy for individuals and organizations to store, publish and discover datasets. The data portals can be public like Datahub¹ and the Europe's Public Data portal² or private like Quandl³ and Engima⁴. The data available in private portals is of higher quality as it is manually curated but in lesser quantity compared to what is available in public portals. Similarly, in some public data portals, administrators manually review datasets information, validate, correct and attach suitable metadata information.

Data models vary across data portals. While exhaustively surveying the range of data models, we did not find any that offers enough granularity to completely describe complex datasets facilitating search, discovery and recommendation. For example, the Datahub uses an extension of the Data Catalog Vocabulary (DCAT) [38] which prohibits a semantically rich representation of complex datasets like DBpedia⁵ that has multiple endpoints and thousands of dump files with content in several languages [73]. Moreover, to properly integrate Open Data into business, a dataset should include the following information:

- *Access information*: a dataset is useless if it does not contain accessible data dumps or query-able endpoints;
- *License information*: businesses are always concerned with the legal implications of using external content. As a result, datasets should include both machine and human readable license information that indicates permissions, copyrights and attributions;
- *Provenance information*: depending on the dataset license, the data might not be legally usable if there are no information describing its authoritative and versioning information. Current models under-specify these aspects limiting the usability of many datasets.

In this paper, we perform a comprehensive survey of the main data portals and dataset models, that is: CKAN, DKAT, Public Open Data, Socrata, VoID, DCAT and Schema.org. We further analyze these models and suggest a classification for metadata information. Based on this classification, we propose HDL, an harmonized dataset model that addresses the shortcomings of existing dataset models. The remainder of the paper is structured as follows. In Section 3.2, we present the existing dataset models used by various data portals. In Section 3.3, we present our classification for the different metadata information. In Section 3.4, we describe our proposed

¹<http://datahub.io>

²<http://publicdata.eu>

³<https://quandl.com/>

⁴<http://enigma.io/>

⁵<http://dbpedia.org>

model and suggest a set of best practices to ensure proper metadata presentation and we finally conclude and outline some future work in Section 5.6.

3.2 Data Portals and Dataset Models

There are many data portals that host a large number of private and public datasets. Each portal present the data based on a model used by the underlying software. In this section, we present the results of our landscape survey of the most common data portals and dataset models.

3.2.1 DCAT

The Data Catalog Vocabulary (DCAT) is a W3C recommendation that has been designed to facilitate interoperability between data catalogs published on the Web [38]. The goal behind DCAT is to increase datasets discoverability enabling applications to easily consume metadata coming from multiple sources. Moreover, the authors foresee that aggregated DCAT metadata can facilitate digital preservation and enable decentralized publishing and federated search.

DCAT is an RDF vocabulary defining three main classes: `dcat:Catalog`, `dcat:Dataset` and `dcat:Distribution`. We are interested in both the `dcat:Dataset` class which is a collection of data that can be available for download in one or more formats and the `dcat:Distribution` class which describes the method with which one can access a dataset (e.g. an RSS feed, a REST API or a SPARQL endpoint).

3.2.2 DCAT-AP

The DCAT application profile for data portals in Europe (DCAT-AP)⁶ is a specialization of DCAT to describe public section datasets in Europe. It defines a minimal set of properties that should be included in a dataset profile by specifying mandatory and optional properties. The main goal behind it is to enable cross-portal search and enhance discoverability. DCAT-AP has been promoted by the Open Data Support⁷ to be the standard for describing datasets and catalogs in Europe.

3.2.3 ADMS

The Asset Description Metadata Schema (ADMS) [84] is also a profile of DCAT. It is used to semantically describe assets. An asset is broadly defined as something that can be opened and read using familiar desktop software (e.g. code lists, taxonomies, dictionaries, vocabularies) as opposed to something that needs to be processed like raw data. While DCAT is designed to facilitate interoperability between data catalogs, ADMS is focused on the assets within a catalog.

⁶https://joinup.ec.europa.eu/asset/dcat_application_profile/description

⁷<http://opendatasupport.eu>

3.2.4 VoID

VoID [24] is another RDF vocabulary designed specifically to describe linked RDF datasets and to bridge the gap between data publishers and data consumers. In addition to dataset metadata, VoID describes the links between datasets. VoID defines three main classes: `void:Dataset`, `void:Linkset` and `void:subset`. We are specifically interested in the `void:Dataset` concept. VoID conceptualizes a dataset with a social dimension. A VoID dataset is a collection of raw data, talking about one or more topics, originates from a certain source or process and accessible on the web.

3.2.5 CKAN

CKAN⁸ is the world's leading open-source data management system (DMS). It helps users from different domains (national and regional governments, companies and organizations) to easily publish their data through a set of workflows to publish, share, search and manage datasets. CKAN is the portal powering web sites like Datahub, the Europe's Public Data portal or the U.S Government's open data portal⁹.

CKAN is a complete catalog system with an integrated data storage and powerful RESTful JSON API. It offers a rich set of visualization tools (e.g. maps, tables, charts) as well as an administration dashboard to monitor datasets usage and statistics. CKAN allows publishing datasets either via an import feature or through a web interface. Relevant metadata describing the dataset and its resources as well as organization related information can be added. A Solr¹⁰ index is built on top of this metadata to enable search and filtering.

The CKAN data model¹¹ contains information to describe a set of entities (dataset, resource, group, tag and vocabulary). CKAN keeps the core metadata restricted as a JSON file, but allows for additional information to be added via "extra" arbitrary key/value fields. CKAN supports Linked Data and RDF as it provides a complete and functional mapping of its model to Linked Data formats.

3.2.6 DKAN

DKAN¹² is a Drupal-based DMS with a full suite of cataloging, publishing and visualization features. Built over Drupal, DKAN can be easily customized and extended. The actual data sets in DKAN can be stored either within DKAN or on external sites. DKAN users are able to explore, search and describe datasets through the web interface or a RESTful API.

⁸<http://ckan.org>

⁹<http://data.gov>

¹⁰<http://lucene.apache.org/solr/>

¹¹<http://docs.ckan.org/en/ckan-1.8/domain-model.html>

¹²<http://nucivic.com/dkan/>

The DKAN data model¹³ is very similar to the CKAN one, containing information to describe datasets, resources, groups and tags.

3.2.7 Socrata

Socrata¹⁴ is a commercial platform to streamline data publishing, management, analysis and reusing. It empowers users to review, compare, visualize and analyze data in real time. Datasets hosted in Socrata can be accessed using RESTful API that facilitates search and data filtering.

Socrata allows flexible data management by implementing various data governance models and ensuring compliance with metadata schema standards. It also enables administrators to track data usage and consumption through dashboards with real-time reporting. Socrata is very flexible when it comes to customizations. It has a consumer-friendly experience giving users the opportunity to tell their story with data. Socrata's data model is designed to represent tabular data: it covers a basic set of metadata properties and has good support for geospatial data.

3.2.8 Schema.org

Schema.org¹⁵ is a collection of schemas used to markup HTML pages with structured data. This structured data allows many applications, such as search engines, to understand the information contained in Web pages, thus improving the display of search results and making it easier for people to find relevant data.

Schema.org covers many domains. We are specifically interested in the **Dataset** schema. However, there are many classes and properties that can be used to describe organizations, authors, etc.

3.2.9 Project Open Data

Project Open Data (POD)¹⁶ is an online collection of best practices and case studies to help data publishers. It is a collaborative project that aims to evolve as a community resource to facilitate adoption of open data practices and facilitate collaboration and partnership between both private and public data publishers.

The POD metadata model¹⁷ is based on DCAT. Similarly to DCAT-AP, POD defines three types of metadata elements: Required, Required-if(conditionally required) and Expanded (optional). The metadata model is presented in the JSON format and encourages publishers to extend their metadata descriptions using elements from the "Expanded Fields" list, or from any well-known vocabulary.

¹³<http://docs.getdkan.com/dkan-documentation/dkan-developers/dataset-technical-field-reference/>

¹⁴<http://socrata.com>

¹⁵<http://schema.org>

¹⁶<http://project-open-data.cio.gov/>

¹⁷<https://project-open-data.cio.gov/v1.1/schema/>

3.3 Metadata Classification

A dataset metadata model should contain sufficient information so that consumers can easily understand and process the data that is described. After analyzing the models described in the section 3.2, we find out that a dataset can contain four main sections:

- **Resources:** The actual raw data that can be downloaded or accessed directly via queryable endpoints. Resources can come in various formats such as JSON, XML or RDF.
- **Tags:** Descriptive knowledge about the dataset content and structure. This can range from simple textual representation to semantically rich controlled terms. Tags are the basis for datasets search and discovery.
- **Groups:** Groups act as organizational units that share common semantics. They can be seen as a cluster or a curation of datasets based on shared categories or themes.
- **Organizations:** Organizations are another way to arrange datasets. However, they differ from groups as they are not constructed by shared semantics or properties, but solely on the dataset's association to a specific administration party.

Upon closed examination of the various data models, we group the metadata information into eight main types. Each section discussed above should contain one or more of these types. For example, resources have general, access, ownership and provenance information while tags have general and provenance information only. The eight information types are:

- **General information:** The core information about the dataset (e.g., title, description, ID). The most common vocabulary used to describe this information is Dublin Core¹⁸.
- **Access information:** Information about dataset access and usage (e.g., URL, license title and license URL). In addition to the properties in the models discussed above, there are several vocabularies designed specially to describe data access right e.g. Linked Data Rights¹⁹, the Open Digital Rights Language (ODRL)²⁰.
- **Ownership information:** Authoritative information about the dataset (e.g. author, maintainer and organization). The common vocabularies used to ex-

¹⁸<http://dublincore.org/documents/dcmi-terms/>

¹⁹<http://oeg-dev.dia.fi.upm.es/licensius/static/ldr/>

²⁰<http://www.w3.org/ns/odrl/2/>

pose ownership information are Friend-of-Friend (FOAF)²¹ for people and relationships, vCard [88] for people and organizations and the Organization ontology [29] designed specifically to describe organizational structures.

- **Provenance information:** Temporal and historical information about the dataset creation and update records, in addition to versioning information (e.g. creation data, metadata update data, latest version). Provenance information coverage varies across the modeled surveyed. However, its great importance lead to the development of various special vocabularies like the Open Provenance Model²² and PROV-O [96]. DataID [73] is an effort to provide semantically rich metadata with focus on providing detailed provenance, license and access information.
- **Geospatial information:** Information reflecting the geographical coverage of the dataset represented with coordinates or geometry polygons. There are several additional models and extensions specifically designed to express geographical information. The Infrastructure for Spatial Information in the European Community (INSPIRE) directive²³ aims at establishing an infrastructure for spatial information. Mappings have been made between DCAT-AP and the INSPIRE metadata. CKAN provides as well a spatial extension²⁴ to add geospatial capabilities. It allows importing geospatial metadata from other resources and supports various standards (e.g. ISO 19139) and formats (e.g. GeoJSON).
- **Temporal information:** Information reflecting the temporal coverage of the dataset (e.g. from date to date). There has been some notable work on extending CKAN to include temporal information. `govdata.de` is an Open Data portal in Germany that extends the CKAN data model to include information like `temporal_granularity`, `temporal_coverage_to` and `temporal_granularity_from`.
- **Statistical information:** Statistical information about the data types and patterns in datasets (e.g. properties distribution, number of entities and RDF triples). This information is particularly useful to explore a dataset as it gives detailed insights about the raw data when provided properly. VoID is the only model that provides statistical information about a dataset. VoID defines properties to express different statistical characteristics of datasets like the total number of triples, total number of entities, total number of distinct classes, etc. However, there are other vocabularies such as SCOVO [77] that can model and publish statistical data about datasets.

²¹<http://xmlns.com/foaf/spec/>

²²<http://open-biomed.sourceforge.net/opmv/>

²³<http://inspire.ec.europa.eu/>

²⁴<https://github.com/ckan/ckanext-spatial>

- **Quality information:** Information that indicates the quality of the dataset on the metadata and instance levels. In addition to that, a dataset should include an openness score that measures its alignment with the Linked Data publishing standards [95]. Quality information is only expressed in the POD metadata. However, `govdata.de` extends the CKAN model also to include a `ratings_average` field. Moreover, there are various other vocabularies like daQ [30] that can be used to express datasets quality. The RDF Review Vocabulary²⁵ can also be used to express reviews and ratings about the dataset or its resources.

3.4 Towards A Harmonized Model

Since establishing a common vocabulary or model is the key to communication, we identified the need for an harmonized dataset metadata model containing sufficient information so that consumers can easily understand and process datasets. To create the mappings between the different models, we performed various steps:

- Examine the model or vocabulary specification and documentation.
- Examine existing datasets using these models and vocabularies. <http://dataportals.org> provides a comprehensive list of Open Data Portals from around the world. It was our entry point to find out portals using CKAN or DKAN as their underlying DMS. We also investigated portals known to be using specific DMS. Socrata, for example, maintains a list of Open Data portals using their software on their homepage such as <http://pencolorado.org> and <http://data.maryland.gov>.
- Examine the source code of some portals. This was specifically the case for Socrata as their API returns the raw data serialized as JSON rather than the dataset's metadata. As a consequence, we had to investigate the Socrata Open Data API (SODA) source code²⁶ and check the different classes and interfaces.

CKAN	DKAN	POD	DCAT	VoID	Schema.org	Socrata
resources	resources	distribution	dcat:Distribution	void:Dataset → void:dataDump	CreativeWork:keywords	attachments
tags	tags	keyword	dcat:Dataset → :keyword	void:Dataset → :keyword	Dataset:distribution	tags
groups	groups	theme	dcat:Dataset → :theme	-	CreativeWork:about	category
organization	organization	publisher	dcat:Dataset → :publisher	void:Dataset → :publisher	-	-

Table 3.1: Data models sections mapping

The first task is to map the four main information sections across those models. Table 3.1 shows our proposed mappings. For the ontologies (DCAT, VoID), the first part represents the class and the part after `→` represents the property. For

²⁵<http://vocab.org/review/>

²⁶<https://github.com/socrata/soda-java/tree/master/src/main/java/com/socrata/model>

Schema.org, the first part refers to the schema and the second part after : refers to the property.

Table ?? presents the full mappings between the models across the information groups. Entries in the CKAN marked with * are properties from CKAN extensions and not included in the original data model. Similar to the sections mappings, for the ontologies (DCAT, VoID), the first part represents the class and the part after → represents the property. However, sometimes the part after → refers to another resource. For example, to describe the dataset’s maintainer email in DCAT, the information should be presented in the `dcat:Dataset` class using the `dcat:contactPoint` property. However, the range of this property is a resource of type `vcard` which has the property `hasEmail`.

For Schema.org, similar to the sections mapping, the first part refers to the schema and the second part after : refers to the property. However, if the property is inherited from another schema we denote that by using a → as well. For example, the size of a dataset is a property for a `Dataset` schema specified in its `distribution` property. However, the type of `distribution` is `dataDownload` which is inherited from the `MediaObject` schema. The size for `MediaObject` is defined in its `contentSize` property which makes the mapping string `Dataset:distribution→DataDownload→MediaObject:contentSize`.

Examining the different models, we noticed a lack of a complete model that covers all the information types. There is an abundance of extensions and application profiles that try to fill in those gaps, but they are usually domain specific addressing specific issues like geographic or temporal information. To the best of our knowledge, there is still no complete model that encompasses all the described information types.

HDL aims at filling this gap by taking the best from these models. HDL is currently modeled in JSON²⁷ but converting it to a standalone OWL ontology is part of our future work.

The CKAN model controls the values to be used in describing some dataset properties. For example, the `resource_type` property can have the values: `file`: direct accessible bitstream, `file.upload`: file uploaded to the CKAN FileStore²⁸, `api`, `visualization`, `code`: the actual source code or a reference to a code repository and `documentation`. However, using the Roomba tool [3], we managed to generate portal-wide reports about the representation of various fields in CKAN portals. The goal behind these reports is to find what are the frequent fields data publishers are adding as `extras` fields.

We created two “key:object meta-field values” reports using Roomba. The first one aims to collect the list of `extras` values using the query string `extras>value:extras>name` and the second one is to list the file types specified for resources using the query string `resources>resource_type:resources>name`. We

²⁷<https://github.com/ahmadassaf/.opendata-checker/blob/master/model/hdl.json>

²⁸<http://docs.ckan.org/en/ckan-1.8/filestore.html>

run the report generation process on two prominent data portals: the Linked Open Data (LOD) cloud hosted on the Datahub containing 259 datasets and the Africa's largest open data portal, OpenAfrica²⁹ that contains 1653 datasets.

After examining the results, we noticed that for OpenAfrica, 53% of the datasets have contain additional information about the geographical coverage of the dataset (e.g. `spatial-reference-system`, `spatial_harvester`, `bbox-east-long`, `bbox-north-long`, `bbox-south-long`, `bbox-west-long`). In addition, 16% of the datasets have additional provenance and ownership information (e.g `frequency-of-update`, `dataset-reference-date`). For the LOD cloud, the main information embedded in the `extras` fields are about the structure and statistical distribution of the dataset (e.g. `namespace`, `number of triples` and `links`). The OpenAfrica resources did not specify any extra resource types. However, in the LOD cloud, we observe that multiple resources define additional types (e.g. `example`, `api/sparql`, `publication`, `example`).

Roomba easily enables to perform such tests and to gather a detailed view about the kind of missing information data publishers require in the core model. We further plan to run Roomba on various portals to collect more information about such missing data to include it in HDL.

3.5 Conclusion and Future Work

In this paper, we surveyed the landscape of various models and vocabularies that described datasets on the web. Since establishing a common vocabulary or model is the key to communication, we identified the need for an harmonized dataset metadata model containing sufficient information so that consumers can easily understand and process datasets. We have identified four main sections that should be included in the model: resources, groups, tags and organizations. Furthermore, we have classified the information to be included into eight types. Our main contribution is a set of mappings between each properties of those models. This has lead to the design of HDL, an harmonized dataset model, that takes the best out of these models and extends them to ensure complete metadata coverage to enable data discovery, exploration and reuse.

At the moment, HDL is available as a hierarchical JSON file. As part of our future work, we plan to refine HDL and present it as a fully fledged OWL ontology. At the moment, HDL contains some values that were frequently defined in CKAN `extras` fields. However, we plan to broaden our analysis of these values by running Roomba on additional portals and present the top results as enumerations, ensuring a fine-grained representation of a dataset. We further plan to create mappings between HDL and all the various models to ensure full compatibility. These mappings, for example, can be used to extend Roomba allowing it to perform metadata profiling on other portals like DKAN. Finally, we plan to create a set of supporting tools that

²⁹<http://africaopendata.org/>

allow validation of generation of HDL profiles.

Conclusion of Part ??

Part II

Towards A complete Dataset Profile

Data Aggregation and Modeling

4.1 Introduction

From 12 datasets cataloged in 2007, the Linked Open Data cloud has grown to nearly 1000 datasets containing more than 82 billion triples¹ [22]. Data is being published by both the public and private sectors and covers a diverse set of domains from life sciences to media or government data. The Linked Open Data cloud is potentially a gold mine for organizations and individuals who are trying to leverage external data sources in order to produce more informed business decisions [17]. This success lies in the cooperation between data publishers and consumers. Consumers are empowered to find, share and combine information in their applications easily. However, the heterogeneous nature of data sources reflects directly on the data quality as these sources often contain inconsistent as well as misinterpreted and incomplete metadata information. Considering the significant variation in size, the languages used and the freshness of the data, one realizes that finding useful datasets without prior knowledge is increasingly complicated. This can be clearly noticed in the LOD Cloud where few datasets such as DBPedia [21], Freebase [16] and YAGO [93] are favored over less popular datasets that may include domain specific knowledge more suitable for the tasks at hand. For example, for the task of building context-aware recommender systems in an academic digital library over the LOD cloud, popular datasets like the Semantic Web Dog Food², DBLP³ or Yovisto⁴ can be favored over lesser known but more specific datasets like VIAF⁵ which links authority files of 20 national libraries, list of subject headings for public libraries in Spain⁶ or the French dissertation search engine⁷.

Dataset discovery can be done through public data portals like Datahub⁸ and Europe's Public Data⁹ or private ones like Quandl¹⁰ and Engima¹¹. Private portals

¹<http://datahub.io/dataset?tags=lod>

²<http://datahub.io/dataset/semantic-web-dog-food>

³<http://datahub.io/dataset/dblp>

⁴<http://datahub.io/dataset/yovisto>

⁵<http://datahub.io/dataset/viaf>

⁶<http://datahub.io/dataset/lista-encabezamientos-materia>

⁷<http://datahub.io/dataset/thesesfr>

⁸<http://datahub.io>

⁹<http://publicdata.eu>

¹⁰<https://quandl.com/>

¹¹<http://enigma.io/>

harness manually curated data from various sources and expose them to users either freely or through paid plans. The data available is of higher quality but lesser quantity compared to what is available in public portals. Similarly, in some public data portals, administrators manually review datasets information, validate, correct and attach suitable metadata information. This information is mainly in the form of predefined tags such as *media*, *geography*, *life sciences* for organization and clustering purposes. However, the diversity of those datasets makes it harder to classify them in a fixed number of predefined tags that can be subjectively assigned without capturing the essence and breadth of the dataset [66]. Furthermore, the increasing number of datasets available makes the metadata review and curation process unsustainable even when outsourced to communities.

Data profiling is the process of creating descriptive information and collect statistics about that data. It is a cardinal activity when facing an unfamiliar dataset [70]. Data profiling reflects the importance of datasets without the need for detailed inspection of the raw data. It also helps in assessing the importance of the dataset, improving users' ability to search and reuse part of the dataset and in detecting irregularities to improve its quality. Data profiling includes typically several tasks:

- **Metadata profiling:** Provides general information on the dataset (dataset description, release and update dates), legal information (license information, openness), practical information (access points, data dumps), etc.
- **Statistical profiling:** Provides statistical information about data types and patterns in the dataset (e.g. properties distribution, number of entities and RDF triples).
- **Topical profiling:** Provides descriptive knowledge on the dataset content and structure. This can be in form of tags and categories used to facilitate search and reuse.

In this work, we address the challenges of automatic validation and generation of descriptive datasets profiles. This paper proposes Roomba, an extensible framework consisting of a processing pipeline that combines techniques for data portals identification, datasets crawling and a set of pluggable modules combining several profiling tasks. The framework validates the provided dataset metadata against an aggregated standard set of information. Metadata fields are automatically corrected when possible (e.g. adding a missing license URL reference). Moreover, a report describing all the issues highlighting those that cannot be automatically fixed is created to be sent by email to the dataset's maintainer. There exist various statistical and topical profiling tools for both relational and Linked Data. The architecture of the framework allows to easily add them as additional profiling tasks. However, in this paper, we focus on the task of dataset metadata profiling. We validate our framework against a manually created set of profiles and manually check its accuracy by examining the results of running it on various CKAN-based data portals.

The remainder of the paper is structured as follows. In Section 4.2, we present the motivation behind our framework. In Section 4.3, we review relevant related work. In Section 4.4, we describe our proposed framework’s architecture and components that validate and generate dataset profiles. In Section 4.6, we evaluate the framework and we finally conclude and outline some future work in Section 5.6.

4.2 Motivation

Metadata provisioning is one of the Linked Data publishing best practices mentioned in [20]. Datasets should contain the metadata needed to effectively understand and use them. This information includes the dataset’s license, provenance, context, structure and accessibility. The ability to automatically check this metadata helps in:

- **Delaying data entropy:** *Information entropy* refers to the degradation or loss limiting the information content in raw or metadata. As a consequence of information entropy, data complexity and dynamicity, the life span of data can be very short. Even when the raw data is properly maintained, it is often rendered useless when the attached metadata is missing, incomplete or unavailable. Comprehensive high quality metadata can counteract these factors and increase dataset longevity [65].
- **Enhancing data discovery, exploration and reuse:** Users who are unfamiliar with a dataset require detailed metadata to interpret and analyze accurately unfamiliar datasets. A study conducted by the European Union commission [99] found that both business and users are facing difficulties in discovering, exploring and reusing public data. due to missing or inconsistent metadata information.
- **Enhancing spam detection:** Portals hosting public open data like Datahub allow anyone to freely publish datasets. Even with security measures like captchas and anti-spam devices, detecting spam is increasingly difficult. In addition to that, the increasing number of datasets hinders the scalability of this process, affecting the correct and efficient spotting of datasets spam.

4.3 Related Work

Data Catalog Vocabulary (DCAT) [38] and the Vocabulary of Interlinked Datasets (VoID) [27] are concerned with metadata about RDF datasets. There exist several tools aiming at exposing dataset metadata using these vocabularies. In [24], the authors generate VoID descriptions limited to a subset of properties that can be automatically deduced from resources within the dataset. However, it still provides data consumers with interesting insights. Flemming’s Data Quality Assessment

Tool¹² provides basic metadata assessment as it computes data quality scores based on manual user input. The user assigns weights to the predefined quality metrics and answer a series of questions regarding the dataset. These include, for example, the use of obsolete classes and properties by defining the number of described entities that are assigned disjoint classes, the usage of stable URIs and whether the publisher provides a mailing list for the dataset. The ODI certificate¹³, on the other hand, provides a description of the published data quality in plain English. It aspires to act as a mark of approval that helps publishers understand how to publish good open data and users how to use it. It gives publishers the ability to provide assurance and support on their data while encouraging further improvements through an ascending scale. ODI comes as an online and free questionnaire for data publishers focusing on certain characteristics about their data. Although these approaches try to perform metadata profiling, they are either incomplete or manual. In our framework, we propose a more automatized and complete approach.

Metadata profiling: The Project Open Data Dashboard¹⁴ tracks and measures how US government web sites implement the Open Data principles to understand the progress and current status of their public data listings. A validator analyzes machine readable files: e.g. JSON files for automated metrics like the resolved URLs, HTTP status and content-type. However, deep schema information about the metadata is missing like description, license information or tags. Similarly on the LOD cloud, the Datahub LOD Validator¹⁵ gives an overview of Linked Data sources cataloged on the Datahub. It offers a step-by-step validator guidance to check a dataset completeness level for inclusion in the LOD cloud. The results are divided into four different compliance levels from basic to reviewed and included in the LOD cloud. Although it is an excellent tool to monitor LOD compliance, it still lacks the ability to give detailed insights about the completeness of the metadata and overview on the state of the entire LOD cloud group and it is very specific to the LOD cloud group rules and regulations.

Statistical profiling: Calculating statistical information on datasets is vital to applications dealing with query optimization and answering, data cleansing, schema induction and data mining [55, 42, 66]. Semantic sitemaps [26] and RDFStats [67] are one of the first to deal with RDF data statistics and summaries. ExpLOD [61] creates statistics on the interlinking between datasets based on `owl:sameAs` links. In [70], the author introduces a tool that induces the actual schema of the data and gather corresponding statistics accordingly. LODStats [10] is a stream-based approach that calculates more general dataset statistics. ProLOD++ [1] is a Web-based tool that allows LOD analysis via automatically computed hierarchical clustering [14]. Aether [71] generates VoID statistical descriptions of RDF datasets. It also provides

¹²<http://linkeddata.informatik.hu-berlin.de/LDSrcAss/datenquelle.php>

¹³<https://certificates.theodi.org/>

¹⁴<http://labs.data.gov/dashboard/>

¹⁵<http://validator.lod-cloud.net/>

a Web interface to view and compare VoID descriptions. LODOP [41] is a MapReduce framework to compute, optimize and benchmark dataset profiles. The main target for this framework is to optimize the runtime costs for Linked Data profiling. In [59] authors calculate certain statistical information for the purpose of observing the dynamic changes in datasets.

Topical Profiling: Topical and categorical information facilitates dataset search and reuse. Topical profiling focuses on content-wise analysis at the instances and ontological levels. GERBIL [98] is a general entity annotation framework that provides machine processable output allowing efficient querying. In addition, there exist several entity annotation tools and frameworks [25] but none of those systems are designed specifically for dataset annotation. In [43], the authors created a semantic portal to manually annotate and publish metadata about both LOD and non-RDF datasets. In [66], the authors automatically assigned Freebase domains to extracted instance labels of some of the LOD Cloud datasets. The goal was to provide automatic domain identification, thus enabling improving datasets clustering and categorization. In [15], the authors extracted dataset topics by exploiting the graph structure and ontological information, thus removing the dependency on textual labels. In [39], the authors generate VoID and VoL descriptions via a processing pipeline that extracts dataset topic models ranked on graphical models of selected DBpedia categories.

Dataset Search: Dataset search can be done without relying on attached metadata (tags and categories). For example, there exist several approaches to create LOD indexes. In [5], the authors used VoID descriptions to optimize query processing by determining relevant query-able datasets. In [47], the authors created an approximate index structure (QTree) and an algorithm for answering conjunctive queries over Linked Data. SchemEX [63] is a stream-based approach leveraging type and property information of RDF instances to create schema-level indexes.

Semantic search engines like Sindice [31], Swoogle [36] and Watson [28] help in entities lookup but they are not designed specifically for dataset search. In [79], the authors utilized the sig.ma index [45] to identify appropriate data sources for interlinking. Dataset search and discovery is currently done via data portals that rely on attached metadata to provide dataset search features as they run a Solr index on the metadata schemas. Having missing or inconsistent information will affect the search results quality.

Although the above mentioned tools are able to provide various types of information about a dataset, there exists no approach that aggregates this information and is extensible to combine additional profiling tasks. To the best of our knowledge, this is the first effort towards extensible automatic validation and generation of descriptive dataset profiles.

4.4 Profiling Data Portals

In this section, we provide an overview of Roomba’s architecture and the processing steps for validating and generating dataset profiles. Figure 5.1 shows the main steps which are the following: (i) data portal identification; (ii) metadata extraction; (iii) instance and resource extraction; (iv) profile validation (v) profile and report generation.

Roomba is built as a Command Line Interface (CLI) application using Node.js. Instructions on installing and running the framework are available on its public Github repository¹⁶. The various steps are explained in detail below.

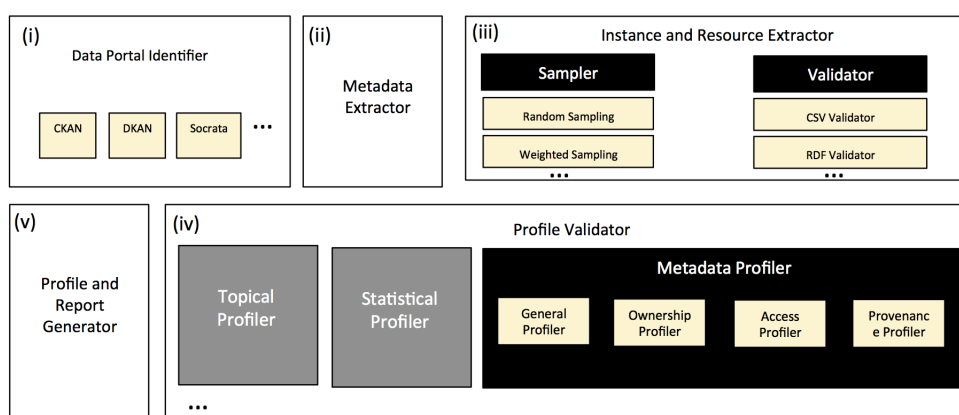


Figure 4.1: Processing pipeline for validating and generating dataset profiles

4.4.1 Data Portal Identification

Data portals can be considered as data access points providing tools to facilitate data publishing, sharing, searching and visualization. CKAN¹⁷ is the world’s leading open-source data portal platform powering web sites like DataHub, Europe’s Public Data and the U.S Government’s open data. Modeled on CKAN, DKAN¹⁸ is a standalone Drupal distribution that is used in various public data portals as well. Socrata¹⁹ helps public sector organizations improve data-driven decision making by providing a set of solutions including an open data portal. In addition to these tradition data portals, there is a set of tools that allow exposing data directly as RESTful APIs like Datatank²⁰ and Database-to-API²¹.

Roomba should be extensible to any data portal. Since every portal has its own API and data model, identifying the software powering data portals is a vital first

¹⁶<https://github.com/ahmadassaf/opendata-checker>

¹⁷<http://ckan.org>

¹⁸<http://nucivic.com/dkan/>

¹⁹<http://www.socrata.com>

²⁰<http://thedataatank.com>

²¹<https://github.com/project-open-data/db-to-api>

step. We rely on several Web scraping techniques in the identification process which includes a combination of the following:

- **URL inspection:** Various CKAN based portals are hosted on subdomains of the <http://ckan.net>. For example, CKAN Brazil (<http://br.ckan.net>). Checking the existence of certain URL patterns can detect such cases.
- **Meta tags inspection:** The `<meta>` tag provides metadata about the HTML document. They are used to specify page description, keywords, author, etc. Inspecting the `content` attribute can indicate the type of the data portal. We use CSS selectors to check the existence of these meta tags. An example of a query selector is `meta[content*='ckan']` (all meta tags with the attribute `content` containing the string *CKAN*). This selector can identify CKAN portals whereas the `meta[content*='Drupal']` can identify DKAN portals.
- **Document Object Model (DOM) inspection:** Similar to the meta tags inspection, we check the existence of certain DOM elements or properties. For example, CKAN powered portals will have DOM elements with class names like `ckan-icon` or `ckan-footer-logo`. A CSS selector like `.ckan-icon` will be able to check if a DOM element with the class name `ckan-icon` exists. The list of elements and properties to inspect is stored in a separate configurable object for each portal. This allows the addition and removal of elements as deemed necessary.

The identification process for each portal can be easily customized by overriding the default function. Moreover, adding or removing steps from the identification process can be easily configured.

After those preliminary checks, we query one of the portal's API endpoints. For example, DataHub is identified as CKAN, so we will query the API endpoint on http://datahub.io/api/action/package_list. A successful request will list the names of the site's datasets, whereas a failing request will signal a possible failure of the identification process.

4.4.2 Metadata Extraction

Data portals expose a set of information about each dataset as metadata. The model used varies across portals. However, a standard model should contain information about the dataset's title, description, maintainer email, update and creation date, etc. We divided the metadata information into the following types:

General information: General information about the dataset. e.g., title, description, ID, etc. This general information is manually filled by the dataset owner. In addition to that, tags and group information is required for classification and enhancing dataset discoverability. This information can be entered manually or inferred modules plugged into the topical profiler.

Access information: Information about accessing and using the dataset. This includes the dataset URL, license information i.e., license title and URL and information about the dataset's resources. Each resource has as well a set of attached metadata e.g., resource name, URL, format, size.

Ownership information: Information about the ownership of the dataset. e.g., organization details, maintainer details, author. The existence of this information is important to identify the authority on which the generated report and the newly corrected profile will be sent to.

Provenance information: Temporal and historical information on the dataset and its resources. For example, creation and update dates, version information, version, etc. Most of this information can be automatically filled and tracked.

Building a standard metadata model is not the scope of this paper, and since we focus on CKAN-based portals, we validate the extracted metadata against the CKAN standard model²².

After identifying the underlying portal software, we perform iterative queries to the API in order to fetch datasets metadata and persist them in a file-based cache system. Depending on the portal software, we can issue specific extraction jobs. For example, in CKAN-based portals, we are able to crawl and extract the metadata of a specific dataset, all the datasets in a specific group (e.g. LOD cloud) or all the datasets in the portal.

4.4.3 Instance and Resource Extraction

From the extracted metadata we are able to identify all the resources associated with that dataset. They can have various types like a SPARQL endpoint, API, file, visualization, etc. However, before extracting the resource instance(s) we perform the following steps:

- **Resource metadata validation and enrichment:** Check the resource attached metadata values. Similar to the dataset metadata, each resource should include information about its mimetype, name, description, format, valid de-referenceable URL, size, type and provenance. The validation process issue an HTTP request to the resource and automatically fills up various missing information when possible, like the mimetype and size by extracting them from the HTTP response header. However, missing fields like name and description that needs manual input are marked as missing and will appear in the generated summary report.
- **Format validation:** Validate specific resource formats against a linter or a validator. For example, node-csv²³ for CSV files and n3²⁴ to validate N3 and

²²http://demo.ckan.org/api/3/action/package_show?id=adur_district_spending

²³<https://github.com/wdavidw/node-csv>

²⁴<https://github.com/RubenVerborgh/N3.js>

Turtle RDF serializations.

Considering that certain datasets contain large amounts of resources and the limited computation power of some machines on which the framework might run on, a sampler module can be introduced to execute various sample-based strategies detailed as they were found to generate accurate results even with comparably small sample size of 10%. These strategies introduced in [39] are:

- **Random Sampling:** Randomly selects resources instances.
- **Weighted Sampling:** Weighs each resources as the ratio of the number of datatype properties used to define a resource over the maximum number of datatype properties over all the datasets resources.
- **Resource Centrality Sampling:** Weighs each resource as the ration of the number of resource types used to describe a particular resource divided by the total number of resource types in the dataset. This is specific and important to RDF datasets where important concepts tend to be more structured and linked to other concepts.

However, the sampler is not restricted only to these strategies. Strategies like those introduced in [69] can be configured and plugged in the processing pipeline.

4.4.4 Profile Validation

A dataset profile should include descriptive information about the data examined. In our framework, we have identified three main categories of profiling information. However, the extensibility of our framework allows for additional profiling techniques to be plugged in easily (i.e. a quality profiling module reflecting the dataset quality). In this paper, we focus on the task of metadata profiling.

Metadata validation process identifies missing information and the ability to automatically correct them. Each set of metadata (general, access, ownership and provenance) is validated and corrected automatically when possible. Each profiler task has a set of metadata fields to check against. The validation process check if each field is defined and if the value assigned is valid.

There exist many special validation steps for various fields. For example, the email addresses and urls should be validated to ensure that the value entered is syntactically correct. In addition to that, for urls, we issue an HTTP HEAD request in order to check if that URL is reachable. We also use the information contained in a valid `content-header` response to extract, compare and correct some resources metadata values like `mimetype` and `size`.

Despite the legal issues surrounding Linked Data licenses [52], it is still considered a gold mine for organizations who are trying to leverage external data sources in order to produce more informed business decisions [17]. In [53], the authors see the

potential economic effect unfolding in education, transportation, consumer products, electricity, oil and gas, health care and consumer finance. They estimate the potential annual value enabled by Open Data in these domains to be 3 trillion US Dollars across seven domains. As a result, validating license related information is vital. However, from our experiments, we found out that datasets' license information is noisy. The license names if found are not standardized. For example, Creative Commons CCZero can be also CC0 or CCZero. Moreover, the license URI if found and if de-referenceable can point to different reference knowledge bases e.g., <http://opendefinition.org>. To overcome this issue, we have manually created a mapping file standardizing the set of possible license names and the reference knowledge base²⁵. In addition, we have also used the open source and knowledge license information²⁶ to normalize the license information and add extra metadata like the domain, maintainer and open data conformance.

```
{
    "license_id" : ["ODC-PDDL-1.0"],
    "disambiguations" : ["Open Data Commons Public Domain
        Dedication and License (PDDL)"]
},
{
    "license_id" : ["CC-BY-SA-4.0", "CC-BY-SA-3.0"],
    "disambiguations" : ["cc-by-sa", "CC BY-SA", "Creative
        Commons Attribution Share-Alike"]
}
```

Listing 4.1: License mapping file sample

4.4.5 Profile and Report Generation

The validation process highlights the missing information and presents them in a human readable report. The report can be automatically sent to the dataset maintainer email if exists in the metadata. In addition to the generated report, the enhanced profiles are represented in JSON using the CKAN data model and are publicly available²⁷.

Data portal administrators need an overall knowledge of the portal datasets and their properties. Our framework has the ability to generate numerous reports of all the datasets by passing formatted queries. There are two main sets of aggregation tasks that can be run:

- **Aggregating meta-field values:** Passing a string that corresponds to a valid field in the metadata. The field can be flat like `license_title` (ag-

²⁵<https://github.com/ahmadassaf/opendata-checker/blob/master/util/licenseMappings.json>

²⁶<https://github.com/okfn/licenses>

²⁷<https://github.com/ahmadassaf/opendata-checker/tree/master/results>

gregates all the license titles used in the portal or in a specific group) or nested like `resource>resource_type` (aggregates all the resources types for all the datasets). Such reports are important to have an overview of the possible values used for each metadata field.

- **Aggregating key:object meta-field values:** Passing two meta-field values separated by a colon : e.g., `resources>resource_type:resources>name`. These reports are important as you can aggregate the information needed when also having the set of values associated to it printed.

For example, the meta-field value query `resource>resource_type` run against the LODCloud group will result in an array containing `[file, api, documentation...]` values. These are all the resource types used to describe all the datasets of the group. However, to be able to know also what are the datasets containing resources corresponding to each type, we issue a key:object meta-field query `resource>resource_type:name`. The result will be a JSON object having the `resource_type` as the key and an array of corresponding datasets titles that has a resource of that type.

Metadata Report
group information is missing. Check organization information as they can be mixed sometimes organization_image_url field exists but there is no value defined
Tag Statistics
There is a total of: 21 [undefined] vocabulary_id fields 100.00%
License Report
License information has been normalized !
Resource Statistics
There is a total of: 10 [missing] url-type fields 100.00%
There is a total of: 9 [missing] created fields 90.00%
There is a total of: 10 [undefined] cache_last_updated fields 100.00%
There is a total of: 10 [undefined] size fields 100.00%
There is a total of: 10 [undefined] hash fields 100.00%
There is a total of: 10 [undefined] mimetype_inner fields 100.00%
There is a total of: 7 [undefined] mimetype fields 70.00%
There is a total of: 10 [undefined] cache_url fields 100.00%
There is a total of: 6 [undefined] name fields 60.00%
There is a total of: 9 [undefined] webstore_url fields 90.00%
There is a total of: 9 [undefined] last_modified fields 90.00%
There is one [undefined] format field 10.00%
Resource Connectivity Issues
There are 2 connectivity issues with the following URLs: – \url{http://dbpedia.org/void/Dataset}
Un-Reachable URLs Types

```
There are: 1 unreachable URLs of type [ file ]
```

Listing 4.2: Excerpt of the DBpedia validation report

4.5 Experiments and Evaluation

In this section, we provide the experiments and evaluation of the proposed framework. All the experiments are reproducible by our tool and their results are available in its Github repository. A CKAN dataset metadata describes four main sections in addition to the core dataset's properties. These sections are:

- **Resources:** The distributable parts containing the actual raw data. They can come in various formats (JSON, XML, RDF, etc.) and can be downloaded or accessed directly (REST API, SPARQL endpoint).
- **Tags:** Provide descriptive knowledge on the dataset content and structure. They are used mainly to facilitate search and reuse.
- **Groups:** A dataset can belong to one or more group that share common semantics. A group can be seen as a cluster or a curation of datasets based on shared categories or themes.
- **Organizations:** A dataset can belong to one or more organization controlled by a set of users. Organizations are different from groups as they are not constructed by shared semantics or properties, but solely on their association to a specific administration party.

Each of these sections contains a set of metadata corresponding to one or more type (general, access, ownership and provenance). For example, a dataset resource will have general information such as the resource name, access information such as the resource url and provenance information such as creation date. The framework generates a report aggregating all the problems in all these sections, fixing field values when possible. Errors can be the result of missing metadata fields, undefined field values or field value errors (e.g. unreachable URL or incorrect email addresses).

4.5.1 Experimental Setup

We ran our tool on two CAKN-based data portals. The first one is datahub.io targeting specifically the LOD cloud group. The current state of the LOD cloud report [75] indicates that the LOD cloud contains 1014 datasets. They were harvested via a LDSpider crawler [51] seeded with 560 thousands URIs. Roomba, on the other hand, fetches datasets hosted in data portals where datasets have attached relevant metadata. As a result, we relied on the information provided by the Datahub CKAN

API. Examining the tags available, we found two candidate groups. The first one tagged with “lodcloud” returned 259 datasets, while the second one tagged with “lod” returned only 75 datasets. After manually examining the two lists, we found out the datasets grouped with the tag “lodcloud” are the correct ones. To qualify other CKAN-based portals for the experiments, we use <http://dataportals.org/> which contains a comprehensive list of Open Data portals from around the world. In the end, we chose the Amsterdam data portal²⁸. The portal was commissioned in 2012 by the Amsterdam Economic Board Open Data Exchange (ODE) and covers a wide range of information domains (energy, economy, education, urban development, etc.) about Amsterdam metropolitan region.

We ran our tool on two CAKN-based data portals. The first is the Datahub targeting specifically the LOD cloud group. The current state of the LOD cloud report [75] indicates that the LOD cloud contains 1014 datasets. They were harvested via an LDSpider crawler [51] seeded with 560 thousands URIs. Roomba on the other hand, fetches datasets hosted in data portals where datasets have attached relevant metadata. As a result, we relied on the information provided by the Datahub CKAN API. Examining the tags available, we found two candidate groups. The first tagged with “lodcloud” returned 259 datasets, while the second tagged with “lod” returned only 75 datasets. After manually examining the two lists, we found out the datasets grouped with the tag “lodcloud” are the correct ones. To qualify other CKAN-based portals for the experiments, we used dataportals.org, which contains a comprehensive list of Open Data portals from around the world. In the end, we chose the Amsterdam data portal²⁹. The portal was commissioned in 2012 by the Amsterdam Economic Board Open Data Exchange (ODE), and covers a wide range of information domains (energy, economy, education, urban development, etc.) about Amsterdam metropolitan region.

We ran the Roomba instance and resource extractors in order to cache the meta-data files for these datasets locally and ran the validation process. The experiments were executed on a 2.6 Ghz Intel Core i7 processor with 16GB of DDR3 memory machine. The approximate execution time alongside the summary of the datasets’ properties are presented in table 4.1.

Data Portal	No. Datasets	No. Groups	No. Resources	Processing Time
LOD Cloud	259	N/A	1068	140 mins
Amsterdam Open Data	172	18	480	35 mins

Table 4.1: Summary of the experiments details

In our evaluation, we focused on two aspects: i) *profiling correctness* which manually assesses the validity of the errors generated in the report, and ii) *profiling com-*

²⁸<http://data.amsterdamopendata.nl/>

²⁹<http://data.amsterdamopendata.nl/>

pleteness which assesses if the profilers cover all the errors in the datasets metadata.

4.5.2 Profiling Correctness

To measure profile correctness, we need to make sure that the issues reported by Roomba are valid on the dataset, group and portal levels.

On the dataset level, we choose three datasets from both the LOD Cloud and the Amsterdam data portal. The datasets details are shown in table 4.2.

Dataset Name	Data Portal	Group ID	Resources	Tags
dbpedia	Datahub	lodcloud	10	21
event-media	Datahub	lodcloud	9	15
bbc-music	Datahub	lodcloud	2	14
bevolking.cijfers.amsterdam	Amsterdam	bevolking	6	12
bevolking-prognoses-amsterdam	Amsterdam	bevolking	1	3
religieuze.samenkomstlocaties	Amsterdam	bevolking	1	8

Table 4.2: Datasets chosen for the correctness evaluation

To measure the profiling correctness on the groups level, we selected four groups from the Amsterdam data portal containing a total of 25 datasets. The choice was made to cover groups in various domains that contain a moderate number of datasets that can be checked manually (between 3-9 datasets). Table 4.3 summarizes the groups chosen for the evaluation.

Group Name	Domain	Datasets	Resources	Tags
bestuur-en-organisatie	Management	9	45	101
bevolking	Population	3	8	23
geografie	Geography	8	16	56
openbare-orde-veiligheid	Public Order & Safety	5	19	34

Table 4.3: Groups chosen for the correctness evaluation

After running Roomba and examining the results on the selected datasets and groups, we found out that our framework provides 100% correct results on the individual dataset level and on the aggregation level over groups. Since our portal level aggregation is extended from the group aggregation, we can infer that the portal level aggregation also produces complete correct profiles. However, the lack of a standard way to create and manage collections of datasets was the source of some errors when comparing the results from these two portals. For example, in Datahub, we noticed that all the datasets **groups** information were missing, while in the Amsterdam Open Data portal, all the **organisation** information was missing. Although the error detection is correct, the overlap in the usage of group and organization can give a false indication about the metadata quality.

4.5.3 Profiling Completeness

We analyzed the completeness of our framework by manually constructing a set of profiles that act as a golden standard. These profiles cover the range of uncommon problems that can occur in a certain dataset³⁰. These errors are:

- Incorrect `mimetype` or `size` for resources;
- Invalid number of tags or resources defined;
- Check if the license information can be normalized via the `license_id` or the `license_title` as well as the normalization result;
- Syntactically invalid `author_email` or `maintainer_email`.

After running our framework at each of these profiles, we measured the completeness and correctness of the results. We found out that our framework covers indeed all the metadata problems that can be found in a CKAN standard model correctly.

4.6 Experiments and Evaluation

In this section, we describe our experiments when running the Roomba tool on the LOD cloud. All the experiments are reproducible by our tool and their results are available on its Github repository at <https://github.com/ahmadassaf/opendata-checker>.

4.6.1 Experimental Setup

The current state of the LOD cloud report [75] indicates that there are more than 1014 datasets available. These datasets have been harvested by the LDSpider crawler [51] seeded with 560 thousands URIs. However, since Roomba requires the datasets metadata to be hosted in a data portal where either the dataset publisher or the portal administrator can attach relevant metadata to it, we rely on the information provided by the Datahub CKAN API. We consider two possible groups: the first one tagged with “lodcloud” returns 259 datasets, while the second one tagged with “lod” returns only 75 datasets. We manually inspect these two lists and find out that the API result for the tag “lodcloud” is the correct one. The 259 datasets contain a total of 1068 resources. We run the instance and resource extractor from Roomba in order to cache the metadata files for these datasets locally and we launch the validation process which takes around two and a half hours on a 2.6 Ghz Intel Core i7 processor with 16GB of DDR3 memory machine.

³⁰<https://github.com/ahmadassaf/opendata-checker/tree/master/test>

Metadata Field		Error %	Section	Error Type	Auto Fix
General	group	100%	Dataset	Missing	-
	vocabulary_id	100%	Tag	Undefined	-
	url-type	96.82%	Resource	Missing	-
	mimetype_inner	95.88%	Resource	Undefined	Yes
	hash	95.51%	Resource	Undefined	Yes
	size	81.55%	Resource	Undefined	Yes
Access	cache_url	96.9%	Resource	Undefined	-
	webstore_url	91.29%	Resource	Undefined	-
	license_url	54.44%	Dataset	Missing	Yes
	url	30.89%	Resource	Unreachable	-
	license_title	16.6%	Dataset	Undefined	Yes
Provenance	cache_last_updated	96.91%	Resource	Undefined	Yes
	webstore_last_updated	95.88%	Resource	Undefined	Yes
	created	86.8%	Resource	Missing	Yes
	last_modified	79.87%	Resource	Undefined	Yes
	version	60.23%	Dataset	Undefined	-
Ownership	maintainer_email	55.21%	Dataset	Undefined	-
	maintainer	51.35%	Dataset	Undefined	-
	author_email	15.06%	Dataset	Undefined	-
	organization_image_url	10.81%	Dataset	Undefined	-
	author	2.32%	Dataset	Undefined	-

Table 4.4: Top metadata fields error % by type

4.6.2 Results and Evaluation

CKAN dataset metadata includes three main sections in addition to the core dataset's properties. Those are the **groups**, **tags** and **resources**. Each section contains a set of metadata corresponding to one or more metadata type. For example, a dataset resource will have general information such as the resource name, access information such as the resource url and provenance information such as creation date. The framework generates a report aggregating all the problems in all these sections, fixing field values when possible. Errors can be the result of missing metadata fields, undefined field values or field value errors (e.g. unreachable URL or incorrect email address).

Figures 4.2 and 4.3 show the percentage of errors found in metadata fields by section and by information type respectively. We observe that the most erroneous information for the dataset core information is related to ownership since this information is missing or undefined for 41% of the datasets. Datasets resources have the poorest metadata. 64% of the general metadata, all the access information and 80% of the provenance information contain missing or undefined values. Table 4.4 shows the top metadata fields errors for each metadata information type.

We notice that 42.85% of the top metadata problems can be fixed automatically. Among them, 44.44% of these problems can be fixed by our tool while the others need tools that are plugged into the data portal. We further present and discuss the results grouped by metadata information type in the following sub-sections.

4.6.3 General information

34 datasets (13.13%) do not have valid `notes` values. `tags` information for the datasets are complete except for the `vocabulary_id` as this is missing from all the datasets' metadata. All the datasets `groups` information are missing `display_name`, `description`, `title`, `image_display_url`, `id`, `name`. After manual examination, we observe a clear overlap between group and organization information. Many datasets like `event-media` use the organization field to show group related information (being in the LOD Cloud) instead of the publishers details.

4.6.4 Access information

25% of the datasets access information (being the dataset URL and any URL defined in its groups) have issues: generally missing or unreachable URLs. 3 datasets (1.15%) do not have a URL defined (tip, uniprot databases, uniprot citations) while 45 datasets (17.3%) defined URLs are not accessible at the time of writing this paper. One dataset does not have resources information (bio2rdf chebi) while the other datasets have a total of 1068 defined resources.

On the datasets resources level, we notice wrong or inconsistent values in the `size` and `mimetype` fields. However, 44 datasets have valid `size` field values and 54 have valid `mimetype` field values but they were not reachable, thus providing incorrect information. 15 fields (68%) of all the other access metadata are missing or have undefined values. Looking closely, we notice that most of these problems can be easily fixed automatically by tools that can be plugged to the data portal. For example, the top six missing fields are the `cache_last_updated`, `cache_url`, `urltype`, `webstore_last_updated`, `mimetype_inner` and `hash` which can be computed and filled automatically. However, the most important missing information which require manual entry are the dataset's `name` and `description` which are missing from 817 (76.49%) and 98 (9.17%) resources respectively. A total of 334 resources (31.27%) URLs were not reachable, thus affecting highly the availability of these datasets. CKAN resources can be of various predefined types (*file*, *file.upload*, *api*, *visualization*, *code and documentation*). Roomba also breaks down these unreachable resources according to their types: 211 (63.17%) resources do not have valid `resource_type`, 112 (33.53%) are files, 8 (2.39%) are re metadata and one (0.029%) are example and documentation types.

To have more details about the resources URL types, we created a *key : objectmeta-fieldvalues* group level report on the LOD cloud with `resources>format:title`. This will aggregate the resources format information for each dataset.

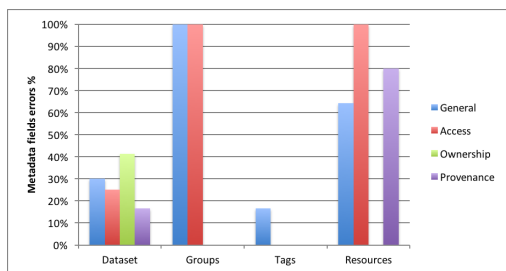


Figure 4.2: Error % by section

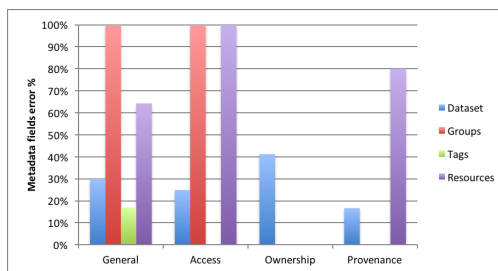


Figure 4.3: Error % by information type

We observe that only 161 (62.16%) of the datasets valid URLs have SPARQL endpoints defined using the `api/sparql` resource format. 92.27% provided RDF example links and 56.3% provided direct links to RDF down-loadable dumps.

The noisiest part of the access metadata is about license information. A total of 43 datasets (16.6%) does not have a defined `license_title` and `license_id` fields, where 141 (54.44%) have missing `license_url` field.

4.6.5 Ownership information

Ownership information is divided into direct ownership (author and maintainer) and organization information. Four fields (66.66%) of the direct ownership information are missing or undefined. The breakdown for the missing information is: 55.21% `maintainer_email`, 51.35% `maintainer`, 15.06% `author_email`, 2.32% `author`. Moreover, our framework performs checks to validate existing email values. 11 (0.05%) and 6 (0.05%) of the defined `author_email` and `maintainer_email` fields are not valid email addresses respectively. For the organization information, two field values (16.6%) were missing or undefined. 1.16% of the `organization_description` and 10.81% of the `organization_image_url` information with two out of these URLs are unreachable.

4.6.6 Provenance information

80% of the resources provenance information are missing or undefined. However, most of the provenance information (e.g. `metadata_created`, `metadata_modified`) can be computed automatically by tools plugged into the data portal. The only field requiring manual entry is the `version` field which was found to be missing in 60.23% of the datasets.

4.6.7 Enriched Profiles

Roomba can automatically fix, when possible, the license information (title, url and id) as well as the resources mimetype and size.

20 resources (1.87%) have incorrect `mimetype` defined, while 52 resources (4.82%) have incorrect `size` values. These values have been automatically fixed based on the values defined in the HTTP response header.

We have noticed that most of the issues surrounding license information are related to ambiguous entries. To resolve that, we manually created a mapping file³¹ standardizing the set of possible license names and urls using the open source and knowledge license information³². As a result, we managed to normalize 123 (47.49%) of the datasets' license information.

To check the impact of the corrected fields, we seeded Roomba with the enriched profiles. Since Roomba uses file based cache system, we simply replaced all the datasets `json` files in the `\cache\datahub.io\datasets` folder with those generated in `\cache\datahub.io\enriched`. After running Roomba again on the enriched profiles, we observe that the errors percentage for missing `size` fields decreased by 32.02% and for `mimetype` fields by 50.93%. We also notice that the error percentage for missing `license_urls` decreased by 2.32%.

4.7 Conclusion and Future Work

In this paper, we proposed a scalable automatic approach for extracting, validating, correcting and generating descriptive linked dataset profiles. This approach applies several techniques in order to check the validity of the metadata provided and to generate descriptive and statistical information for a particular dataset or for an entire data portal. Based on our experiments running the tool on the LOD cloud, we discovered that the general state of the datasets needs attention as most of them lack informative access information and their resources suffer low availability. These two metrics are of high importance for enterprises looking to integrate and use external linked data.

It has been noticed that the issues surrounding metadata quality affect directly dataset search as data portals rely on such information to power their search index. We noted the need for tools that are able to identify various issues in this metadata and correct them automatically. We evaluated our framework manually against two prominent data portals and proved that we can automatically scale the validation of datasets metadata profiles completely and correctly.

As part of our future work, we plan to introduce workflows that will be able to correct the rest of the metadata either automatically or through intuitive manually-driven interfaces. We also plan to integrate statistical and topical profilers to be able to generate full comprehensive profiles. We also intend to suggest a ranked standard metadata model that will help generate more accurate and scored metadata quality

³¹<https://github.com/ahmadassaf/opendata-checker/blob/master/util/licenseMappings.json>

³²<https://github.com/okfn/licenses>

profiles. We also plan to run this tool on various CKAN-based data portals, schedule periodic reports to monitor the evolution of datasets metadata. Finally, at some stage, we plan to extend this tool for other data portal types like DKAN and Socrata.

Data Aggregation and Modeling

5.1 Introduction

In the last few years the Semantic Web gained a momentum supported by the introduction of many related initiatives like the Linked Open Data (LOD)¹. From 12 datasets cataloged in 2007, the Linked Open Data cloud has grown to nearly 1000 datasets containing more than 82 billion triples. Data is being published by both public and private sectors and covers a diverse set of domains from life sciences to military. This success lies in the cooperation between data publishers and consumers where users are empowered to find, share and combine information in their applications easily.

We are entering an era where open is the new default. Governments, universities, organizations and even individuals are publicly publishing huge amounts of open data. This openness should be accompanied with a certain level of trust or guarantees about the quality of data. The Linked Open Data is a gold mine for those trying to leverage external data sources in order to produce more informed business decisions [17]. However, the heterogeneous nature of sources reflects directly on the data quality as these sources often contain inconsistent as well as misinterpreted and incomplete information.

Traditional data quality is a thoroughly researched field with several benchmarks and frameworks to grasp its dimensions [58, 12, 100]. Data quality principles typically rely on many subjective indicators that are complex to measure automatically. The quality of data is indeed realized when it is used [57], thus directly relating to the ability of satisfying users' continuous needs.

Web documents that are by nature unstructured and interlinked require different quality metrics and assessment techniques than traditional datasets. For example, the importance and quality of Web documents can be subjectively calculated via algorithms like Page Rank [68]. Ensuring data quality in Linked Open Data is much more complex. It consists of structured information supported by models, ontologies and vocabularies and contains queryable endpoints and links. This makes data quality assurance a challenge. Despite the fact that Linked Open Data quality is a trending and highly demanded topic, very few efforts are currently trying to standardize, track and formalize frameworks to issue scores or certificates that will help data consumers in their integration tasks.

¹<http://lod-cloud.net>

Data quality assessment is the process of evaluating if a piece of data meets the consumers need in a specific use case [13]. The dimensionality of data quality makes it dependent on the task and users requirements. For example, DBpedia [21] is a knowledge base containing data extracted from structured and semi-structured sources. It is used in a variety of applications e.g. annotation systems [76], exploratory search [72] and recommendation engines [32]. However, DBpedia's data is not integrated into critical systems e.g. life critical (medical applications) or safety critical (aviation applications) as its data quality is found to be insufficient. In this paper, we first propose a comprehensive objective framework to evaluate the quality of Linked Data sources. Secondly, we present an extensible quality measurement tool that helps on one hand data owners to rate the quality of their dataset and get some hints on possible improvements, and on the other hand data consumers to choose their data sources from a ranked set. The aim of this paper is to provide researchers and practitioners with a comprehensive understanding of the objective issues surrounding Linked Data quality.

The framework we propose is based on a refinement of the data quality principles described in [9] and surveyed in [101]. Some attributes have been grouped for more detailed quality assessments while we have also extended them by adding for each attribute a set of objective indicators. These indicators are measures that provide users with quality metrics measurable by tools regardless of the use case. For example, when measuring the quality of DBpedia dataset, an objective metric would be the availability of human or machine readable license information rather than the trustworthiness of the publishers.

Furthermore, we surveyed the landscape of Linked Data quality tools to discover that they only cover a subset of the proposed objective quality indicators. As a result, we extend Roomba which is a framework to assess and build dataset profiles with an extensible quality measurement tool and evaluate it by measuring the quality of the LOD cloud group. The results demonstrate that the general quality of LOD cloud needs more attention as most of the datasets suffer from various quality issues.

This paper is structured as follows: Section 5.2 presents the various data quality assessment methodologies. Section 5.3 presents our framework with its objective quality measures and indicators. Section 5.4 presents our tool for evaluating those indicators. Section 5.5 reviews the existing tools and frameworks in the Linked Open Data quality landscape. Section 5.6 presents concluding remarks and identifies future work.

5.2 Data Quality Assessment

In [101], the authors present a comprehensive systematic review of data quality assessment methodologies applied to LOD. They have extracted 26 quality dimensions and a total of 110 objective and subjective quality indicators. However, some of those

objective indicators are dependent on the use case thus there is no clear separation on what can be automatically measured. For example, data completeness is generally a subjective dimension. However, the authors specified that the detection of the degree on which all the real-world objects are represented, detection of number of missing values for specific property and detection of the degree to which instances in the dataset are interlinked are considered as objective indicators given the presence of a gold standard or the original data source to compare with. Moreover, lots of the defined performance dimensions like low latency, high throughput or scalability of a data source were defined as objective but are still dependent on multiple subjective factors like network congestion. In addition, there were some missing objective indicators vital to the quality of LOD e.g. indication of the openness of the dataset.

The ODI certificate² provides a description of the published data quality in plain English. It aspires to act as a mark of approval that helps publishers understand how to publish good open data and users how to use it. It gives publishers the ability to provide assurance and support on their data while encouraging further improvements through an ascending scale.

ODI comes as an online and free questionnaire for data publishers focusing on certain characteristics about their data. The questions are classified into the following categories: general information (about dataset, publisher and type of release), legal information (e.g. rights to publish), licensing, privacy (e.g. whether individuals can be identified), practical information (e.g. how to reach the data), quality, reliability, technical information (e.g. format and type of data) and social information (e.g. contacts, communities, etc.). Based on the information provided by the data publisher, a certificate is created with one of four different ratings.

Although ODI is a great initiative, the issued certificates are self-certified. ODI does not verify or review submissions but retains the right to revoke a certificate at any time. The dynamicity of Linked Data makes it also very difficult to update the certificates manually, especially when these changes are frequent and affect multiple categories. There is clearly a need for automatic certification which can be supplemented with some manual input for categories that cannot be processed by machines.

The emerging critical need for large, distributed, heterogeneous, and complex structured datasets identified the necessity to establish industry cooperation between vendors of RDF and Graph database technologies in developing, endorsing, and publishing reliable and insightful benchmark results. The Linked Data Benchmark Council (LDBC)³ aims to bridge the gap between the industry and the new trending stack of semantic technologies and their vendors. LDBC aims more specifically at developing new benchmarks that will lead to significant progress in scalability, storage, indexing and query optimization techniques to become the de facto standard

²<https://certificates.theodi.org/>

³<http://ldbc.eu/>

for publishing performance results. LDBC is a promising initiative, but it is still work in progress with the final report expected on the first quarter of 2015.

In [90], the authors propose a methodology for assessing Linked Data quality. It consists of three main steps: (1) requirement analysis, (2) quality assessment and (3) quality improvement. Considering the multidimensionality of data quality, the methodology requires users to provide the details of a use case or a scenario that describes the intended usage of the data. Moreover, quality issues identification is done with the help of a checklist. The user must have prior knowledge about the details of the data in order to fill this list. Tools implementing the proposed methodology should be able to generate comprehensive quality measures. However, they will require heavy manual intervention and deep knowledge on the data examined. These issues highly affect detecting quality issue on large scale.

Despite all the recent efforts in providing frameworks and tools for data quality in Linked Open Data, there is still no automatic framework for the objective assessment of Linked Data quality.

5.3 Objective Linked Data Quality Classification

The basic idea behind Linked Data is that its usefulness increases when it is more interlinked with other datasets. Tim Berners-Lee defined four main principles for publishing data that can ensure a certain level of uniformity reflecting directly data's usability [95]:

- **Make the data available on the Web:** assign URIs to identify things.
- **Make the data machine readable:** use HTTP URIs so that looking up these names is easy.
- **Use publishing standards:** when the lookup is done provide useful information using standards like RDF.
- **Link your data:** include links to other resources to enable users to discover more things.

Building on these principles, we group the quality attributes into four main categories:

- **Quality of the entities :** quality indicators that focus on the data at the instance level.
- **Quality of the dataset:** quality indicators at the dataset level.
- **Quality of the semantic model:** quality indicators that focus on the semantic models, vocabularies and ontologies.

- **Quality of the linking process:** quality indicators that focus on the inbound and outbound links between datasets.

In [9], the authors identified 24 different Linked Data quality attributes. In this paper, we refine these attributes into a condensed framework of 10 objective measures. Since these measures are rather abstract, we should rely on quality indicators that reflect data quality [7]. In this paper, we transform the quality indicators presented as a set of questions in [9] into more concrete quality indicator metrics. Independent indicators for entity quality are mainly subjective e.g. the degree to which all the real-world objects are represented, the scope and level of details, etc. However, since entities are governed by the underlying model, we have grouped their indicators with those of the modeling quality. Table 1 lists the refined measures alongside their quality indicators. These attributes are presented in the following sections.

Table 5.1: Objective Linked Data Quality Framework

Quality Attribute	Quality Category	ID	Quality Indicator
Completeness	Dataset Level	1	Existence of supporting structured metadata [49]
		2	Supports multiple serializations [101]
		3	Has different data access points
		4	Uses datasets description vocabularies
		5	Existence of descriptions about its size
		6	Existence of descriptions about its structure (MIME Type, Format)
		7	Existence of descriptions about its organization and categorization
	Links Level	8	Existence of information about the kind and number of used vocabu
	Model Level	9	Existence of dereferencable links for the dataset [49, 23, 46]
		10	Absence of disconnected graph clusters [23]
		11	Absence of omitted top concept [49]
		12	Has complete language coverage [23]
		13	Absence of unidirectional related concepts [49]
		14	Absence of missing labels [23]
		15	Absence of missing equivalent properties [60]
Availability	Dataset Level	16	Absence of missing inverse relationships [60]
		17	Absence of missing domain or range values in properties [60]
		18	Existence of an RDF dump that can be downloaded by users [7][49]
Licensing	Dataset Level	19	Existence of a queryable endpoint that responds to direct queries
		20	Existence of valid dereferencable URLs (respond to HTTP request)
		21	Existence of human and machine readable license information [50]
Freshness	Dataset Level	22	Existence of de-referenceable links to the full license information [50]
		23	Specifies permissions, copyrights and attributions [101]
		24	Existence of timestamps that can keep track of its modifications [40]
Correctness	Dataset Level	25	Includes the correct MIME-type for the content [49]
		26	Includes the correct size for the content
		27	Absence of syntactic errors on the instance level [49]
	Links Level	28	Absence of syntactic errors [81]

Continued on

Table 5.1 Objective Linked Data Quality Framework

Quality Attribute	Quality Category	ID	Quality Indicator
	Model Level	29	Use the HTTP URI scheme (avoid using URNs or DOIs) [23]
		30	Contains marked top concepts [23]
		31	Absence of broader concepts for top concepts [23]
		32	Absence of missing or empty labels [2, 23]
		33	Absence of unprintable characters [2, 23] or extra white spaces in labels
		34	Absence of incorrect data type for typed literals [49, 2]
		35	Absence of omitted or invalid languages tags [82, 23]
		36	Absence of terms without any associative or hierarchical relationships
Comprehensibility	Dataset Level	37	Existence of at least one exemplary RDF file [101]
		38	Existence of at least one exemplary SPARQL query [101]
		39	Existence of general information (title, URL, description) for the dataset
		40	Existence of a mailing list, message board or point of contact [7]
	Model Level	41	Absence of misuse of ontology annotations [23, 60]
		42	Existence of annotations for concepts [60]
Provenance	Dataset Level	43	Existence of documentation for concepts [23, 60]
		44	Existence of metadata that describes its authoritative information [40]
		45	Usage of a provenance vocabulary
Coherence	Model Level	46	Usage of a versioning
		47	Absence of misplaced or deprecated classes or properties [49]
		48	Absence of relation and mappings clashes [82]
		49	Absence of blank nodes [50]
		50	Absence of invalid inverse-functional values [49]
		51	Absence of cyclic hierarchical relations [91, 82, 23]
		52	Absence of undefined classes and properties usage [49]
		53	Absence of solely transitive related concepts [23]
Consistency	Model Level	54	Absence of redefinitions of existing vocabularies [49]
		55	Absence of valueless associative relations [23]
		56	Consistent usage of preferred labels per language tag [8, 23]
		57	Consistent usage of naming criteria for concepts [60]
		58	Absence of overlapping labels
		59	Absence of disjoint labels [23]
		60	Absence of atypical use of collections, containers and reification [49]
Security	Dataset Level	61	Absence of wrong equivalent, symmetric or transitive relationships [60]
		62	Absence of membership violations for disjoint classes [49]
		63	Uses login credentials to restrict access [101]
		64	Uses SSL or SSH to provide access to their dataset [101]

5.3.1 Completeness

Data completeness can be judged in the presence of a task where the ideal set of attributes and objects are known. It is generally a subjective measure depending highly on the scenario and use-case in hand. For example, an entity is considered to be complete if it contains all the attributes needed for a given task, has complete language coverage [23] and has documentation properties [6, 23]. Dataset completeness has some objective measures which we include in our framework. A dataset is

considered to be complete if it:

- Contains supporting structured metadata [49].
- Provides data in multiple serializations (N3, Turtle, etc.) [101].
- Contains different data access points. These can either be a queryable endpoint (i.e. SPARQL endpoint, REST API, etc.) or a data dump file.
- Uses datasets description vocabularies like DCAT⁴ or VOID⁵.
- Provides descriptions about its size e.g. `void:statItem`, `void:numberOfTriples` or `void:numberOfDocuments`.
- Existence of descriptions about its format.
- Contains information about its organization and categorization e.g. `dcterms:subject`.
- Contains information about the kind and number of used vocabularies [101].

Links are considered to be complete if the dataset and all its resources have defined links [49, 23, 46]. Models are considered to be complete if they do not contain disconnected graph clusters [23]. Disconnected graphs are the result of incomplete data acquisition or accidental deletion of terms that leads to deprecated terms. In addition to that, models are considered to be complete if they have complete language coverage (each concept labeled in each of the languages that are also used on the other concepts) [23], do not contain omitted top concepts or unidirectional related concepts [49] and if they are not missing labels [23], equivalent properties, inverse relationships, domain or range values in properties [60].

5.3.2 Availability

A dataset is considered to be available if the publisher provides data dumps e.g. RDF dump, that can be downloaded by users [7, 49], its queryable endpoints e.g. SPARQL endpoint, are reachable and respond to direct queries and if all of its inbound and outbound links are dereferencable.

5.3.3 Correctness

A dataset is considered to be correct if it includes the correct MIME-type and size for the content [49] and doesn't contain syntactic errors [49]. Links are considered to be correct if they lack syntactic errors and use the HTTP URI scheme (avoid using URNs or DOIs) [23]. Models are considered to be correct if the top concepts are marked and do not have broader concepts (for example having incoming `hasTopConcept` or

⁴<http://www.w3.org/TR/vocab-dcat/>

⁵<http://www.w3.org/TR/void/>

outgoing `topConceptOf` relationships) [23]. Moreover, if they don't contain incorrect data type for typed literals [49][2], no omitted or invalid languages tags [82, 23], does not contain "orphan terms" (orphan terms are terms without any associative or hierarchical relationships and if the labels are not empty, do not contain unprintable characters [2, 23] or extra white spaces [82].

5.3.4 Consistency

Consistency implies lack of contradictions and conflicts. The objective indicators are mainly associated with the modeling quality. A model is considered to be consistent if it does not contain overlapping labels (two concepts having the same preferred lexical label in a given language when they belong to the same schema) [8, 23], consistent preferred labels per language tag [23, 82], atypical use of collections, containers and reification [49], wrong equivalent, symmetric or transitive relationships [60], consistent naming criteria in the model [23, 60], overlapping labels in a given language for concepts in the same scheme [23] and membership violations for disjoint classes [49, 60].

5.3.5 Freshness

Freshness is a measure for the recency of data. The basic assumption is that old information is more likely to be outdated and unreliable [40]. Dataset freshness can be identified if the dataset contains timestamps that can keep track of its modifications. Data freshness could be considered as a subjective measure. However, our concern is the existence of temporal information allowing dataset consumers to subjectively decide its freshness for their scenario.

5.3.6 Provenance

Provenance can be achieved at the dataset level by including metadata that describes its authoritative information (author, maintainer, creation date, etc.), versioning information and verifying if the dataset uses a provenance vocabulary like PROV [96].

5.3.7 Licensing

Licensing is a quality attribute that is measured on the dataset level. It includes the availability of machine readable license information [50], human readable license information in the documentation of the dataset or its source [50] and the indication of permissions, copyrights and attributions specified by the author [101].

5.3.8 Comprehensibility

Dataset comprehensibility is identified if the publisher provides general information about the dataset (e.g. title, description, URI). In addition, if he indicates at least

one exemplary RDF file and SPARQL query and provides an active communication channel (mailing list, message board or e-mail) [7]. A model is considered to be comprehensible if there is no misuse of ontology annotations and that all the concepts are documented and annotated [23, 60].

5.3.9 Coherence

Coherence is the ability to interpret data as expected by the publisher or vocabulary maintainer [49]. The objective coherence measures are mainly associated with the modeling quality. A model is considered to be coherent when it does not contain undefined classes and properties [49], blank nodes [50], deprecated classes or properties [49], relations and mappings clashes [82], invalid inverse-functional values [49], cyclic hierarchical relations [91, 82, 23], solely transitive related concepts [23], redefinitions of existing vocabularies [49] and valueless associative relations [23].

5.3.10 Security

Security is a quality attribute that is measured on the dataset level. It is identified if the publishers use login credentials, SSL or SSH to provide access to their dataset, or if they only grant access to specific users [101].

5.4 An Extensible Objective Quality Assessment Framework

Roomba is built as a Command Line Interface (CLI) application using Node.js. Instructions on installing and running the framework are available on its public Github repository. Related functions are encapsulated into modules that can be easily plugged in/out the processing pipeline. Figure 5.1 shows the main steps which are the following: (i) Data portal identification; (ii) metadata extraction; (iii) instance and resource extraction; (iv) profile validation (v) profile and report generation.

For this paper, we have extended Roomba with a new quality module to measure datasets quality. We have implemented 7 submodules that will check various dataset quality indicators. Various additional quality measures can be easily plugged in/out.

Data portals can be considered as data access points providing tools to facilitate data publishing, sharing, searching and visualization. CKAN⁶ is the world's leading open-source data portal platform powering websites and the target of our tool. We have identified that most of the dataset quality issues can be assessed by examining the accompanying dataset metadata. Building a standard metadata model is not the scope of this paper, and since we focus on CKAN-based portals, we assess the

⁶<http://ckan.org>

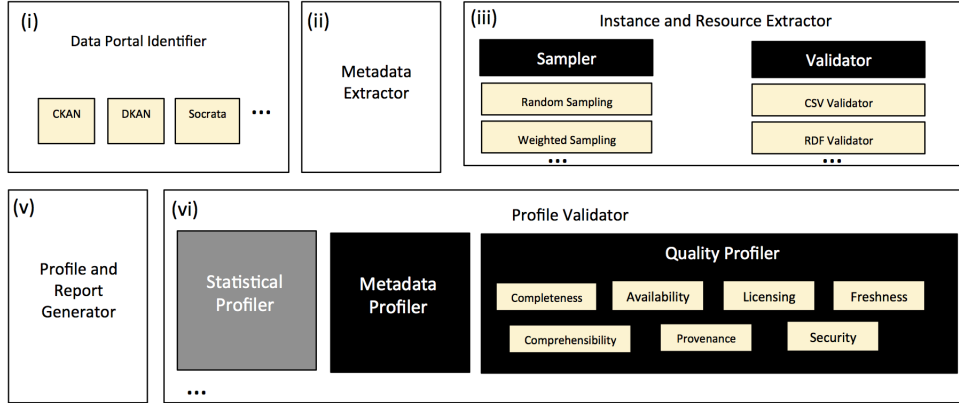


Figure 5.1: Processing pipeline for objective dataset quality assessment

quality issues using the CKAN standard model⁷. Table 5.2 shows the various quality indicators checked by our tool.

In our framework, we have presented 30 objective quality indicators related to dataset and links quality. The Roomba quality module is able to assess and score 23 of them. We excluded security related quality indicators as LOD cloud group members should not restrict access to their datasets.

5.4.1 Quality Score Calculation

A CKAN portal contains a set of datasets $\mathbf{D} = \{D_1, \dots, D_n\}$. We denote the set of resources $R_i = \{r_1, \dots, r_k\}$, groups $G_i = \{g_1, \dots, g_k\}$ and tags $T_i = \{t_1, \dots, t_k\}$ for $D_i \in \mathbf{D} (i = 1, \dots, n)$ by $\mathbf{R} = \{R_1, \dots, R_n\}$, $\mathbf{G} = \{G_1, \dots, G_n\}$ and $\mathbf{T} = \{T_1, \dots, T_n\}$ respectively.

Our quality framework contains a set of measures $\mathbf{M} = \{M_1, \dots, M_n\}$. We denote the set of quality indicators $Q_i = \{q_1, \dots, q_k\}$ for $M_i \in \mathbf{M} (i = 1, \dots, n)$ by $\mathbf{Q} = \{Q_1, \dots, Q_n\}$. Each quality indicator has a weight, context and a score $Q_i < weight, context, score >$. In Roomba, all the weights are equal and set to 1. However, they can be adjusted manually to rank the quality indicators. Each Q_i of M_i (for $i = 1, \dots, n$) is applied to one or more of the resources, tags or groups. The indicator context is defined where $\exists Q_i \in \mathbf{R} \cup \mathbf{G} \cup \mathbf{T}$.

The quality indicator score is based on a ratio between the number of violations \mathbf{V} and the total number of instances where the rule applies \mathbf{T} multiplied by the specified weight for that indicator.

$$Q \text{ weightedscore} = (V/T) * Q < weight > \quad (5.1)$$

Q weightedscore is an error ratio. A quality measure score should reflect the alignment of the dataset with respect to the quality indicators. The quality measure

⁷http://demo.ckan.org/api/3/action/package_show?id=adur_district_spending

Quality Indicator	Assessment Method
1	Check if there is a valid metadata file by issuing a <code>package.show</code> request to the CKAN API
2	Check if the <code>format</code> field for the dataset resources is defined and valid
3	Check the <code>resource.type</code> field with the following possible values <code>file</code> , <code>file.upload</code> , <code>api</code> , <code>visualization</code> , <code>code</code> , <code>documentation</code>
4	Check the resources <code>format</code> field for <code>meta/void</code> value
5	Check the resources <code>size</code> or the <code>triples</code> extras fields
6	Check the <code>format</code> and <code>mimetype</code> fields for resources
7	Check if the dataset has a <code>topic</code> tag and if it is part of a valid group in CKAN
9	Check if the dataset and all its resources have has a valid URI
18	Check if there is a dereferencable resource with a description containing string <code>dump</code>
19	Check if there is a dereferencable resource with <code>resource.type</code> of type <code>api</code>
20	Check if all the links assigned to the dataset and its resources are dereferencable
21	Check if the dataset contains valid <code>license.id</code> and <code>license.title</code>
22	Check if the <code>license.url</code> is dereferencable
24	Check if the dataset and its resources contain the following meta-data fields <code>metadata.created</code> , <code>metadata.modified</code> , <code>revision.timestamp</code> , <code>cache.last.updated</code>
25	Check if the <code>content-type</code> extracted from the a valid HTTP request is equal to the corresponding <code>mimetype</code> field.
26	Check if the <code>content-length</code> extracted from the a valid HTTP request is equal to the corresponding <code>size</code> field.
28,29	Check that all the links are valid HTTP scheme URIs
37	Check if there is at least one resource with a <code>format</code> value corresponding to one of <code>example/rdf+xml</code> , <code>example/turtle</code> , <code>example/ntriples</code> , <code>example/x-quads</code> , <code>example/rdfa</code> , <code>example/x-trig</code>
39	Check if the dataset and its tags and resources contain general metadata <code>id</code> , <code>name</code> , <code>type</code> , <code>title</code> , <code>description</code> , <code>URL</code> , <code>display_name</code> , <code>format</code>
40	Check if the dataset contain valid <code>author_email</code> or <code>maintainer_email</code> fields
44	Check if the dataset and its resources contain provenance metadata <code>maintainer</code> , <code>owner_org</code> , <code>organization</code> , <code>author</code> , <code>maintainer_email</code> , <code>author_email</code>
46	Check if the dataset contain and its resources contain versioning information <code>version</code> , <code>revision.id</code>

Table 5.2: Objective Quality Assessment Methods for CKANbased Data Portals

score \mathbf{M} is calculated by dividing the weighted quality indicator scores sum by the total number of instances in its context, as the following formula shows:

$$M = 1 - ((\sum_{i=1}^n Q \text{ weightedscore}) / |Q \text{ context}|) \quad (5.2)$$

5.4.2 Experiments and Analysis

In this section, we provide the experiments and evaluation of the proposed framework. All the experiments are reproducible by Roomba and their results are available on

its Github repository. We have run the framework on the LOD cloud containing 259 datasets at the time of writing this paper. We ran the instance and resource extractor in order to cache the metadata files for these datasets locally and ran the quality assessment process which took around two hours and a half hour on a 2.6 Ghz Intel Core i7 processor with 16GB of DDR3 memory machine.

Dataset Quality Report	
completeness quality Score	: 50.22%
availability quality Score	: 26.22%
licensing quality Score	: 19.59%
freshness quality Score	: 79.49%
correctness quality Score	: 72.06%
comprehensibility quality Score	: 31.62%
provenance quality Score	: 74.07%
Average total quality Score	: 50.47%
Quality Indicators Average Error %	
Quality Indicator : Supports multiple serializations:	11.35%
Quality Indicator : Has different data access points:	19.31%
Quality Indicator : Uses datasets description vocabularies:	88.80%
Quality Indicator : Existence of descriptions about its size:	86.30%
Quality Indicator : Existence of descriptions about its structure:	83.67%

Listing 5.1: Excerpt of the LOD cloud group quality report

We found out that licensing, availability and comprehensibility had the worst quality measures scores: 19.59%, 26.22% and 31.62% respectively. On the other hand, the LOD cloud datasets have good quality scores for freshness, correctness and provenance as most of the datasets have an average of 75% for each one of those measures.

Figure 5.2 shows the average errors percentage in quality indicators grouped by the corresponding measures. After examining the results, we notice that the worst quality indicators scores are for the comprehensibility measure where 99.61% of the datasets did not have valid exemplary RDF file (QI.37) and did not define valid point of contact (QI.40). Moreover, we noticed that 96.41% of the datasets queryable endpoints (SPARQL endpoints) failed to respond to direct queries (QI.19). After careful examination, we found that the cause was incorrect assignment for metadata fields. Data publishers specified the resource `format` field as an `api` instead of the specifying the `resource_type` field.

To drill down more on the availability issues, we generated a metadata profile assessment report using Roomba's metadata profiler. We found out that 25% of the datasets access information (being the dataset URL and any URL defined in its

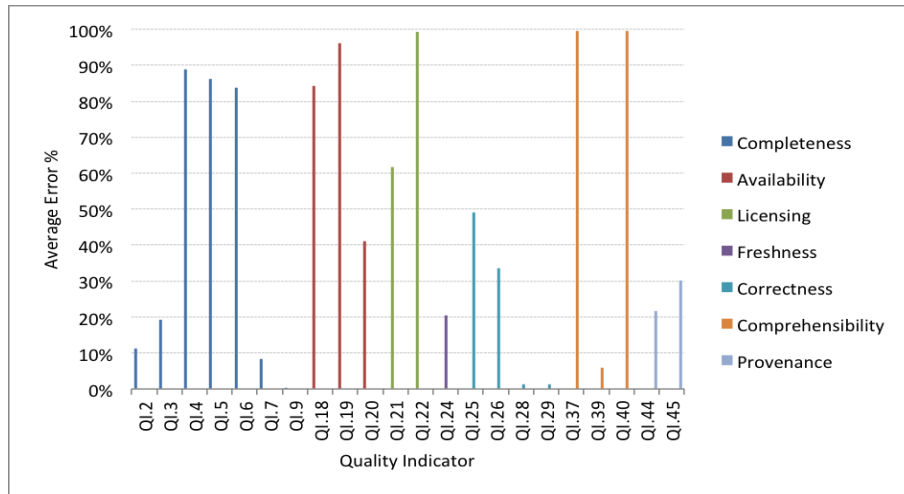


Figure 5.2: Average Error % per quality indicator for LOD group

groups) has issues related to them (missing or unreachable URLs). Three datasets (1.15%) did not have a URL defined while 45 datasets (17.3%) defined URLs were not accessible at the time writing this paper. Out of the 1068 defined resources 31.27% were not reachable. All these issues resulted in a 26.22% average availability score. This can highly affect the usability of those datasets especially in an enterprise context.

5.5 Linked Data Quality Tools

In this section, we present the results of our survey on the Linked Data quality tools. There exists a number of data quality frameworks and tools that are either standalone or implemented as modules in data integration tools. These approaches can be classified into automatic, semi-automatic, manual or crowdsourced approaches.

5.5.1 Information Quality

RDF is the standard to model information in the Semantic Web. Linked Data publishers can pick from a plethora of tools that can automatically check their RDF files for quality problems⁸. Syntactic RDF checkers are able to detect errors in RDF documents like the W3C RDF Validator⁹, RDF:about validator and Converter¹⁰ and The Validating RDF Parser (VRP)¹¹. The RDF Triple-Checker¹² is an online tool that helps find typos and common errors in RDF data. Vapour¹³ [34] is a validation

⁸<http://www.w3.org/2001/sw/wiki/SWValidators>

⁹<http://www.w3.org/RDF/Validator/>

¹⁰<http://rdfabout.com/demo/validator/>

¹¹<http://139.91.183.30:9090/RDF/VRP/index.html>

¹²<http://graphite.ecs.soton.ac.uk/checker/>

¹³<http://validator.linkeddata.org/vapour>

service to check whether semantic Web data is correctly published according to the current best practices [95].

ProLOD [14], ProLOD++ [1], Aether [71] and LODStats [10] are not purely quality assessment tools. They are Linked Data profiling tools providing clustering and labeling capabilities, schema discovery and statistics about data types and patterns. The statistics are about properties distribution, link-to-literal ratio, number of entities and RDF triples, average properties per entity and average error.

5.5.2 Modeling Quality

Reusing existing ontologies is a common practice that Linked Data publishers are always trying to adopt. However, ontologies and vocabularies development is often a long error-prone process especially when many contributors are working consecutively or collaboratively [81]. This can introduce deficiencies such as redundant concepts or conflicting relationships [83]. Getting to choose the right ontology or vocabulary is vital to ensure modeling correctness and consistency.

5.5.2.1 Semi-automatic Approaches

DL-Learner [54] uses supervised machine learning techniques to learn concepts from user-provided examples. CROCUS [33] applies a cluster-based approach for instance-level error detection. It validates identified errors by non-expert users and iterate to reach higher quality ontologies that can be safely used in industrial environments.

5.5.2.2 Automatic Approaches

qSKOS¹⁴ [23] scans SKOS vocabularies to provide reports on vocabulary resources and relations that are problematic. PoolParty checker¹⁵ is an online service based on qSKOS. Skosify [82] supports OWL and RDFS ontologies by converting them into well-structured SKOS vocabularies. It includes automatic correction abilities for quality issues that have been observed by reviewing vocabularies on the Web. The OOPS! pitfall scanner [86] evaluates OWL ontologies against a rules catalog and provides the user with a set of guidelines to solve them. ASKOSI¹⁶ retrieves vocabularies from different sources, stores and displays the usage frequency of the different concepts used by different applications. It promotes reusing existing information systems by providing better management and presentation tools.

Some errors in RDF will only appear after reasoning (incorrect inferences). In [37, 94] the authors perform quality checking on OWL ontologies using integrity constraints involving the Unique Name Assumption (UNA) and the Closed World As-

¹⁴<https://github.com/cmader/qSKOS>

¹⁵<http://www.poolparty.biz/>

¹⁶<http://www.w3.org/2001/sw/wiki/ASKOSI>

sumption (CWA). Pellet¹⁷ provides reasoning services for OWL ontologies. It incorporates a number of heuristics to detect and repair quality issues among disjoint properties, negative property assertions and reflexive, irreflexive, symmetric, and anti-symmetric properties. Eyeball¹⁸ provides quality inspection for RDF models (including OWL). It provides checks for a variety of problems including the usage of unknown predicates, classes, poorly formed namespaces, literal syntax validation, type consistency and other heuristics. RDF:Alerts¹⁹ provides validation for many issues highlighted in [49] like misplaced, undefined or deprecated classes or properties.

5.5.3 Dataset Quality

Considering the large amount of available datasets in the Linked Open Data, users have a hard time trying to identify appropriate datasets that suit certain tasks. The most adopted approaches are based on link assessment. Provenance-based approaches and entity-based approaches are also used to compute not only dataset rankings, but also rankings on the entity level.

5.5.3.1 Manual Ranking Approaches

Sieve [85] is a framework for expressing quality assessment and fusion methods. It is implemented as a component of the Linked Data Integration Framework (LDIF)²⁰. Sieve leverages the LDIF provenance metadata as quality indicators to produce quality assessment scores. However, despite its nice features, it is only targeted to perform data fusion based on user-configurable conflict resolution tasks. Moreover, since Sieve main input is provenance metadata, it is only limited to domains that can provide such metadata associated with their data.

SWIQA [44] is a framework providing policies or formulas controlling information quality assessment. It is composed of three layers: data acquisition, query and ontology layers. It uses query templates based on the SPARQL Inferencing Notation (SPIN)²¹ to express quality requirements. The queries are built to compute weighted and unweighted quality scores. At the end of the assessment, it uses vocabulary elements to annotate important values of properties and classes, assigning inferred quality scores to ontology elements and classifying the identified data quality problems.

5.5.3.2 Crowd-sourcing Approaches

There are several quality issues that can be difficult to spot and fix automatically. In [2] the authors highlight the fact that the RDFification process of some data can

¹⁷<http://clarkparsia.com/pellet>

¹⁸<http://jena.sourceforge.net/Eyeball/>

¹⁹<http://swse.deri.org/RDFAlerts/>

²⁰<http://ldif.wbsg.de/>

²¹<http://spinrdf.org/>

be more challenging than others, leading to errors in the Linked Data provisioning process that needs manual intervention. This can be more visible in datasets that have been semi-automatically translated to RDF from their primary source (the best example for this case is DBpedia [21]). The authors introduce a methodology to adjust crowdsourcing input from two types of audience: 1) Linked Data experts, researchers and enthusiasts through a contest to find and classify erroneous RDF triples and 2) Crowdsourcing through the Amazon Mechanical Turk²².

TripleCheckMate [35] is a crowdsourcing tool used by the authors to run out their assessment supported by a semi-automatic quality verification metrics. The tool allows users to select resources, identify and classify possible issues according to a pre-defined taxonomy of quality problems. It measures inter-rater agreements, meaning that the resources defined are checked multiple times. These features turn out to be extremely useful to analyze the performance of users and allow better identification of potential quality problems. TripleCheckMate is used to identify accuracy issues in the object extraction (completeness of the extraction value for object values and data types), relevancy of the extracted information, representational consistency and interlinking with other datasets.

5.5.3.3 Semi-automatic Approaches

Luzzu [56] is a generic Linked Data quality assessment framework. It can be easily extended through a declarative interface to integrate domain specific quality measures. The framework consists of three stages closely corresponding to the methodology in [90]. They believe that data quality cannot be tackled in isolation. As a result, they require domain experts to identify quality assessment metrics in a schema layer. Luzzu is ontology driven. The core vocabulary for the schema layer is the Dataset Quality Ontology (daQ) [30]. Any additional quality metrics added to the framework should extend it.

RDFUnit²³ is a tool centered around the definition of data quality integrity constraints [64]. The input is a defined set of test cases (which can be generated manually or automatically) presented in SPARQL query templates. One of the main advantages for this approach is the ability to discover quality problems beyond conventional quality heuristics by encoding domain specific semantics in the test cases.

LiQuate [89] is based on probabilistic models to analyze the quality of data and links. It consists of two main components: A Bayesian Network builder and an ambiguity detector. They rely on data experts to represent probabilistic rules. LiQuate identifies redundancies (redundant label names for a given resource), incompleteness (incomplete links among a given set of resources) and inconsistencies (inconsistent links).

²²<https://www.mturk.com/>

²³<http://github.com/AKSW/RDFUnit>

Quality Assessment of Data Sources (Flemming’s Data Quality Assessment Tool)²⁴ calculates data quality scores based on manual user input. The user should assign weights to the predefined quality metrics and answer a series of questions regarding the dataset. These include, for example, the use of obsolete classes and properties by defining the number of described entities that are assigned disjoint classes, the usage of stable URIs and whether the publisher provides a mailing list for the dataset. The main disadvantage for using this tool is the manual intervention which requires deep knowledge in the dataset examined. Moreover, the tool lacks support for several quality concerns like completeness or consistency.

LODGRfine [74] is the Open Refine²⁵ of Linked Data. It does not act as a quality assessment tool, but it is powerful in cleaning and refining raw instance data. LODGRfine can help detect duplicates, empty values, spot inconsistencies, extract Named Entities, discover patterns and more. LODGRfine helps in improving the quality of the dataset by improving the quality of the data at the instance level.

5.5.3.4 Automatic Ranking Approaches

The Project Open Data Dashboard²⁶ tracks and measures how US government websites implement the Open Data principles to understand the progress and current status of their public data listings. A validator analyzes machine readable files e.g. JSON files for automated metrics like the resolved URLs, HTTP status and content-type. However, deep schema information about the metadata is missing like description, license information or tags.

Similarly on the LOD cloud, the Data Hub LOD Validator²⁷ gives an overview of Linked Data sources cataloged on the Data Hub. It offers a step-by-step validator guidance to check a dataset completeness level for inclusion in the LOD cloud. The results are divided into four different compliance levels from basic to reviewed and included in the LOD cloud. Although it is an excellent tool to monitor LOD compliance, it still lacks the ability to give detailed insights about the completeness of the metadata and overview on the state of the whole LOD cloud group and is very specific to the LOD cloud group rules and regulations.

Link-based Approaches

The basic idea behind link assessment tools is to provide rankings for datasets based on the cardinality and types of the relationships with other datasets. Traditional link analysis has proven to be an effective way to measure the quality of Web documents search. Algorithms like PageRank [68] and HITS [62] became successful based on the assumption that a certain Web document is considered to have higher

²⁴<http://linkeddata.informatik.hu-berlin.de/LDSrcAss/datenquelle.php>

²⁵<http://openrefine.org/>

²⁶<http://labs.data.gov/dashboard/>

²⁷<http://validator.lod-cloud.net/>

importance or rank if it has more incoming links than other Web documents [18][92]. However, the basic assumption that links are equivalent does not suit the heterogeneous nature of links in the Linked Open Data. Thus, the previous approaches fall short to provide reliable rankings as the types of the links can have a direct impact on the ranking computation [78]. The first adaption of PageRank for Semantic Web resources was the Ontology Rank algorithm implemented in the Swoogle search engine [36]. They use a rational random surfing model that takes into account the different types of links between discovered sets and compute rankings based on three levels of granularity: documents, terms and RDF graphs. ReConRank [4] rankings are computed at query time based on two levels of granularity: resources and context graphs. DING [78] adapted the PageRank to rank datasets based on their interconnections. DING can also automatically assign weights to different link types based on the nature of the predicate involved in the link. Broken links are a major threat to Linked Data. They occur when resources are removed, moved or updated. DSNotify²⁸[11] is a framework that informs data consumers about the various types of events that occur on data sources. Their approach is based on an indexing infrastructure that extracts feature vectors and stores them to an index. A monitoring module detects events on sources and write them to a central event log which pushes notifications to registered applications. LinkQA [46] is a fully automated approach which takes a set of RDF triples as an input and analyzes it to extract topological measures (links quality). However, the authors depend only on five metrics to determine the quality of data (degree, clustering coefficient, centrality, sameAs chains and descriptive richness through sameAs).

Provenance-based Approaches

Provenance-based assessment methods are an important step towards transparency of data quality in the Semantic Web. In [80]²⁹ the authors use a provenance model as an assessment method to evaluate the timeliness of Web data. Their model identifies types of “provenance elements” and the relationships between them. Provenance elements are classified into three types: actors, executions and artifacts. The assessment procedure is divided into three steps: 1) Creating provenance graph based on the defined model 2) Annotating the graph with impact values 3) Calculating the information quality score. In [40] the authors describe a set of provenance-based assessment metrics to support quality assessment and repair in Linked Open Data. They rely on both data and metadata and use indicators like the source reputation, freshness and plausibility. In [48] the authors introduce the notion of naming authority which connects an identifier with the source to establish a connection to its provenance. They construct a naming authority graph that acts as input to derive PageRank scores for the data sources.

Entity-based Approaches

²⁸<http://www.cibiv.at/~niko/dsnotify/>

²⁹<http://trdf.sourceforge.net>

Sindice [97] uses a set of techniques to rank Web data. They use a combination of query dependent and query independent rankings implemented in the Semantic Information Retrieval Engine (SIREn)³⁰ to produce a final entity rank. Their query dependent approach rates individual entities by aggregating the the score of the matching terms with a term frequency - inverse subject frequency (tf-isf) algorithm. Their query independent ranking is done using hierarchical links analysis algorithms [31]. The combination of these two approaches is used to generate a global weighted rank based on the dataset, entities and links ranks.

5.5.4 Queryable End-point Quality

The availability of Linked Data is highly dependent on the performance qualities of its queryable end-points. The standard query language for Semantic Web resources is SPARQL. As a result, we focus on tools measuring the quality of SPARQL endpoints. In [19]³¹ the authors present their findings to measure the discoverability of SPARQL endpoints by analyzing how they are located and the metadata used to describe them. In addition to that, they also analyze endpoints interoperability by identifying features of SPARQL 1.0 and SPARQL 1.1 that are supported. The authors tackled the endpoints efficiency by testing the time taken to answer generic, content-agnostic SPARQL queries over HTTP.

Summary

We notice that there is a plethora of tools (syntactic checkers or statistical profilers) that automatically check the quality of information at the entities level. Moreover, various tools can automatically check the models against the objective quality indicators mentioned. OOPS! covers all of them with additional support for the other common modeling pitfalls in [60]. PoolParty covers also a wide set of those indicators but it targets SKOS vocabularies only. However, we notice a lack in automatic tools to check the dataset quality especially in its completeness, licensing and provenance measures. Table 3 summarizes the automatic dataset quality approaches that have implemented tools (full circle denotes full quality indicator assessment, while half circle denoted partial assessment). As can be seen in this table Roomba covers most of the quality indicators with its focus on completeness, correctness provenance and licensing. Roomba is not able to check the existence of information about the kind and number of used vocabularies (QI.8), license permissions, copyrights and attributes (QI.23), exemplary SPARQL query (QI.38), usage of provenance vocabulary (QI.45) and is not able to check the dataset for syntactic errors (QI.27).

These shortcomings are mainly due to the limitations in the CKAN dataset model. However, syntactic checkers and additional modules to examine vocabularies usage could be easily integrated in Roomba to fix QI.27, QI.8 and QI.45. Roomba's metadata quality profiler can fix QI.23 as we have manually created a mapping file stan-

³⁰<http://siren.sindice.com/>

³¹<http://labs.mondeca.com/sparqlEndpointsStatus/>

standardizing the set of possible license names and their information³². We have also used the open source and knowledge license information³³ to normalize license information and add extra metadata like the domain, maintainer and open data conformance.

5.6 Conclusions and Future Work

In this paper, we have presented a comprehensive objective quality framework applied to the Linked Open Data. We have built upon previous efforts with focus on objective data quality measures. We have identified a total of 64 quality indicators that were mapped when suitable to four main categories (entity, dataset, links, models). We have also surveyed more than 30 different tools that measure different quality aspects of Linked Open Data. We identified several gaps in the current tools and identified the need for a comprehensive evaluation and assessment framework and specifically for measuring quality on the dataset level. As a result, we presented an extension of Roomba (An extensible tool to assess and generate dataset profiles) that covers 82% of the suggested datasets objective quality indicators. Based on our experiments running Roomba on the LOD cloud, we discovered that the general state of the datasets needs attention as most of them have low completeness, provenance, licensing and comprehensibility quality scores.

In future work, we plan to integrate tools assessing models quality in addition to syntactic checkers with Roomba. This will provide a complete coverage of the proposed quality indicators. We also intend to suggest ranked quality indicators to improve the quality report. We also plan to run this tool on various CKAN based data portals and schedule periodic reports to monitor their quality evolution. Finally, at some stage, we plan to extend this tool for other data portal types like DKAN and Socrata.

³²<https://github.com/ahmadassaf/opendata-checker/blob/master/util/licenseMappings.json>

³³<https://github.com/okfn/licenses>

Conclusions and Future Perspectives

In this chapter, we summarize the major achievements of this thesis and we give an outlook on future perspectives.

6.1 Achievements

6.2 Perspectives

List of Publications

A.1 Journals

A.2 Conferences and Workshops

Bibliography

- [1] Z. Abedjan, T. Gruetze, A. Jentzsch, and F. Naumann. Profiling and mining RDF data with ProLOD++. In *30th IEEE International Conference on Data Engineering (ICDE)*, pages 1198–1201, 2014. 28, 58
- [2] Maribel Acosta, Amrapali Zaveri, Elena Simperl, and Dimitris Kontokostas. Crowdsourcing Linked Data quality assessment. In *12th International Semantic Web Conference (ISWC)*, 2013. 50, 52, 59
- [3] Assaf Ahmad, Sénart Aline, and Troncy Raphaël. Roomba: Automatic Validation, Correction and Generation of Dataset Metadata. In *24th World Wide Web Conference (WWW), Demos Track*, Florence, Italy, 2015. 17
- [4] Hogan Aidan, Harth Andreas, and Decker Stefan. ReConRank: A Scalable Ranking Method for Semantic Web Data with Context. In *2nd Workshop on Scalable Semantic Web Knowledge Base Systems*, 2006. 62
- [5] K. Alexander, R. Cyganiak, M. Hausenblas, and J. Zhao. Describing Linked Datasets. In *2nd International Workshop on Linked Data on the Web (LDOW)*, 2009. 29
- [6] Miles Alistair and Bechhofer Sean. SKOS Simple Knowledge Organization System Reference. W3C Recommendation, 2009. <http://www.w3.org/TR/skos-reference/>. 50
- [7] Flemming Annika. Quality Characteristics of Linked Data Publishing Data-sources. Master’s thesis, Humboldt-Universitt zu Berlin, 2010. 49, 50, 51, 53
- [8] Isaac Antoine and Summers Ed. SKOS Simple Knowledge Organization System Primer. W3C Working Group Note, 2009. 50, 52
- [9] Ahmad Assaf and Aline Senart. Data Quality Principles in the Semantic Web. In *6th International Conference on Semantic Computing ICSC ’12*, 2012. 46, 49
- [10] Sören Auer, Jan Demter, Michael Martin, and Jens Lehmann. LODStats - an Extensible Framework for High-performance Dataset Analytics. In *18th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, pages 353–362, Galway, Ireland, 2012. 28, 58
- [11] Haslhofer Bernhard and Popitsch Niko. DSNotify: Detecting and Fixing Broken Links in Linked Data Sets. In *8th International Workshop on Web Semantics*, 2009. 62

- [12] Stvilia Besiki, Gasser Les, Twidale Michael B., and Smith Linda C. A framework for information quality assessment. *Journal of the American Society for Information Science and Technology*, 2007. 45
- [13] Christian Bizer and Richard Cyganiak. Quality-driven information filtering using the wiqa policy framework. *Jorunal of Web Semantics*, 7(1), 2009. 46
- [14] C. Bohm, F. Naumann, Z. Abedjan, D. Fenz, T. Grutze, D. Hefenbrock, M. Pohl, and D. Sonnabend. Profiling linked open data with ProLOD. In *26th International Conference on Data Engineering Workshops (ICDEW)*, 2010. 28, 58
- [15] Christoph Böhm, Gjergji Kasneci, and Felix Naumann. Latent Topics in Graph-structured Data. In *21st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2663–2666, Maui, Hawaii, USA, 2012. 29
- [16] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *ACM International Conference on Management of Data (SIGMOD)*, 2008. 25
- [17] D Boyd and Kate Crawford. Six provocations for big data. *A Decade in Internet Time: Symposium on the Dynamics of the Internet and Society*, 2011. 9, 25, 33, 45
- [18] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. In *7th International Conference on World Wide Web (WWW'98)*, 1998. 62
- [19] C Buil-Aranda and Aidan Hogan. SPARQL Web-Querying Infrastructure: Ready for Action? In *12th International Semantic Web Conference (ISWC)*, 2013. 63
- [20] Bizer Christian. Evolving the Web into a Global Data Space. In *28th British National Conference on Advances in Databases*, 2011. 9, 27
- [21] Bizer Christian, Lehmann Jens, Kobilarov Georgi, Auer Sören, Becker Christian, Cyganiak Richard, and Hellmann Sebastian. DBpedia - A Crystallization Point for the Web of Data. *Journal of Web Semantics*, 7(3), 2009. 25, 46, 60
- [22] Bizer Christian, Heath T, and Berners-Lee T. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 2009. 25

- [23] Mader Christian, Haslhofer Bernhard, and Isaac Antoine. Finding quality issues in SKOS vocabularies. *Theory and Practice of Digital Libraries*, 2012. 49, 50, 51, 52, 53, 58
- [24] BöhM Christoph, Lorey Johannes, and Naumann Felix. Creating void Descriptions for Web-scale Data. *Journal of Web Semantics*, 9(3):339–345, 2011. 12, 27
- [25] Marco Cornolti, Paolo Ferragina, and Massimiliano Ciaramita. A Framework for Benchmarking Entity-annotation Systems. In *22nd World Wide Web Conference (WWW)*, 2013. 29
- [26] Richard Cyganiak, Holger Stenzhorn, Renaud Delbru, Stefan Decker, and Giovanni Tummarello. Semantic Sitemaps: Efficient and Flexible Access to Datasets on the Semantic Web. In *5th European Semantic Web Conference (ESWC)*, pages 690–704, Tenerife, Spain, 2008. 28
- [27] Richard Cyganiak, Jun Zhao, Michael Hausenblas, and Keith Alexander. Describing Linked Datasets with the VoID Vocabulary. W3C Note, 2011. <http://www.w3.org/TR/void/>. 27
- [28] Mathieu d’Aquin and Enrico Motta. Watson, More Than a Semantic Web Search Engine. *Semantic Web Journal*, 2011. 29
- [29] Reynolds Dave. The Organization Ontology. W3C Recommendation, 2014. <http://www.w3.org/TR/vocab-org>. 15
- [30] Jeremy Debattista, Christoph Lange, and Sören Auer. daQ, an Ontology for Dataset Quality Information. In *Workshop on Linked Data on the Web co-located with the 23rd International World Wide Web Conference (WWW 2014)*, 2014. 16, 60
- [31] Renaud Delbru, Nickolai Toupikov, and Michele Catasta. Hierarchical link analysis for ranking web data. In *7th European Semantic Web Conference (ESWC)*, 2010. 29, 63
- [32] Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, Davide Romito, and Markus Zanker. Linked Open Data to Support Content-based Recommender Systems. In *8th International Conference on Semantic Systems - I-SEMANTICS ’12*, 2012. 46
- [33] Cherix Didier, Usbeck Ricardo, Both Andreas, and Lehmann Jens. CROCUS: Cluster-based ontology data cleansing. In *2nd International Workshop on Semantic Web Enterprise Adoption and Best Practice*, 2014. 58

- [34] Berrueta Diego, Fernández Sergio, and Frade Iván. Cooking HTTP content negotiation with Vapour. In *4th Workshop on Scripting for the Semantic Web (SFSW'08)*, 2008. 57
- [35] Kontokostas Dimitris, Zaveri Amrapali, Auer Sören, and Lehmann J. TripleCheckMate: A Tool for Crowdsourcing the Quality Assessment of Linked Data. *4th Conference on Knowledge Engineering and Semantic Web*, 2013. 60
- [36] L Ding, Tim Finin, A Joshi, R Pan, and RS Cost. Swoogle: A semantic web search and metadata engine. In *13st ACM International Conference on Information and Knowledge Management (CIKM)*, 2004. 29, 62
- [37] Sirin Evren, Smith Michael, and Wallace Evan. Opening, Closing Worlds - On Integrity Constraints. In *5th OWLED Workshop on OWL: Experiences and Directions*, 2008. 58
- [38] Maali Fadi and Erickson John. Data Catalog Vocabulary (DCAT). W3C Recommendation, 2014. <http://www.w3.org/TR/vocab-dcat/>. 10, 11, 27
- [39] Besnik Fetahu, Stefan Dietze, Bernardo Pereira Nunes, Marco Antonio Casanova, Davide Taibi, and Wolfgang Nejdl. A Scalable Approach for Efficiently Generating Structured Dataset Topic Profiles. In *11th European Semantic Web Conference (ESWC)*, 2014. 29, 33
- [40] Giorgos Flouris, Yannis Roussakis, and M Poveda-Villalón. Using provenance for quality assessment and repair in linked open data. In *2nd Joint Workshop on Knowledge Evolution and Ontology Dynamics (EvoDyn'12)*, 2012. 49, 50, 52, 62
- [41] Benedikt Forchhammer, Anja Jentzsch, and Felix Naumann. LODOP - Multi-Query Optimization for Linked Data Profiling Queries. In *International Workshop on Dataset PROFiling and Federated Search for Linked Data (PROFILES)*, Heraklion, Greece, 2014. 29
- [42] Matias Frosterus, Eero Hyvönen, and Joonas Laitio. Creating and Publishing Semantic Metadata about Linked and Open Datasets. In *Linking Government Data*. 2011. 28
- [43] Matias Frosterus, Eero Hyvönen, and Joonas Laitio. DataFinland - A Semantic Portal for Open and Linked Datasets. In *8th Extended Semantic Web Conference (ESWC)*, pages 243–254, 2011. 29
- [44] C Fürber and M Hepp. SWIQA - A Semantic Web information quality assessment framework. 2011. 59

- [45] Tummarello Giovanni, Cyganiak Richard, Catasta Michele, Danielczyk Szymon, Delbru Renaud, and Decker Stefan. Sig.ma: Live views on the Web of data. *Journal of Web Semantics*, 8(4), 2010. 29
- [46] Christophe Guéret, Paul T. Groth, Claus Stadler, and Jens Lehmann. Assessing Linked Data Mappings Using Network Measures. In *9th European Semantic Web Conference (ESWC)*, 2012. 49, 51, 62
- [47] Andreas Harth, Katja Hose, Marcel Karnstedt, Axel Polleres, Kai-Uwe Sattler, and Jürgen Umbrich. Data Summaries for On-demand Queries over Linked Data. In *19th World Wide Web Conference (WWW)*, 2010. 29
- [48] Andreas Harth, Sheila Kinsella, and Stefan Decker. Using naming authority to rank data and ontologies for web search. In *8th International Semantic Web Conference (ISWC)*, 2009. 62
- [49] Aidan Hogan, Andreas Harth, Alexandre Passant, Stefan Decker, and Axel Polleres. Weaving the pedantic web. 2010. 49, 50, 51, 52, 53, 59
- [50] Aidan Hogan, JüRgen Umbrich, Andreas Harth, Richard Cyganiak, Axel Polleres, and Stefan Decker. An empirical survey of Linked Data conformance. *Journal of Web Semantics*, 2012. 49, 50, 52, 53
- [51] Robert Isele, Jürgen Umbrich, Christian Bizer, and Andreas Harth. LDspider: An Open-source Crawling Framework for the Web of Linked Data. In *9th International Semantic Web Conference (ISWC), Posters & Demos Track*, 2010. 36, 37, 39
- [52] Prateek Jain, Pascal Hitzler, Krzysztof Janowicz, and Chitra Venkatramani. There's No Money in Linked Data, 2013. <http://knoesis.wright.edu/faculty/pascal/pub/nomoneylod.pdf>. 9, 33
- [53] Manyika James and Doshi Elizabeth Almasi. Open data: Unlocking innovation and performance with liquid information. Technical report, McKinsey Business Technology Office, 2001. 33
- [54] Lehmann Jens and Sonnenburg Soeren. DL-Learner: Learning Concepts in Description Logics. *Journal of Machine Learning Research*, 2009. 58
- [55] Anja Jentzsch. Profiling the Web of Data. In *13th International Semantic Web Conference (ISWC), Doctoral Consortium*, Trentino, Italy, 2014. 28
- [56] Debattista Jeremy, Londoño Santiago, Lange Christoph, and Auer Sören. LUZZU - A framework for linked data quality assessment. *CoRR*, abs/1412.3750, 2014. 60

-
- [57] Joseph. M. Juran and A. Blanton Godfrey. *Juran's quality handbook*. McGraw Hill, 1999. 45
- [58] Kahn Beverly K., Strong Diane M., and Wang Richard Y. Information quality benchmarks: product and service performance. *Communications of the ACM*, 2002. 45
- [59] Tobias Käfer, Ahmed Abdelrahman, Jürgen Umbrich, Patrick O'Byrne, and Aidan Hogan. Observing Linked Data Dynamics. In *10th European Semantic Web Conference (ESWC)*, 2013. 29
- [60] C.Maria Keet, María del Carmen Suárez-Figueroa, and María Poveda-Villalón. The Current Landscape of Pitfalls in Ontologies. In *International Conference on Knowledge Engineering and Ontology Development (KEOD)*, 2013. 49, 50, 51, 52, 53, 63
- [61] Shahan Khatchadourian and Mariano P. Consens. ExpLOD: Summary-based Exploration of Interlinking and RDF Usage in the Linked Open Data Cloud. In *7th Extended Semantic Web Conference (ESWC)*, pages 272–287, Heraklion, Greece, 2010. 28
- [62] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *ACM Journal*, 1999. 61
- [63] Mathias Konrath, Thomas Gottron, Steffen Staab, and Ansgar Scherp. SchemEX - Efficient Construction of a Data Catalogue by Stream-based Indexing of Linked Data. *Journal of Web Semantics*, 16, 2012. 29
- [64] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, Roland Cornelissen, and Amrapali Zaveri. Test-driven Evaluation of Linked Data Quality. In *23rd International Conference on World Wide Web (WWW'14)*, 2014. 60
- [65] Kovács-Láng. Global Terrestrial Observing System. Technical report, GTOS Central and Eastern European Terrestrial Data Management and Accessibility Workshop, 2000. 27
- [66] S. Lalithsena, P. Hitzler, A. Sheth, and P. Jain. Automatic Domain Identification for Linked Open Data. In *IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, pages 205–212, 2013. 26, 28, 29
- [67] Andreas Langegger and Wolfram Woss. RDFStats - An Extensible RDF Statistics Generator and Library. In *20th International Workshop on Database and Expert Systems Application (DEXA)*, pages 79–83, 2009. 28

- [68] Page Lawrence, Brin Sergey, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, 1998. 45, 61
- [69] Jure Leskovec and Christos Faloutsos. Sampling from Large Graphs. In *12th ACM International Conference on Knowledge Discovery and Data Mining (KDD'12)*, 2006. 33
- [70] Huiying Li. Data Profiling for Semantic Web Data. In *International Conference on Web Information Systems and Mining (WISM)*, pages 472–479, 2012. 26, 28
- [71] Eetu Mäkelä. Aether - Generating and Viewing Extended VoID Statistical Descriptions of RDF Datasets. In *11th European Semantic Web Conference (ESWC), Demo Track*, Heraklion, Greece, 2014. 28, 58
- [72] Nicolas Marie, Fabien Gandon, Myriam Ribière, and Florentin Rodio. Discovery Hub: On-the-fly Linked Data Exploratory Search. In *The 9th International Conference on Semantic Systems*, 2013. 46
- [73] Brümmer Martin, Baron Ciro, Ermilov Ivan, Freudenberg Markus, Kontokostas Dimitris, and Hellmann Sebastian. DataID: Towards Semantically Rich Metadata for Complex Datasets. In *10th International Conference on Semantic Systems*, 2014. 10, 15
- [74] Verlic Mateja. LODGrefine - LOD-enabled Google Refine in Action. In *8th International Conference on Semantic Systems - I-SEMANTICS '12*, 2012. 61
- [75] Schmachtenberg Max, Bizer Christian, and Paulheim Heiko. Adoption of the Linked Data Best Practices in Different Topical Domains. In *13th International Semantic Web Conference (ISWC)*, 2014. 36, 37, 39
- [76] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia Spotlight: Shedding Light on the Web of Documents. In *7th International Conference on Semantic Systems*, 2011. 46
- [77] Hausenblas Michael, Halb Wolfgang, Raimond Yves, Feigenbaum Lee, and Ayers Danny. SCOVO: Using Statistics on the Web of Data. In *ESWC*, 2009. 15
- [78] Toupikov Nickolai, Umbrich J, and Delbru Renaud. DING! Dataset ranking using formal descriptions. In *2nd International Workshop on Linked Data on the Web (LDOW)*, 2009. 62

- [79] Andriy Nikolov, Mathieu d'Aquin, and Enrico Motta. What Should I Link to? Identifying Relevant Sources and Classes for Data Linking. In *Joint International Semantic Technology Conference (JIST)*, 2011. 29
- [80] Hartig Olaf and Zhao Jun. Using web data provenance for quality assessment. In *8th International Semantic Web Conference (ISWC)*, 2009. 62
- [81] Suominen Osma and Mader Christian. Assessing and Improving the Quality of SKOS Vocabularies. *Journal on Data Semantics*, 2013. 49, 58
- [82] Suominen Osma and Hyvönen Eero. Improving the quality of SKOS vocabularies with skosify. In *The 18th International Conference on Knowledge Engineering and Knowledge Management*, 2012. 50, 52, 53, 58
- [83] Harpring Patricia. *Introduction to Controlled Vocabularies: Terminology for Art, Architecture, and Other Cultural Works*. Getty Research Institute, 2010. 58
- [84] Archer Phil and Shukair Gofran. Asset Description Metadata Schema (ADMS). W3C Working Group Note, 2013. <http://www.w3.org/TR/vocab-adms>. 11
- [85] Mendes PN, Mühleisen Hannes, and Bizer Christian. Sieve: linked data quality assessment and fusion. 2012. 59
- [86] Mara Poveda-Villalón, MariCarmen Suárez-Figueroa, and Asunción Gmez-Pérez. Validating Ontologies with OOPS! In *18th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, 2012. 58
- [87] NISO Press. Understanding Metadata. Technical report, National Information Standards Organization, 2004. 9
- [88] Iannella Renato and McKinney James. vCard Ontology - for describing People and Organizations. W3C Interest Group Note, 2014. <http://www.w3.org/TR/vcard-rdf>. 15
- [89] Edna Ruckhaus, Oriana Baldizan, and Maria-Esther Vidal. Analyzing Linked Data Quality with LiQuate. In *11th European Semantic Web Conference (ESWC)*, 2014. 60
- [90] Anisa Rula and Amrapali Zaveri. Methodology for Assessment of Linked Data Quality. In *1st Workshop on Linked Data Quality (LDQ)*, 2014. 48, 60
- [91] Dagobert Soergel. Thesauri and ontologies in digital libraries. In *2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, 2002. 50, 53
- [92] Chakrabarti Soumen, Dom Byron E., S. Kumar Ravi, Raghavan Prabhakar, Rajagopalan Sridhar, Tomkins Andrew, Gibson David, and Kleinberg Jon. Mining the web's link structure. *Computer*, 1999. 62

- [93] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Core of Semantic Knowledge. In *16th International World Wide Web Conference (WWW)*, 2007. 25
- [94] Jiao Tao, Li Ding, and Deborah L. McGuinness. Instance Data Evaluation for Semantic Web-Based Knowledge Management Systems. In *42nd Hawaii International Conference on System Sciences, HICSS'09*, pages 1–10, 2009. 58
- [95] Berners-Lee Tim. Linked Data - Design Issues. W3C Personal Notes, 2006. <http://www.w3.org/DesignIssues/LinkedData>. 16, 48, 58
- [96] Lebo Timothy, Sahoo Satya, and McGuinness Deborah. PROV-O: The PROV Ontology. W3C Recommendation, 2013. <http://www.w3.org/TR/prov-o>. 15, 52
- [97] Giovanni Tummarello, Renaud Delbru, and Eyal Oren. Sindice.com: Weaving the open linked data. In *6th International Semantic Web Conference (ISWC)*, 2007. 63
- [98] Ricardo Usbeck, Michael Röder, Axel-Cyrille Ngonga-Ngomo, Ciro Baron, Andreas Both, Martin Brümmer, Diego Ceccarelli, Marco Cornolti, Didier Cherix, Bernd Eickmann, Paolo Ferragina, Christiane Lemke, Andrea Moro, Roberto Navigli, Francesco Piccinno, Giuseppe Rizzo, Harald Sack, René Speck, Raphaël Troncy, Jörg Waitelonis, and Lars Wesemann. GERBIL - General Entity Annotation Benchmark Framework. In *24th World Wide Web Conference (WWW)*, 2015. 29
- [99] Graham Vickery. Review of Recent Studies on PSI-use and Related Market Developments. Technical report, EC DG Information Society, 2011. 9, 27
- [100] Wang Richard Y. and Strong Diane M. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 1996. 45
- [101] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality Assessment Methodologies for Linked Open Data. *Semantic Web Journal*, 2012. 46, 49, 50, 51, 52, 53

