Automated Quality Assessment of Metadata across Open Data Portals

SEBASTIAN NEUMAIER, Vienna University of Economics and Business
JÜRGEN UMBRICH, Vienna University of Economics and Business
AXEL POLLERES, Vienna University of Economics and Business

The Open Data movement has become a driver for publicly available data on the Web. More and more data – from governments, public institutions but also from the private sector – is made available online and is mainly published in so called Open Data portals. However, with the increasing number of published resources, there are a number of concerns with regards to the quality of the data sources and the corresponding metadata, which compromise the searchability, discoverability and usability of resources.

In order to get a more complete picture of the severity of these issues, the present work aims at developing a generic metadata quality assessment framework for various Open Data portals: we treat data portals independently from the portal software frameworks by mapping the specific metadata of three widely used portal software frameworks (CKAN, Socrata, OpenDataSoft) to the standardized DCAT metadata schema. We subsequently define several quality metrics, which can be evaluated automatically and in a efficient manner. Finally, we report findings based on monitoring a set of over 260 Open Data portals with 1.1M datasets. This includes the discussion of general quality issues, e.g. the retrievability of data, and the analysis of our specific quality metrics.

CCS Concepts: •**General and reference** → **Measurement; Metrics;** •**Information systems** → *Web searching and information discovery;* Digital libraries and archives;

Additional Key Words and Phrases: Open Data, quality assessment, data quality, data portal

ACM Reference Format:
Sebastian Neumaier, Jürgen Umbrich, and Axel Polleres, 2015. Automated Quality Assessment of Metadata across Open Data Portals. *ACM J. Data Inform. Quality* V, N, Article A (January YYYY), 29 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

As a direct result of the increased momentum within the Open Data movement more data is made available online and the expectation rises that people can use and exploit this data in innovative ways and generate added value out of it. We can identify many areas where Open Data is used and valuable, e.g. by governments to increase transparency and democratic control, or by private companies to encourage innovative use of their data. Having said that, it is impossible to predict how, when and where value can be created in the future: innovations enabled by freely available data can come from any unforeseen place or use case.

With the rising number of available resources, a range of impediments for the Open Data movement have been listed [Zuiderwijk et al. 2012] and (meta-)data quality issues in Open Data portals have been identified as one of the core problems for wider

This work has been supported by the Austrian Research Promotion Agency (FFG) under the project ADEQUATe (grant no. 849982).
Author's addresses: Vienna University of Economics and Business, Building D2, Welthandelsplatz 1, 1020 Vienna, Austria. Department of Information Systems and Operations, Institute for Information Business.

DOI: http://dx.doi.org/10.1145/0000000.0000000

adoption and also as a barrier for the overall success of Open Data. In fact, there have been a number of reports confirming that there exists a quality problem in Open Data [Kucera et al. 2013; Reiche et al. 2014; Umbrich et al. 2015].

Most of the current "open" data form part of a dataset that is published in Open Data portals which are basically catalogues similar to digital libraries (cf. Figure 1): in such catalogues, a *dataset* aggregates a group of data files (referred to as *resources* or distributions) which are available for access or download in one or more formats (e.g., CSV, PDF, Microsoft Excel, etc.). Additionally, a dataset contains *metadata* (i.e., basic descriptive information in structured format) about these resources, e.g. authorship, provenance or licensing information.

For the data to be discoverable by consumers, the publishers need to describe their data in an accurate and comprehensive manner. Missing or incorrect metadata information prevents consumers from finding relevant data for their needs and as a consequence requires a substantial amount of time to (manually) scan the portals and the data itself to locate relevant data sources. Even worse, if a user finds interesting datasets, the data might not be available due to outdated links or might not conform with the format declared in the metadata (e.g., wrong file formats or formats that do not conform with their specifications).

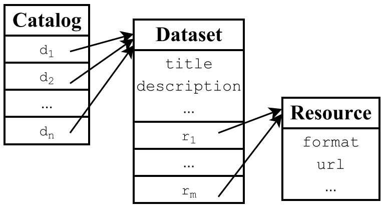


Fig. 1: High-level structure of a Data Catalog.

In order to better understand the severity of such quality issues, we periodically measure and assess the quality of information in Open Data portals for various quality dimensions such as the retrievability of the actual data or the existence of contact or license information. We argue that the insights gained from such a large-scale assessment are not only useful to inform data and portal providers about particular problems, but can also help to identify how and at what stage in the publishing life cycle quality improvement methods need to be applied. For example it may be possible to develop tools to semi-automatically support the creation of data and its metadata, but also algorithms to automatically improve and repair wrong metadata descriptions.

To this end, in this work we present a quality assessment and evolution monitoring framework for web-based data portal platforms, which offer their metadata in different and heterogeneous models. In particular, we make the following contributions:

(i) We provide a *generic formal model* which can be used to represent data and metadata in web portals and discuss general characteristics and quality metrics independently from the portal software frameworks.

(ii) We define a *set of quality metrics* for the DCAT metadata standard, grouped by five different dimensions and present mappings from metadata of three major Open Data portal software providers to our metadata model.

(iii) We introduce our efficient *quality assessment and monitoring framework* that is able to periodically process hundreds of data portals.

(iv) We *report findings* based on monitoring a set of over 261 Open Data portals. This includes the discussion of general quality issues, e.g. the retrievability of resources, and the analysis of the DCAT specific quality metrics (cf. contribution (ii)).

The remainder of this paper is structured as follows: We present preliminaries and background information in Section 2, whereafter we propose a generic model for web-based data portals in Section 3, our contribution (i). Aligned to contribution (ii), we introduce concrete quality metrics based on DCAT in Section 4. We introduce contribution (iii), our QA framework and its implementation, in Section 5, and present and discuss contribution (iv), our concrete findings, in Section 6. We discuss related publications and projects in Section 7 and finally we conclude with Section 8.

2. BACKGROUND

There exist two prominent software frameworks for publishing Open Data: the commercial Socrata Open Data portal and the open source framework CKAN, developed by the Open Knowledge Foundation. While the CKAN software is increasingly popular among cities, governments and private data provider worldwide including government portals of countries in Europe, South and North American and the Middle East, the customers of Socrata can be found mainly in the US. Furthermore, there is the recent data publishing platform OpenDataSoft, deployed mainly in French Open Data catalogs. These three portal frameworks provide ecosystems to describe, publish and consume datasets, i.e., metadata descriptions along with pointers to data resources. Such portal frameworks typically consist of a content management system, some query and search features as well as RESTful APIs to allow agents to interact with the platform and automatically retrieve the metadata and data from this portals. The metadata usually can be retrieved in a structured format via the API (e.g. as JSON data); however, the metadata schemas are heterogeneous wrt. to the underlying software framework.

2.1. Modeling Data Portals

Our proposed model for web-based data portal model (cf. Section 3) is inspired by the Streams, Structures, Spaces, Scenarios, Societies (5S) model [Gonçalves et al. 2004; Fox et al. 2012], which describes the components of digital libraries (e.g., metadata catalogs, collections, browsing and indexing services) through higher level mathematical objects. In detail, we base parts of our work on the “structure” and the “scenario” concepts, which are used within the 5S model to define a *descriptive metadata structure* and a set of *services* for a digital library, respectively.

2.2. Efforts Towards Homogenized Metadata

There exists various standard proposal for metadata vocabularies in data catalogues.

The Data Catalog Vocabulary (DCAT) [Maali and Erickson 2014] is a W3C metadata recommendation for publishing data on the Web. DCAT is defined in RDF and reuses the Dublin Core Metadata vocabulary. While Dublin Core (DC) is a metadata standard that has been specified by the Dublin Core Metadata Initiative [Weibel et al. 1998]. It contains elements for describing resources that are used primarily for cataloging, archiving and indexing of documents (e.g., in archives, libraries).

The recent DCAT application profile for data portals in Europe (DCAT-AP) [1] extends the DCAT core vocabulary and aims towards the integration of datasets from different European data portals. In its current version (v1.1) it extends the existing DCAT schema by a set of additional properties. DCAT-AP allows to specify the version and the period of time of a dataset. Further, it classifies certain predicates as “optional”, “recommended” or “mandatory”. For instance, in DCAT-AP it is mandatory for a `dcat:Distribution` to hold a `dcat:accessURL`. The European Data Portal[2] (launched in November 2015) which currently harvests 68 European data catalogs, supports DCAT-AP metadata.

Recently, in [Assaf et al. 2015] the authors propose HDL, an harmonized dataset model. HDL is mainly based on a set of frequent CKAN keys. On this basis, the authors define mappings from other metadata schemas, including Socrata, DCAT and Schema.org. For instance, HDL maps the Socrata key `description` and DCAT information `dcat:Dataset`→`dct:description` to the CKAN key `notes`. In Section 4 we will base our metadata mapping partially on the work by Assaf et al. [2015].

[1] https://joinup.ec.europa.eu/asset/dcat_application_profile/description, last accessed 2016-02-26

[2] http://www.europeandataportal.eu/, last accessed 2016-02-26

Lastly, in 2011 the W3C published the VoID vocabulary [Alexander et al. 2011] as an Interest Group Note. VoID – the Vocabulary for Interlinked Datasets – is an RDF schema for describing metadata about linked datasets: it has been developed specifically for data in RDF representation and is therefore complementary to the DCAT model and not fully suitable to model metadata on Open Data portals (which usually host resources in various formats) in general.

3. FORMAL MODEL AND METRICS FOR WEB DATA PORTALS

We base our terminology and formalization of web data portals on the 5S model [Gonçalves et al. 2004] in the following way: Out of the extensive body of definitions in the 5S model we use the term and definition of *services* to introduce and define a set of services, which are offered by a portal (e.g., via an API request). Further, we use the graph-based definition of *descriptive metadata structures* of Gonçalves et al. [2004] to formalize metadata descriptions of the available datasets.

3.1. Generic Model for Web Data Portals

Let $\mathcal{P}$ denote a corpus of data portals, where $P \in \mathcal{P}$ is a single data portal, accessible via the URL h_P, which holds a set of dataset URLs $D_P = \{d_1, \ldots, d_n\}$ and a set of services $\mathrm{Serv} = \{\texttt{list}, \texttt{meta}, \texttt{show}, \texttt{resource}\}$:

$$P = (h_P, D_P, \mathrm{Serv}) \tag{1}$$

Such data portals further provide *metadata descriptions* of the listed datasets. A metadata description (see Section 3.1.2 below) is a structured document holding important contextual information about a dataset. In the following, we use the notation $\mathcal{M}$ to denote the set of all available metadata descriptions over $\mathcal{P}$. Note that exactly one metadata description $m \in \mathcal{M}$ is associated with a dataset URL.

In the context of data portals, a *resource* is any target of an URL, which can be hosted internally (i.e., hosted on the same server as the portal) or externally (i.e., a reference to a distant web or file server). Typically we can access resources via links in the metadata descriptions or using the API of the data portal and usually we encounter downloadable files. We denote the set of all resource URLs occurring over the set $\mathcal{P}$ of data portals as $\mathcal{R}$ and the set of all dataset URLs as $\mathcal{D}$ respectively, where $D_P \subseteq \mathcal{D}$ for all $P \in \mathcal{P}$.

3.1.1. Services. In the following, we define the set of services $\mathrm{Serv} = \{\texttt{list}, \texttt{meta}, \texttt{show}, \texttt{resource}\}$. These services are used by our framework to implement the harvesting and quality computation, cf. Section 5. That is, we rely on the availability of the services for the automated computation of our metrics. Next, we describe the services in detail.

list. Let list be a service that returns the set of all dataset URLs for a given Portal, i.e. formally defined as the function $\texttt{list} : \mathcal{P} \to 2^{\mathcal{D}}$, where in particular $\texttt{list}(P) = D_P$ for a portal $P = (h_P, D_P, \mathrm{Serv})$.

meta. Let meta be a service, formalized by a function $\texttt{meta} : \mathcal{P} \times \mathcal{D} \to \mathcal{M}$, that assigns each dataset URL $d \in D_P$ in a portal $P \in \mathcal{P}$ exactly one metadata description $m \in \mathcal{M}$.

show. Let show be a service that provides the set of metadata descriptions for a given data portal P, i.e. $\texttt{show} : \mathcal{P} \to 2^{\mathcal{M}}$ with $\texttt{show}(P) = \{\texttt{meta}(P, d) \mid d \in D_P\}$.

resource. In general, a dataset can describe and reference multiple resources. Therefore, the service resource returns a set of resource URLs for a given dataset URL:

$$\texttt{resource} : \mathcal{P} \times \mathcal{D} \to 2^{\mathcal{R}} \tag{2}$$

Note, that a specific resource URL can be described in various datasets. We can use this service to describe the set of all resource URLs occurring on a data portal P:

$$\bigcup_{d \in D_P} \texttt{resource}(P, d) \subseteq \mathcal{R} \tag{3}$$

Usually these services are directly available as HTTP-based RESTful APIs and therefore are accessed via the portal URL h_P, as in the case of CKAN, Socrata and OpenDataSoft. In case one of the services is not directly available as an API, we can implement the services for that particular portal software. For instance, the `resource` service can be implemented by using the `meta` service and extracting the resource URLs from the returned metadata. This flexibility allows us to integrate future portals which are hosted by other content-management software, e.g, HTML based portals without available APIs.

3.1.2. Metadata Descriptions. We assume metadata is organized as (potentially nested) key-value pairs, where the key contains the label of a property describing the respective dataset, and its value contains the corresponding description or numerical value of this property. On the previously mentioned CKAN portals the metadata description of a dataset is accessible via the an API service (`meta` or `show`). The metadata returned is represented using JSON data and holds references to the actual resources.

We provide a general characterization of a metadata description which is applicable to any occurring concrete metadata instance for data portals. As such, we propose the following tree-based model of a metadata description which is inspired by the graph-based definition of a "descriptive metadata specification" in the 5S model [Gonçalves et al. 2004]:

Let a metadata description $m \in \mathcal{M}$ be a a labelled tree $m = (V, E, \text{label})$ with the dataset URL as its root where each node $v \in V$ and each edge $e \in E$ can hold a label $\text{label}(v)$ (or $\text{label}(e)$ respectively), defined by the labeling function label. If there is no label specified for some node or edge then the function label returns ϵ (denoting an empty string).

The successor nodes of the root can be either internal nodes (i.e., a node with out-degree > 1) or a leaf nodes (also called terminal nodes). In the following, we denote the set of labels over all leaf nodes in a metadata instance m by $\text{leaves}(m)$. A path δ in m is a set of consecutive edges from the root to a leaf node. Let $\text{leaf}(\delta)$ return the single label of the leaf node of the corresponding path.

We note that in principle this generic metadata model covers any tree-based data structures such as XML, JSON and (acyclic) RDF descriptions – also typically represented nowadays in serialization formats such as JSON-LD [Sporny et al. 2014]. The RDF view is labelled intuitively correspond to triples $(n_1, \text{label}(v), n_2)$ for each edge $v \in V$ between nodes n_1, n_2.

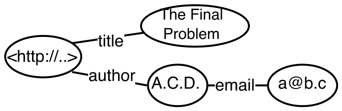
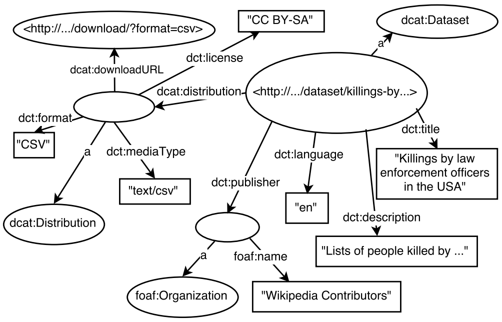


Fig. 2: Metadata description modeled as a tree.

Interpretation. The root $r \in V$ of a metadata instance m represents a dataset d in a portal and is labeled by the dataset URL. The adjacent edges of root r represent attributes and properties of the corresponding dataset. For instance, these edges could be labeled "title" and "author" (cf. Figure 2). The label of an attached leaf node of an edge holds the value of a metadata property and branches at internal nodes describe sub-properties.

```
{
 "datasetid":"killings-by-law-enforcement-...",
 "metas":{
   "publisher":"Wikipedia Contributors",
   "language":"en",
   "license":"CC BY-SA",
   "title":"Killings by law enforcement officers",
   "references":"http://en.wikipedia...",
   "keyword":[
     "killings",
     "law enforcement officers",
     "USA"
   ],
   "description":"Lists of people killed by ..."
 }
}
```



DCAT represented in tree structure.

Fig. 3: Example mapping of an OpenDataSoft metadata description to DCAT.

3.2. DCAT Model Instantiation

The DCAT model [Maali and Erickson 2014] includes four main classes: `dcat:Catalog`, `dcat:CatalogRecord`, `dcat:Dataset` and `dcat:Distribution`. The definition of a `dcat:Catalog` corresponds to the concept of data portals previously introduced in Section 3.1, i.e., it describes a web-based data catalog and holds a collection of datasets (using the `dcat:dataset` property). A `dcat:Dataset` describes a metadata instance which can hold one or more distributions, a publisher, and a set of properties describing the dataset. A `dcat:Distribution` instance provides the actual references to the resource (using `dcat:accessURL` or `dcat:downloadURL`). Further, it contains properties to describe license information (`dct:license`[3]), format (`dct:format`) and media-type (`dct:mediaType`) descriptions and general descriptive information (e.g, `dct:title` and `dcat:byteSize`).

In the following, we will apply the RDF structure of a `dcat:Dataset` together with its distributions and properties to the tree-shaped concept of a metadata description introduced in Section 3.1. We label the root node of the metadata description with the `dct:Dataset` subject (i.e., the dataset URL) and add an edge for each of the properties, linked with a node for the corresponding objects and values, respectively. For instance, the leaf of the path (`dcat:dataset`, `dct:publisher`, `foaf:homepage`) is used to describe provenance information in DCAT. Figure 3 (right) displays the tree structure of a DCAT metadata description.

3.3. Metrics over General Data Portal Model

Initially, we define the set of all possible *metadata properties* within a metadata description. Clearly, a tree-shaped metadata description consists of a set of *paths* from the root node to the leaves. The sequence of edge labels of these paths describe metadata properties and the corresponding leaves hold the values of these paths. For instance, the path labelled $\langle author, email\rangle$ in Figure 2 describes the "email address of an author".

In the following definitions, let Δ_m be the set of all paths from the root of a single metadata instance m to the leaves. We use δ to denote a single path in a metadata description and write $\mathrm{label}(\delta)$ for the sequence of labels on the path. Note that necessarily $|\Delta_m| = |\,\mathrm{leaves}(m)|$ holds.

[3] `dct:` denotes the Dublin Core Metadata namespace.

3.3.1. Path Selector Function. Let $s_K(m)$ be a path selector function over a metadata description m which we assume for simplicity to be defined by a set of keys K, i.e., $s_K(m) = \{\delta \mid \delta \in \Delta_m \text{ s.t. } K \cap \mathrm{label}(\delta) \neq \emptyset\}$. if we apply a path selector function with the $K = \{\texttt{dct:distribution}\}$ to the tree-structured DCAT metadata in Figure 3 (right figure) this will return 5 paths of this DCAT metadata description, including for example a path δ_1 with $\mathrm{label}(\delta_1) = \langle \texttt{dcat:distribution}, \texttt{dct:format} \rangle$ with $leaf(\delta_1) =$"CSV".

3.3.2. Boolean Evaluation Functions over a Path. Let $f(\delta)$ be a boolean function over the leaf of a specific path δ which returns either 0 or 1. For instance $f(\delta)$, we will use the boolean function $\mathit{nonEmpty}(\delta) = (\mathrm{leaf}(\delta) \neq \epsilon)$ to determine if the labelling of some leaf of a path is non-empty.

Another example would be the function $\mathit{isValidEmail}(\cdot)$ which is used to evaluate a regular expression on the value of the leaf of a given path. Further, we can use such a boolean function for evaluating user-defined functions, e.g. a function $\mathit{isOpenFormat}(\cdot)$ (cf. details below in section 4.2.2) which returns 1 if the specified value of $\mathrm{leaf}(\delta)$ is contained in a predefined set of labels corresponding to open format descriptions. For instance, for the path δ_1 from above $\mathit{isOpenFormat}(\delta) = 1$ if we assume the value "CSV" among the set of open file formats.

3.3.3. Aggregation of Labels. Finally, for our metrics we will use an aggregation function $\mathit{agg} \in \{\mathit{min}, \mathit{max}, \mathit{avg}\}$ to specify how to aggregate $f(\delta)$ for all paths $\delta \in s_K(m)$, to denote minimum, maximum and average. For the special case that $s_K = \emptyset$ (i.e., none of the paths in the metadata description is matching the specified selector) we assume that agg returns 0 for any specific aggregation function, i.e., overall the aggregation always returns a value between 0 and 1.

3.3.4. Quality Metrics over Paths. We now define a basic quality metric over a metadata description m as:

$$\mathsf{Metric}(s_K, f, \mathit{agg})(m) = \mathit{agg}(\{f(\delta) \mid \delta \in s_K(m)\}) \tag{4}$$

For instance, we will use a OpenFormat quality metric defined as follows:

$$\mathsf{OpenFormat} = \mathsf{Metric}(s_{\{\texttt{dct:format},\texttt{dct:mediaType}\}}, \mathit{isOpenFormat}, \mathit{avg}) \tag{5}$$

3.3.5. Combined Quality Metrics. We can also combine several basic metrics again by aggregation. For instance, to calculate the average existence of discovery information in a DCAT metadata description, we use the following *combined metrics*:

$$\mathsf{Discovery} = \mathit{avg}\left(\begin{array}{l} \{\ \mathsf{Metric}(s_{\{\texttt{dct:title}\}}, \mathit{nonEmpty}, \mathit{max}), \\ \quad \mathsf{Metric}(s_{\{\texttt{dct:description}\}}, \mathit{nonEmpty}, \mathit{max}), \\ \quad \mathsf{Metric}(s_{\{\texttt{dct:keyword}\}}, \mathit{nonEmpty}, \mathit{max})\} \end{array}\right) \tag{6}$$

Here, we calculate the average existence over results of different DCAT path selector functions. When applying this Discovery metric to the example in Figure 3, we observe a total value of 2/3: the first and second metrics (title and description) returns 1 since they exist and are non-empty (i.e., $\max$ aggregation yields 1), while the third metric returns 0 since there is no path with the `dct:keyword` property in the figure.

4. METADATA MAPPING AND QUALITY DIMENSIONS

When investigating the metadata structure of common data publishing frameworks and portals (such as CKAN and Socrata) one observes different metadata schemas and heterogeneity issues. For instance, the Socrata framework describes licensing information by the single metadata key `license`, while in CKAN there are three different keys for specifying the ID, the URL and the name of a license.

This observation highlights the need for a common schema applicable to a range of metadata sources that can be used in order to improve the comparability and simplify the integration of data from different portals. This allows to compute our quality metrics for web data portals independently from their corresponding publishing software and metadata format.

As a first step towards a generalized metadata schema, we propose a manual mapping for metadata schemas observed on CKAN, Socrata and OpenDataSoft portals to the DCAT metadata standard. The proposed mapping is intended as a homogenization of different metadata sources by using the W3C's DCAT vocabulary [Maali and Erickson 2014]. Our decision in favour of DCAT was influenced by the increasing momentum in terms of integration and implementations of DCAT in existing Open Data systems: CKAN has a plugin for exporting DCAT, Socrata can export DCAT per default and OpenDataSoft is using overlapping metadata key names to DCAT by design. That is, DCAT serves as a least common denominator for describing datasets in various formats and therefore allows us to homogenise metadata retrieved from different publishing frameworks.

4.1. DCAT Mapping

In Table I we introduce our mapping for the different metadata keywords, which is partially derived from the dataset harmonization framework proposed by [Assaf et al. 2015]. The mapping includes metadata keys from Socrata, CKAN and OpenDataSoft mapped to `dcat:/dct:` (Dublin Core Metadata) properties. The bold headers in the table indicate a class (i.e. an RDF subject) within the DCAT model; the part after → represents an RDF property. Blank fields within the table indicate that we were not able to match a corresponding key with the same semantic meaning. Please note, that individual datasets may contain a suitable key, but that we only map default, regularly occurring metadata keys.

For instance, `dcat:Dataset`→`dct:title` denotes an RDF triple (*`dataset`*, `dct:title`, *`title`*) in the resulting mapping, where *`dataset`* is a `dcat:Dataset` and *`title`* is the corresponding mapped value (i.e., a RDF literal holding the value of the mapped metadata key).

The proposed mapping of the keys is mainly based on matching names. For instance, considering the mapping of the OpenDataSoft metadata keys, we can see that all mapped keys use the same key-names as the DCAT vocabulary. If the key-names are not matching (as for most of the CKAN keys), we mainly rely on existing mappings, further explained in Section 4.1.1.

Figure 3 displays an application of the proposed DCAT mapping for an OpenDataSoft metadata description. The DCAT mapping is presented as a tree, where oval nodes represent RDF resources and square nodes represent literals. Note that the `dct:license` in the DCAT model belongs to a distribution, while in the original metadata it is attached to a dataset instance. (This holds likewise for the license keys in Socrata and CKAN portals.)

4.1.1. Adapting existing Mappings. In order to make use of the proposed homogenization within our QA framework (Section 5) we implemented a mapping algorithm for each of the data management systems covered by Table I.

Regarding the CKAN software we took a closer look at the source code of the DCAT extension for CKAN,[4] currently being developed by the Open Knowledge Foundation. We used the existing mapping of datasets mostly "as is", except for the licenses infor-

[4] https://github.com/ckan/ckanext-dcat, last accessed 2015-11-03. We currently use the code committed on August 13, 2015.

Table I: DCAT mapping of different metadata keys.

DCAT	CKAN	Socrata	OpenDataSoft
`dcat:Dataset`			
→ `dct:title`	title	name	title
→ `dct:description`	notes	description	description
→ `dct:issued`	metadata_created	createdAt	-
→ `dct:modified`	metadata_modified	viewLastModified	modified
→ `dct:identifier`	id	id	datasetid
→ `dcat:keyword`	tags	tags	keyword
→ `dct:language`	language	-	language
→ `dct:publisher`	organization	owner	publisher
→ `dct:contactPoint`	maintainer, author (-email)	tableAuthor	-
→ `dct:accrualPeriodicity`	frequency	-	-
→ `dct:landingPage`	url	-	-
→ `dct:theme`	-	category	theme
`dcat:Distribution`			
→ `dct:title`	resources.name	-	-
→ `dct:issued`	resources.created	-	-
→ `dct:modified`	resources.last_modified	-	-
→ `dct:license`	license_{id, title, url}	licenseId	license
→ `dcat:accessURL`	resources.url	*export URL*[a]	*export URL*[a]
→ `dcat:downloadURL`	resources.download_url	-	-
→ `dct:format`	resources.format	*export format*[a]	*export format*[a]
→ `dct:mediaType`	resources.mimetype	*export mime-type*[a]	*export mime-type*[a]
→ `dct:byteSize`	resources.size	-	-

[a] Socrata and OpenDataSoft offer data export in various formats via the API

mation which is currently not mapped properly: the original mapping in the extension assumes a license key for each resource in a dataset which does not exist in CKAN datasets.

For Socrata portals, we mainly rely on the pre-existing DCAT output. Additionally, we modify the result so that it conforms to the standardized DCAT model. This means, firstly, we replace non-DCAT with standardized DCAT properties in the result if they are synonymous and secondly, we add provenance and authorship information if it is available in the default metadata.

Regarding the homogenization of OpenDataSoft portals we map the values of the metadata keys as described in Table I.

4.2. Concrete DCAT Quality Dimensions

Commonly, Data quality is described as "the fitness for use of information" [Bizer and Cyganiak 2009] and is typically a multidimensional construct. The selection of a proper set of quality dimensions is highly context-specific since their purpose is testing the fitness for use of data for a specific task.

As such, we propose a set of quality dimensions and metrics in the context of the available metadata keys in the DCAT specification. An overview of our quality dimensions and their metrics are listed in Table II. We group our metrics into the following five quality dimensions: Existence, Conformance, Retrievability, Accuracy and Open Data fitness of information.

Our definition of the Existence dimensions is inspired by other commonly used "completeness" metric [Pipino et al. 2002; Bizer and Cyganiak 2009]. However, our

Table II: Quality Dimensions on DCAT keys.

Metric		dcat:Dataset	dcat:Distribution
EXISTENCE *Existence of important information (i.e. exist certain metadata keys)*			
Access*	Is there access information for resources provided?		dcat:accessURL dcat:downloadURL
Discovery	Is information available that can help to discover/search datasets?	dct:title dct:description dcat:keyword	
Contact*	Existence of information that would allow to contact the dataset provider.	dcat:contactPoint dct:publisher	
Rights	Existence of information about the license of the dataset or resource.		dct:license
Preservation	Existence of information about format, size or update frequency of the resources	dct:accrualPeriod.	dct:format dcat:mediaType dcat:byteSize
Date	Existence of information about creation and modification date of metadata and resources respectively.	dct:issued dcat:modified	dct:issued dcat:modified
CONFORMANCE *Does information adhere to a certain format if it exist?*			
AccessURL*	Are the values of access properties valid HTTP URLs?		dcat:accessURL dcat:downloadURL
ContactEmail*	Are the values of contact properties valid emails?	dcat:contactPoint dct:publisher	
ContactURL*	Are the values of contact properties valid HTTP URLs?	dcat:contactPoint dct:publisher	
DateFormat	Is date information specified in a valid date format?	dct:issued dcat:modified	dct:issued dcat:modified
License	Can the license be mapped to the list of licenses reviewed by opendefinition.org?		dct:license
FileFormat	Is the specified file format or media type registered by IANA?		dct:format dcat:mediaType
RETRIEVABILITY *Availability and retrievability of the metadata and data*			
Retrievable	Can the described resources be retrieved by an agent?		dcat:accessURL dcat:downloadURL
ACCURACY *Does information accurately describe the underlying resources?*			
FormatAccr	Is the specified file format accurate?		dct:format dcat:mediaType
SizeAccr	Is the specified file size accurate?		dcat:byteSize
OPEN DATA *Is the specified format and license information suitable to classify a dataset as open?*			
OpenFormat	Is the file format based on an open standard?		dct:format dcat:mediaType
MachineRead	Can the file format be considered as machine readable?		dct:format
OpenLicense	Is the used license conform to the open definition?		dct:license

existence metric differs in the sense that it gives an indication to what extent a certain set of DCAT keys are used (i.e., can be mapped) and contain information, while in other works the completeness is typically defined as the extent to which data is not missing [Pipino et al. 2002]. The existence dimensions (analogous to completeness) can be categorised as contextual [Zaveri et al. 2015] or context-based dimensions [Bizer and Cyganiak 2009], i.e., as dimensions that "highly depend on the context of the task at hand" [Zaveri et al. 2015].

The CONFORMANCE dimension is inspired by the "representational-consistency" dimension which is defined as "the degree to which the format and structure of the information conform to previously returned information" [Zaveri et al. 2015]. However, our conformance dimension differs from consistency in terms of not comparing values to previously returned information, but by checking the conformance of values wrt. a given schema or standard. For instance, the Contact metric is a measure for the existence of contact information, while the ContactEmail metric is a conformance measure which checks if the available contact information is a valid email address.

Our RETRIEVABILITY and ACCURACY dimensions are aligned with existing ones: see accessibility in [Pipino et al. 2002; Umbrich et al. 2015] or availability in [Bizer and Cyganiak 2009] for retrievability, and [Zaveri et al. 2015; Reiche et al. 2014] for accuracy. The accuracy dimensions – FormatAccr and SizeAccr – refer to the compliance of the actual content of the underlying resources with the metadata. In order to accurately assess these dimensions we need to inspect the actual content of the resource. In [Zaveri et al. 2015] the accuracy is therefore considered as an intrinsic quality dimension, i.e., it assesses if information correctly represents the real world.

The OPEN DATA dimension is based on the Open (Knowledge) Definition.[5] It defines "open data" as an item or piece of knowledge which satisfies the following three requirements: (i) freely accessible as a whole, (ii) provided in a machine-readable and open format, and (iii) openly licensed. While (i) is already covered by the RETRIEVABILITY dimension, we introduce the OpenFormat, MachineRead and OpenLicense metric to cover the requirements (ii) and (iii).

4.2.1. Quality dimensions not yet considered. Besides the introduced dimensions, there are further quality dimensions which are relevant in the context of metadata quality in data portals but are not yet considered in our framework since they cannot be computed in an automatic and objective way. *Timeliness* [Pipino et al. 2002] is a measure of how sufficiently up-to-date a dataset is for a certain task (e.g., live timetables or current weather data). However, it is hard to automatically understand the time dimension from the metadata description of a dataset, e.g., to distinguish between static data vs. real-time data. Therefore, we do not yet consider it in our current framework which focuses on automatic completion via periodic (weekly) crawls.

Another commonly applied quality metric is the *information richness*/*uniqueness* [Reiche et al. 2014] of the metadata description, typically measured by how much unique information is provided compared to all other datasets. The problem with this measure for an automatic assessment is in general that portal owners might want to achieve a low uniqueness value for certain metadata keys while having a high value for others. For example, if a portal owner wants that all datasets are published under the same license, the information richness value for license terms would be ideally very low. However, title descriptions of datasets should be as descriptive and distinguishable as possible and as such the quality metric shows ideally a high value. As such, it is hard to establish a general information richness value and it is more likely

[5] http://opendefinition.org/, last accessed 2015-10-30

that one would need to manually adapt this value for specific keys and for the goals of a portal provider. These observations need to be carefully considered for an overall information richness value.

4.2.2. Automated Assessment of Quality Metrics. To calculate the proposed metrics, we use the formulae introduced in Section 3.3. In general, the metrics are assessed by calculating the average (i.e. by using the aggregation function avg) over the set of corresponding DCAT properties. The star (*) besides a metric in Table II indicates that we use the *max* function to aggregate the values, which basically means that one positive evaluation is sufficient.[6]

EXISTENCE. To calculate the existence value for a specific quality metric we use the boolean evaluation function $nonEmpty$ from Section 3. The Access* and Contact* quality metric in Table II are defined by using the max aggregation function, e.g.:

$$\text{Access}^* = \text{Metric}(s_{\{\texttt{dcat:accessURL,dcat:downloadURL}\}}, nonEmpty, max) \tag{7}$$

The other existence metrics are defined using the avg aggregation. Discovery is calculated using the combined metrics as already introduced in Section 3.3.5 and the Rights metric is defined using a single DCAT property:

$$\text{Rights} = \text{Metric}(s_{\{\texttt{dct:license}\}}, nonEmpty, avg) \tag{8}$$

CONFORMANCE. The conformance metrics are assessed by using boolean evaluation functions which are either implemented using regular expressions or by specific functions for mapping licenses and file formats.

— *Using Regular Expressions.* In our conformance evaluation we use regular expressions to validate URLs, email addresses, and date formats of the AccessURL*, ContactEmail*, ContactURL* and DateFormat metrics, respectively. For instance, we calculate the AccessURL* metric of the dataset in Figure 3 by applying a regular expression for URLs to the value of the `dcat:downloadURL` property:

$$\text{AccessURL}^* = \text{Metric}(s_{\{\texttt{dcat:accessURL,dcat:downloadURL}\}}, isValidUrl, max) \tag{9}$$

For the DCAT metdata in Figure 3 the metric evaluates to 1 since `dcat:downloadURL` describes a valid URL.

— *License Conformance.* To validate the metadata description of a given license information we use a list of licenses provided by the Open Definition.[7] This list contains details about 109 different licenses including their typical ID, URL, title and an assessment if they are considered as ”open“ or not. The license information of a dataset in CKAN can be described with three different CKAN keys, namely `license_id`, `license_title` and `license_url`. In Socrata and OpenDataSoft there is only one `license` key which describes the license.

In our framework we implemented the user-defined function as a license matching heuristic which tries to match a dataset license to one of the licenses in the predefined list by performing the following steps. Firstly, we try to perform the match using the `license_id` value, if available. If this check fails we use next the `license_title`, which is matched either against the ID or title in the Open Definition license list. We perform this additional ID match because we observed that in several cases the datasets contain the license ID in the license title field. If this check also fails, we use as a fall back solution the `license_url` value for the match. Once a match was successful we consider a license as compliant.

[6] We introduce this exception because for certain keys (e.g. the `dcat:accessURL` and `dcat:downloadURL`) the existence/availability of a value for one of these keys already provides the desired information.

[7] http://licenses.opendefinition.org/licenses/groups/all.json, last accessed 2015-11-02

— *File Format Conformance.* Regarding the conformance of file formats (FileFormat) we check if the normalized description (i.e., we remove leading dots and use lower case strings) is a format or a media type registered by the Internet Assigned Numbers Authority.[8]

OPEN DATA. The assessment of openness and machine readability of licenses and file formats is based on specific boolean functions (cf. Section 3.3).

— *Format Openness.* Regarding the OpenFormat we use a $\mathit{isOpenFormat}(\cdot)$ function which checks for containment in a predefined set of confirmed open formats:

$$ascii, audio/mpeg, bmp, cdf, csv, csv.zip, dbf, dvi, geojson, geotiff, gzip, html, iati, ical, ics, jpeg2000,$$
$$json, kml, kmz, mpeg, netcdf, nt, ods, pdf, pdf/a, png, psv, psv.zip, rdf, rdfa, rss, rtf, sparql,$$
$$sparql\ web\ form, svg, tar, tiff, tsv, ttl, txt, wms, xml, xml.zip, zip$$

The formula used for calculating the format openness is already introduced in Section 3.3.5.

— *Machine-Readability of Formats.* Likewise, we defined a set of machine-readable file formats for the MachineRead metric:

$$cdf, csv, csv.zip, esri\ shapefile, geojson, iati, ical, ics, json, kml, kmz, netcdf, nt, ods, psv, psv.zip, rdf, rdfa,$$
$$rss, shapefile, shp, shp.zip, sparql, sparql\ web\ form, tsv, ttl, wms, xlb, xls, xls.zip, xlsx, xml, xml.zip$$

The aforementioned collection of open and machine-readable formats are mainly based on a manual evaluation of file formats by the OpenDataMonitor project.[9]

— *Open Data fitness of Licenses.* We confirm the openness of the license (OpenLicense metric) per dataset by evaluating how the specified license is assessed in the list of licenses provided by the Open Definition (same list as in license conformance above). We decide on the openness of a license based on the above introduced license mapping heuristic, i.e., we use a boolean function *isOpenFormat* which returns 1 if we can map a license to the Open Definition list and the license is suitable to publish Open Data (according to this list).
Please note the case that our metric reports only on the confirmed licenses. It might be that the non-confirmed licenses are also adhering to the Open Definition.

RETRIEVABILITY. We calculate the RETRIEVABILITY dimension by defining the boolean function $\mathtt{retr}$ using the HTTP status code of a GET request:[10]

$$\mathtt{retr}(x) = \begin{cases} 1 & \text{if GET}(x) = 2xx \\ 0 & \text{else} \end{cases} \tag{10}$$

Based on this boolean function we define the Retrievable dimension as follows:

$$\text{Retrievable} = \text{Metric}(s_{\{\texttt{dcat:accessURL},\texttt{dcat:downloadURL}\}}, \mathtt{retr}, \mathit{max}) \tag{11}$$

ACCURACY. The accuracy dimension reflects the degree of how accurately the available metadata values describe the actual data. In Table II we introduced two accuracy metrics for DCAT metadata keys: FormatAccr and SizeAccr. Most commonly, one defines different distance functions for the relevant metadata keys, e.g. a function which compares and calculates the distance between the value of the `dcat:byteSize` key and the actual size of the resource.

[8] http://www.iana.org/assignments/media-types/media-types.xhtml, last accessed 2015-11-02

[9] https://github.com/opendatamonitor/odm.restapi/blob/master/odmapi/def_formatLists.py, last accessed 2015-11-16

[10] Note that we automatically follow redirects (i.e. 3xx status codes) and mean here the HTTP return code after such redirects.

A possible indicator for the size of a resource is the `content-length` field in the HTTP response header. However, we observed a considerable number (22%) of resources with missing `content-length` information. Also, if available, this information could also refer to the compressed version and not the actual file size. Therefore, the reliable calculation of the SizeAccr metric requires the download and inspection of all referenced resources. Likewise for the file format we have observed in our experiments that even if the `content-type` header is available it is partially inconsistent with the real file formats (e.g., misconfigured web servers). As such, it is necessary to download and inspect the content to determine the real content-length, encoding and file format of a resource. However, and in order to keep our framework scalable (without the need to download all resources) we currently exclude these accuracy measures in our evaluation.

5. OPEN DATA PORTAL WATCH FRAMEWORK

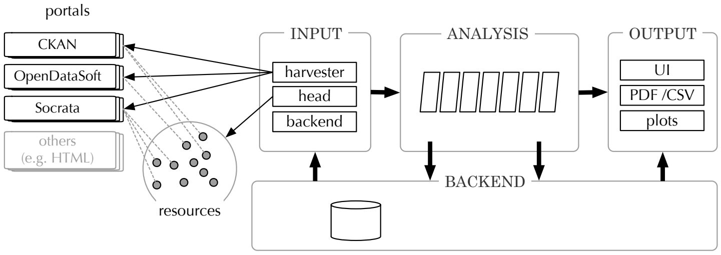


Fig. 4: Open Data Portal Watch architecture

The overall architecture of “Open Data Portal Watch”, our quality assessment and evolution monitoring framework for Open Data (Web) portals, is shown in Figure 4 and comprises of four main building blocks, where each block contains a set of components:

(INPUT) The INPUT block consists of several components to provide various iterators for data items that are processed and/or analyzed.

(ANALYSIS) The data items provided by the input block are then piped through the ANALYSIS block which consists of a set of processing and analyzer components that can be chained into a processing pipeline.

(BACKEND) Both the input and analysis blocks interact with the BACKEND unit in order to store or retrieve raw data or results.

(OUTPUT) The components in the OUTPUT block interact with the backend and analysis blocks and can be used to provide results and information in various formats (e.g., as CSV files or as JSON data for the UI).

5.1. Architecture

In the following, we discuss the components of each block in more detail.

5.1.1. INPUT. We implemented three different modules to access and retrieve data:

(1) *Harvester.* The first component is called the *harvester*. It accesses the online data portals and retrieves all metadata about the datasets. Our framework currently provides three different harvester modules to invoke the specific service functions for the differently portal software (CKAN, Socrata, OpenDataSoft). The challenge we faced is that the service function for the same portal software might react very

different across the portals or are not activated for every portal. Also temporary network or server errors can occur and need to be taken care of.

In our harvesting component, initially, we invoke the `show` service of the portal to directly download the metadata descriptions of the datasets. Ideally this requires only one HTTP GET operation. However, we observed in practice and for the CKAN portals that it is more stable to combine the `show` function with pagination (i.e., not retrieving all metadata descriptions of a portal at once) which results in more requests but less data to generate on the server and to transfer for each request. It turned out, that pagination is extremely beneficial for larger portals with more than 1000 datasets.

Since we encountered server timeouts for some portals using the `show` service, we additionally make use of the `list` service of the portals: we retrieve the list of all dataset URIs and request the metadata description for each single dataset URI (using the `meta` service of the portal). Note, that this single processing highly influences the runtime of our harvesting algorithm. In order to avoid getting locked out by the server due to an excessive amount of HTTP requests, we wait for a short amount of time before executing the next query on a specific portal (cf. web crawling politeness [Najork and Heydon 2002]). It is worth noting that using our implementation it is possible to process multiple portals in parallel.

(2) *Head.* The second component performs HTTP HEAD requests on the resource URLs described in the datasets of the portals to check their availability and to gather more information The list of all unique resource URLs is extracted and stored in the database during the analysis of the harvested datasets. The header information which is retrieved is stored in the backend and analysed in the ANALYSIS block.

(3) *Backend.* The third component of the INPUT module is used to supply the analysis block with data from the database instead of data from the portals and resources, respectively. For instance, this component can be used to recalculated quality dimensions for already stored datasets.

5.1.2. BACKEND. Our backend system is a Postgres (Version 9.4) database instance which makes use of the native JSON type feature to store schema-less information for datasets, resources and the portal metadata. For instance, we store the header information from the HTTP HEAD lookups in the resources table as JSON. In total, we have four main tables:

- One table to store basic information about each portal, such as the URL, API URL and the software).
- One table to store basic properties (e.g., number of datasets and resources) and the aggregated quality metrics for each portal and snapshot.
- One table to store the harvested information about each datasets for each portal and snapshot.
- One table to store all unique resources and the information from the HEAD lookups and the datasets and portals they are described in.

We further partition the dataset and resource table by snapshot for performance reasons.

5.1.3. ANALYSIS. The components in our analysis block can be grouped into three categories: First, a set of components to calculate basic statistical information about the occurrence and distribution of various metrics, such as the number of datasets, resources, response code distribution, frequency count for licenses, formats, organisations, etc. Secondly, a set of quality assessment components, including our DCAT mapping, which calculates the previously introduced DCAT quality dimensions. Eventually, we implemented a set of components that interact with the backend in order to

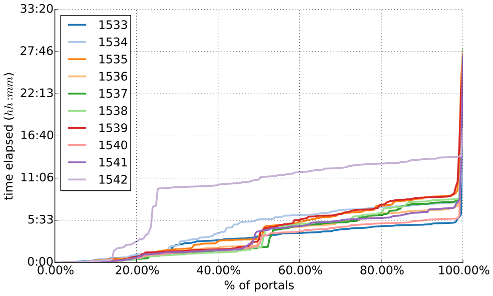


Fig. 5: Elapsed time for harvesting process for the last 11 snapshots.

store the raw harvested data, the resource headers and the results from the quality assessment analyzers.

We pipe the retrieved datasets directly through our analysis block to calculate all measurements “on the fly”. Since the portals can be treated independently, we process them in parallel. The retrieved datasets for each portal and snapshot are in addition stored/archieved in our backend system. This allows us on the one hand to share the collected portal snapshots with other researchers and on the other hand to re-compute metrics, or compute possible new quality metrics for already collected snapshots. In addition, the archived snapshots can be further exploited to analyse changes and modifications to the metadata which we plan to address in future work.

5.2. Server Error Handling

We implemented several strategies to cater for and prevent possible data loss caused by “server errors” during the harvesting process for a portal. If a portal is unavailable we restart the metadata harvesting process at a later stage.

Further, we occasionally observed server or timeout errors while invoking the `show` service due temporary server overload which might be caused by fetching potentially large sets of metadata descriptions. In that case, we re-invoke the `show` service with decreasing pagination size and increasing the wait time.

In order to trace possible server errors (but also bugs in our code) we store the debug and error logs for each harvested snapshot.

5.3. Data & Efficiency evaluation

One of the main requirements of our framework is to be able to periodically monitor the portals which depends on the time elapsed to harvest the metadata of all portals in the system. At the time of writing, we actively monitor 261 portals once a week. Please note, that the monitoring and harvesting process is influenced by external factors which cannot be assured to scale in all possible cases. For instance, if a data portal does not support the download of multiple metadata descriptions via pagination (cf. Section 5.1.1), we have no other alternative than to send a request for each single description (potentially even in a patient way, additionally respecting typical politeness delays between requests[Harth et al. 2006]).

Figure 5 plots the time elapsed to fetch all portals for our last 11 snapshots. The snapshot number in the legend denotes the year and week of the harvesting process; e.g. 1533 is week 33 of the year 2015. The plot shows that our framework fetches the vast majority (>95%) of the portals in 10 to 12 hours and fetches the remaining individual portals in a total of about 27 hours.

Table III: Number of portals and processing time per snapshot

snapshot	1533	1534	1535	1536	1537	1538	1539	1540	1541	1542
$\mid \mathcal{P} \mid$	239	239	239	239	239	239	239	240	256	256
not available	8	10	11	8	8	9	8	8	13	13
fetched	231	229	227	231	229	228	231	230	243	243
fetch aborted	0	0	1	0	2	2	0	2	0	0
time (hh:mm)	27:29	28:03	27:48	26:41	27:35	28:05	26:01	27:33	27:09	17:33

Table IV: Top-5 and bottom-5 portals, ordered by datasets.

domain of portal URL	Origin	Software	$\mid\mathcal{D}\mid$	$\mid\mathcal{R}\mid$
www.data.gc.ca	Canada	CKAN	244948	1163911
data.gov	US	CKAN	162351	763049
ckan.gsi.go.jp	Japan	CKAN	147955	147953
data.noaa.gov	US	CKAN	65915	475330
geothermaldata.org	US	CKAN	56391	62136
data.salzburgerland.com	Austria	CKAN	6	34
www.criminalytics.org	US	Socrata	6	-
bistrotdepays.opendatasoft.com	France	OpenDataSoft	4	-
www.opendatanyc.com	US	Socrata	2	-
ckanau.org	Ecuador	CKAN	1	2

In addition, Table III lists for each snapshot the total number of portals in our system ($\mid \mathcal{P} \mid$) and for how many of these portals we could successfully harvest all dataset descriptions.

As we can see, we had to terminate the fetch process for a maximum of 2 portals for the snapshots 1537, 1538, and 1540. In fact, the two responsible portals are huge CKAN portals for which we had to harvest the datasets one by one using the `meta` service since the `show` service was temporarily not available. Please also note that we have currently between 8 to 13 portals in the system for which we could not start the harvesting process, either because the respective portals were offline or returned API errors at the time of access.

6. QUALITY EVALUATION OVER A CORPUS OF OPEN DATA PORTALS

In this section, we present the findings of our quality assessment for 261 Open Data portals for the snapshot of the fourth week of February 2016.

6.1. Overview of portals

Currently our system holds in total 261 portals, of which are 148 using the CKAN software, 102 the Socrata software and 11 are powered by OpenDataSoft. The full list of all current portals is available on the web-interface of our framework.[11] In total, the 261 portals attribute to 1.1M datasets which describe 2.1M unique resources. Table IV lists the top and bottom 5 portals with respect to the number of datasets. It is worth noting that 4 out of the top-5 portals are based in North America.

We collected the list of portals from various sources. One source is the list of customers on the homepage of the portal software providers (e.g. Socrata[12], OpenDataSoft[13] and CKAN[14]). Another source of portal URLs stems from the dataportals.org

[11] http://data.wu.ac.at/portalwatch/portals

[12] https://www.opendatasoft.com/company/customers/, last accessed 2015-10-14

[13] https://opendata.socrata.com/dataset/Socrata-Customer-Spotlights/6wk3-4ija, accessed 2015-10-14

[14] http://ckan.org/instances/#, accessed 2015-10-14

Table V: Distribution of number of datasets over all portals.

$\lvert D\rvert$	<50	$<10^2$	$<5\times10^2$	$<10^3$	$<5\times10^3$	$<10^4$	$<5\times10^4$	$<10^5$	$>10^5$
$\lvert\mathcal{P}\rvert$	73	21	75	30	36	11	9	3	3

service which lists in total 431 Open Data publishing sites, out of which 125 are CKAN portals. Further, the OpenDataMonitor project also provides a list of 217 portals, including 52 CKAN portals.[15]

Table V lists the distribution of portals regarding their number of datasets. The table cells in Table V should be interpreted as intervals: for instance, in the 3rd column we can see that 75 portals hold between 100 and 500 datasets.

One can observe that the majority of 65% of the portals contains less than 500 datasets. The largest two portals are Canada's Open Government data catalog (`open.canada.ca`) consisting of 245k datasets followed by the official U.S. government data portal data portal `data.gov`.

6.2. Retrievability

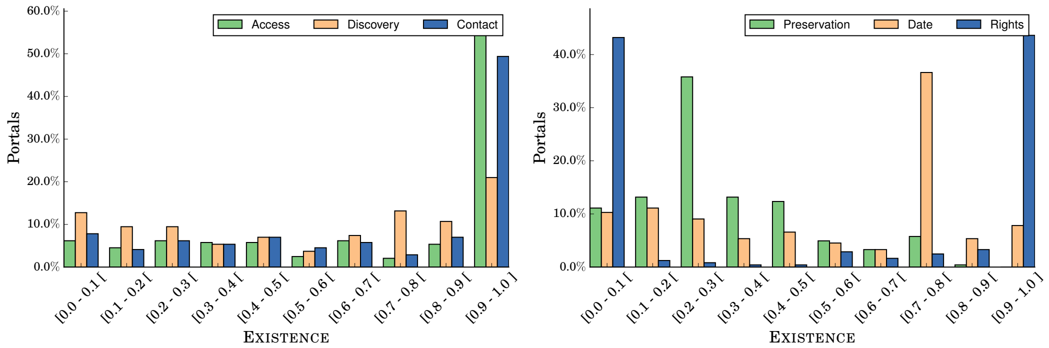
The results of our dataset and resource retrievability analysis are summarized in Table VI. We grouped the response codes by their first digit; *others* indicate socket or connection timeouts. As expected, nearly all datasets could be retrieved without any errors (98%). The 8026 datasets that could not be retrieved responded with a 403 FORBIDDEN HTTP status code, indicating that an access token is required to retrieve the information.

A slightly different picture can be observed regarding the retrievability of the content of the actual resources. Out of a total of 2.6M resource values (i.e., values of the `dcat:accessURL` or `dcat:downloadURL` properties) appearing in 1.1M dataset descriptions, 2.1M are unique distinct values. We performed lookups on the valid URLs among these, resulting in the response code distribution in Table VI. Around 78% of these resources are accessible returning in a response code of 2xx. An slightly alarming observation is that 308k described resources (∼15%) returned a response code of 4xx, indicating that the resources is not available. A closer inspection of these results revealed that 176k resource URLs, hosted on Socrata portals, return a 400 code with the error message " *HEAD is not supported*". 14k resources (∼7%) caused some socket or timeout exception upon the lookup (indicated with others). In general, the number of exceptions should interpreted with caution since the unavailability of the content of a URL might be temporary due to internal server errors or network problems. In future work we plan to distinguish between persistent and temporary errors by considering the evolution of the URL's retrievability.

Table VI: Distribution of response codes.

		2xx	4xx	5xx	others
$\lvert\mathcal{D}\rvert$	1146435	1138246	8026	0	163
$\lvert\mathcal{R}\rvert$	2102778	1641098	308531	14410	138739

[15] http://project.opendatamonitor.eu/, last accessed 2015-11-14



Access, Discovery, and Contact information.

Preservation, Date, and Rights.

Fig. 6: EXISTENCE Histograms.

6.3. Existence

Next, we discuss the results of the metrics for the existence quality dimension which are displayed in the histograms in Figure 6. In general, the metrics are rather equally distributed. A slightly concerning result is the existence of access information: only 50% of the portals have an Contact value over 0.9. For instance, the missing information does not allow data consumers to contact the publishers, e.g., to get more information about the data or to report errors.

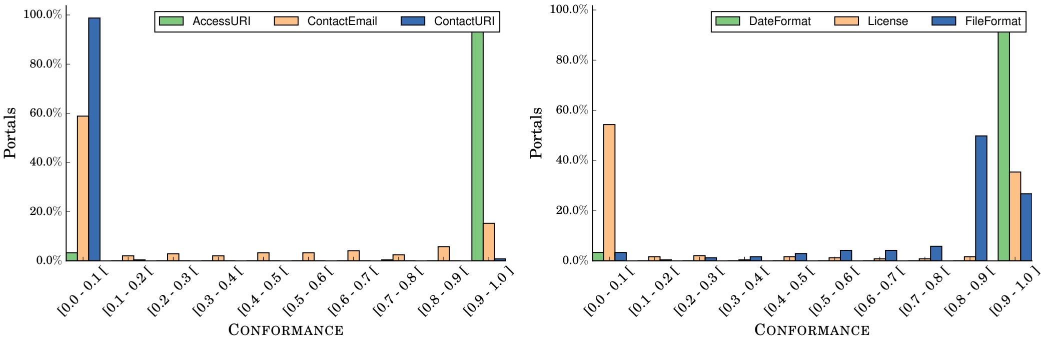
Similarly, only 50% of the portals have a Rights value over 0.9 (i.e., there exists licensing information) and furthermore, about 45% of the portals do not provide any licensing information at all. This absence of license and rights information is extremely worrying considering that one core requirements for Open Data is that the data is published under an license which allows the open use of the content.

6.4. Conformance

Our results about the various metrics in the conformance quality dimension are shown in the histograms in Figure 7. The left histogram in Figure 7 shows the conformance distribution for the AccessURI, ContactEmail and ContactURI metric. Considering the conformance of access URIs (i.e., if the resource references are valid URIs), we observe that over 95% of the portals have an average AccessURI conformance value of over 0.9, indicating that the values are syntactically valid URLs.

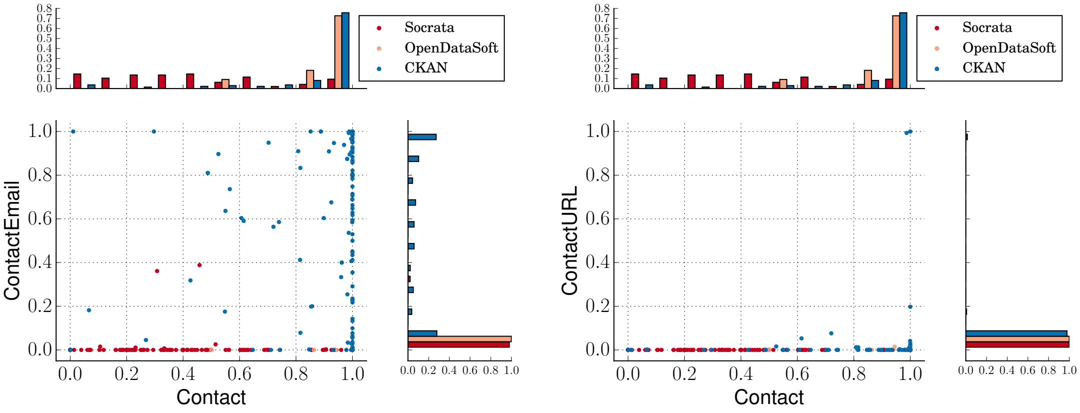
Regarding the conformance of available contact information, we discover that only a small subset consisting of 20% of the portals have an average ContactEmail value of over 0.9 and about 60% of the portals do not really contain any valid email information. Regarding the appearance of URLs, we observed an average URL contactability for almost all portals of less than 0.1 (with one portal in the range 0.1 – 0.2, and only 2 portals with a value over 0.9, namely `data.overheid.nl` and `data.graz.gv.at`). This results show that there are basically no URLs or email addressed provided for contacting the publisher or maintainer of a dataset and a user would need to manual search for such information based on the provided text values.

The right histogram in Figure 7 shows the remaining conformance metrics for DateFormat, License and FileFormat. Interestingly, for only ~40% of the portals we were able to map almost all licenses (a License value over 0.9) to the list of licenses reviewed by opendefinition.org. The majority of the remaining portals have a value of less than 0.1. This shows, that there is more (manual) work necessary to be able to automatically identify the license information.



Access and Contact information. Date, License and File Format information.

Fig. 7: CONFORMANCE Histograms.



Contact vs ContactEmail. Contact vs ContactURI.

Fig. 8: Existence and conformance of contact information.

Regarding the specified file formats, we can see that about 80% of the portals have a FileFormat value of over 0.8, i.e., that for these portals almost all file format description are using format identifiers which are registered by the Internet Assigned Numbers Authority.[16] The DateFormat conformance for the occurring date descriptions is in general very high.

Overall, we can conclude that the majority of the portals have a low contactability value which bears the risk that data consumers stop using dataset if they cannot contact the maintainer or author (e.g., regarding the re-use if the license is not clearly specified or in case of any data related issue). Further, we have to admit that an automated identification of licenses is very hard to achieve, and that a better source of license IDs and licensing information is required to get better license conformance results.

6.4.1. Conformance vs. Existence. In the following we discuss the relation between existence and conformance values for different metrics for all portals in form of scatter plots. We further categorise the portals according to their publishing software.

[16] http://www.iana.org/, last accessed 2015-11-02

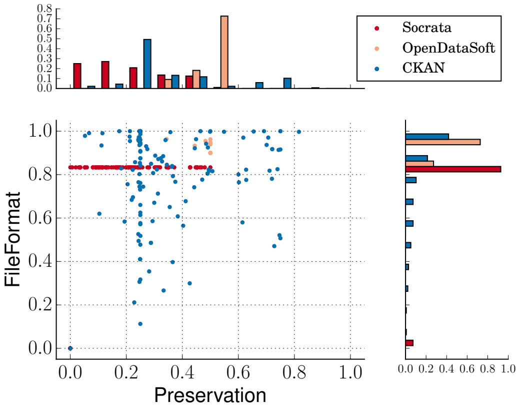


Fig. 9: Preservation vs FileFormat.

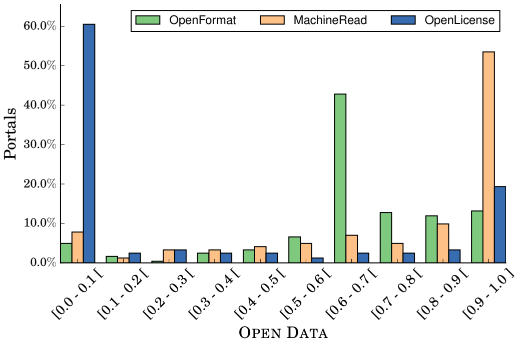


Fig. 10: Openness of Licenses and Formats.

The scatter plots in Figure 8 shows the relation between the Contact-existence and the ContactEmail and ContactURI conformance metrics, respectively. We can see in both plots that in contrary to CKAN portals, the OpenDataSoft and Socrata portals (red and yellow coloured) do not provide contact information which are valid email or URL addresses. Further, we can also see that the existence of any contact information is rather high for OpenDataSoft and CKAN portals and equally distributed for Socrata portals.

Figure 9 displays the Preservation and FileFormat metrics grouped by the portal software. Interestingly, the file format conformance based on the IANA registration is in general rather high. Drilling deeper, we see that almost all Socrata portals have a FileFormat value of about 0.8. The reason for this is that the Socrata software represents the actual data in 6 different formats with their respective media types (e.g. CSV, JSON, XML, RDF) and out of these file formats and media types the values `application/excel` and `application/xml+rdf` are not registered by the IANA; resulting in a conformance values of 10/12.

Noticeable in this plots is the Preservation value of 0.25 (x-axis) for a high percentage of CKAN portals. The reason for this observation is that most of the datasets in CKAN portal provide preservation information which can be mapped to only one of the four DCAT keys, namely `dct:format`, resulting in an average value of 1/4. We observe a similar result for the OpenDataSoft portals with the majority of the portals showing an preservation value between 0.5 – and 0.6.

6.5. Openness

It is crucial for the Open Data movement that published datasets and formats are adhering to the open definition and that everybody is allowed to use, re-use and modify the information which should be provided in an open and machine-readable format.

Table VII shows the top-10 licenses and the number of datasets they are specified in (after applying our introduced license mapping algorithm) and the top-10 used formats and the number of unique resources together with their number of portals they appear in. Bold highlighted values indicate that the license or format is considered as open by our metric.

The first surprising observation is that ∼23% of all the resources are published as PDF files. This is remarkable, because strictly speaking, PDF cannot be considered as an Open Data format: while PDFs may contain structured data (e.g. in tables) there are no standard ways to extract such structured data from PDFs - or general-purpose

Table VII: Top-10 formats and licenses.

license_id	$\|\mathcal{D}\|$	%	$\|\mathcal{P}\|$	format	$\|\mathcal{R}\|$	%	$\|\mathcal{P}\|$
ca-ogl-lgo	239662	32.3	1	**PDF**	804290	22.9	103
notspecified	193043	26	71	**HTML**	776696	22.1	82
dl-de-by-2.0	55117	7.4	7	**XML**	244654	7	90
CC-BY-4.0	49198	6.6	84	**CSV**	226694	6.5	120
us-pd	35288	4.8	1	other	180088	5.1	5
OGL-UK-3.0	33164	4.5	18	XLS	99626	2.8	97
other-nc	27705	3.7	21	orig. d. f.[a]	98135	2.8	1
CC0-1.0	9931	1.3	36	geotif	95391	2.7	2
dl-de-by-1.0	9608	1.3	6	**ZIP**	66963	1.9	87
Europ.Comm.[b]	8604	1.2	2	**tiff**	66075	1.9	12
others	80164	10.8		others	846286	24	

[a] *originator data format*, occurring only on data.gov
[b] http://open-data.europa.eu/kos/licence/EuropeanCommission

document formats in general. Therefore, we do not consider data published in PDFs as machine-readable, nor a suitable way for publishing Open Data.[17]

By looking at the top-10 used formats in VII, we can see that most of the top-10 formats are covered by open formats but only XML and CSV can be considered as machine-readable. The occurrence of the format descriptions "other", "originator data format" and "geotif" within very few portals suggests that there are data catalogs which do not stick to conventions or consist of insufficient format specifications.[18]

Regarding the used license IDs in Table VII, we observe that the confirmed open licenses in the top-10 cover only $\sim$17% of all datasets. Further, we notice that some of the more frequent used licenses are only used in very few portals. For instance, "ca-ogl-lgo" is the Canadian Open Government License[19] which is a share-alike license used throughout the Canadian Open Data portal open.canada.ca.

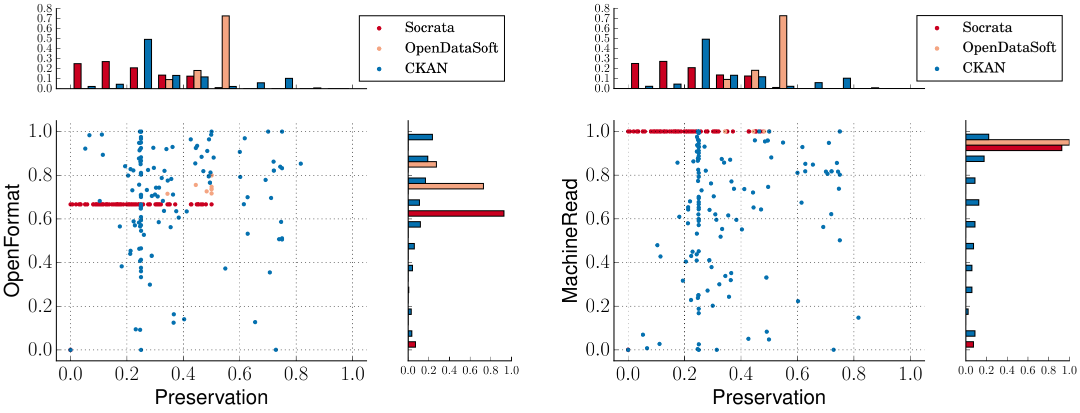
In addition, Figure 10 shows the distribution of the average OpenFormat, MachineRead, and OpenLicense values per portal. We can see that around 20% of the portals have a high average confirmed license openness value of over 0.9. There is also a large group of around 60% of the portals for which we could only confirm an average license openness per dataset of less than 0.1.

Considering file format information, we observe a high machine readability with around 60% of the portals having an average value of over 0.9. In contrast, the average values for the OpenFormat metric spread more or less equally from 0.1 to 0.9, with a peak of about 40% of the portals for the values 0.6 - 0.7.

Overall, we could not confirm for the majority of the portals that their datasets provide an open license and their resources are available in open formats. However, the machine readability of formats yields marginally better results.

6.5.1. Existence and Open Data fitness of formats. The scatter plot in Figure 11 displays the Preservation-existence values vs the confirmed openness (left plot) and vs. machine-readability (right plot) values of file format descriptions.

[17] Although there are works on extracting structured information from PDFs (e.g. tabular data within PDFs [Yildiz et al. 2005]), this topic is complementary to the scope of our paper.
[18] Please note that "geotif" in Table VII is not a spelling error.
[19] http://open.canada.ca/en/open-government-licence-canada, last accessed 2015-11-11



Preservation vs OpenFormat.

Preservation and MachineRead.

Fig. 11: EXISTENCE vs OPEN DATA fitness of Format descriptions.

Inspecting the left side plot focusing on the openness of the file formats, we can again observe a consistent OpenFormat value around 0.65 for the Socrata portals. This results from the 6 possible format representation the Socrata software offers by default. Similarly, we see a high percentage of OpenDataSoft portals with a OpenFormat value in the range between 0.7 – 0.8. In contrast, the values for CKAN portals spread across the whole spectrum.

Considering the machine-readability of the formats on the right plot, we see that all offered file formats for Socrata and OpenDataSoft portals are identified as machine-readable, whereas average values for the CKAN portals are equally distributed.

We notice from the various scatter plots that Socrata and OpenDataSoft portals show very homogeneous results with respect to our DCAT conformance and openness dimensions. This is kind of expected since both frameworks provide not the flexibility to publish data in arbitrary formats. In fact, both systems require to upload the data in tabular representations. In contrast, the CKAN software is highly extensible and does not put any restrictions on the file formats which results in more heterogeneous quality values.

7. RELATED WORK

Data quality assessment and improvement methodologies are widely used in various research areas such as relational databases, data warehouses, information or process management systems [Strong et al. 1997; Jarke and Vassiliou 1997], but also to assess the quality of Linked Open Data. To gain a deeper insight into current approaches for assessing the data quality of Linked Open Data, we refer to the work by Zaveri et al. [2015], which provides a comprehensive literature survey.

Over times, different application and research areas established various measures and techniques to assess the quality of data and services and for keeping up with the increasing complexity of the tasks [Zhu et al. 2012]. Batini et al. [2009] published a detailed and systematic description of methodologies to assess and improve data quality. Generally, the different methodologies involve various phases starting from the definition of quality metrics, the measurement, an analysis phase and possible improvements with small differences how feedback loops are integrated.

7.1. Related Efforts on Metadata Quality Assessment

Pipino et al. [2002] discuss a set of data quality dimensions and their subjective- and objectiveness on a very general level. Similarly, in [Wang and Strong 1996] the authors provide two comprehensive surveys: a survey of data quality attributes and a survey of data quality dimension. Wang et al. grouped the dimensions into four different information quality aspect and built a conceptual framework (i.e., a hierarchy) of data quality: (i) intrinsic, (ii) contextual, (iii) representational, and (iv) accessibility. For instance, the intrinsic quality aspect holds the "believability", "accuracy", "objectivity" and "reputation" dimensions. Michnik and Lo [2009] further refine and extend the four-aspect approach of [Wang and Strong 1996], e.g., by introducing sub-categories. While these efforts discuss quality dimensions on a very general level, we discuss the concrete computation and automated assessment of quality metrics based on the DCAT metadata schema.

In contrast to our approach, in [Margaritopoulos et al. 2008] the authors present the application of "logic rules" to assess the quality of metadata. They identify three different types of rules: *rules of inclusion*, *rules of imposition*, and *rules of restriction*. The definition of these rules is based on dependencies and relations of resources. For instance, applying a *rule of restriction* to a resource's metadata field means that the values "[. . .] must include the values of the same metadata field or records of related resources".

This rule-based approach can be considered as an automated metadata quality assessment. While there are already various approaches of automated metadata quality evaluation [Hughes and Kamat 2005; Najjar et al. 2003], also by using simple statistical evaluations [Greenberg et al. 2001; Moen et al. 1998; Wilson 2007], a manual evaluation is often unavoidable and therefore very common [Greenberg et al. 2001; Moen et al. 1998; Wilson 2007].

Regarding quality assessment within the 5S model (cf. Section 2) Gonçalves et al. [2007] discuss quality dimensions and measures for each of the 5S main concepts. Further, the authors provide an example evaluation of the dimensions and discuss the practical utility of the proposed quality model. In relation, Moreira et al. [2007] presented 5SQual, a quality assessment tool built upon the 5S model which automatically assesses and evaluates eight different quality dimensions. While these quality assessment approaches focus on the concepts of the 5S model, we focus in our work on the quality of metadata descriptions in data catalogs.

7.2. Contributions regarding Open Data Quality

When looking into related work on data quality, to the best of our knowledge, not much work is published for QA in Open Data. However, in recent years, several projects addressed the Open Data domain and we identify projects which deal with the quality of Open Government Data (aligned to Barack Obama's Open Government Directive [Orszag 2009]) and aim to assess and compare the state of Open Data across different countries [World Wide Web Foundation 2015; Bertot et al. 2012]. Further, we identify recent projects which try to assess the progress and spreading of Open Data in the private sector, e.g. the Open Company Data Index,[20] a report by the World Bank Institute which aims to register open corporate data and provides an aggregated quality assessment per country. In the following we highlight projects which propose various metrics to evaluate the (meta-)data quality within open data catalogues.

In relation to data quality assessment in Open Government Data catalogues, such as `data.gov` or `data.gv.at`, recent work by Kučera et al. [Kucera et al. 2013] discusses

[20] http://registries.opencorporates.com/, last accessed 2015-10-30

quality dimensions and requirements of such catalogues. The authors list and discuss relevant quality dimensions (*Accuracy*, *Completeness*, *Consistency* and *Timeliness*) but, unfortunately, the work is short of detail in some respects.

More related to the actual data quality assessment is the Global Open Data Index project[21] and the Open Data Monitor project.[22] Both projects define a set of Open Data specific quality metrics and rank various countries by their state of Open Data. However, while the Global Open Data Index is based on a manual expert evaluation, the Open Data Monitor project mainly uses dimensions which can be assessed automatically (e.g., the completeness of metadata). In principle, to the best of our knowledge, all of the above-mentioned efforts either rely on a manual evaluation of their quality dimensions and therefore do not provide an automated assessment as we do with our framework; or the projects do not deal with heterogeneous metadata and therefore do not provide a generic and large-scale quality analysis of metadata Open Data portals.

Complementary to the quality assessment approaches, the OPQUAST project [de Dona et al. 2012] proposes a checklist for publishing Open Data, including questions related to quality aspects. This checklist is very extensive and the questions reach from general questions about the data catalog (e.g., "*The concept of Open Data is explained*") to in-detail questions about specific metadata keys and available meta-information (e.g., "*It is possible to obtain information regarding the level of trust accorded to the data*").

Most closely related to the efforts in this paper are [Ochoa and Duval 2009; Braunschweig et al. 2012; Reiche et al. 2014]. The authors discuss a set of quality metrics for metadata in digital repositories, including a detailed description, definition and evaluation of the metrics. [Reiche et al. 2014] also identified the need for an automatic quality assessment and monitoring framework to better understand quality issues in Open Data portals and to study the impact of improvement methods over time. The authors developed a prototype of such a framework which is unfortunately now offline.[23]

Although [Reiche et al. 2014] influenced the herein presented metrics and framework, we extended the work of Reiche et al. in terms of generalized and useful quality metrics in the context of Open Data (e.g., by adding a contactability and open format metric), in terms of the extent of monitored data portals and in terms of a continuous monitoring of these portals.

7.3. Alternative Efforts on Modeling Digital Catalogs

Various efforts already exist to study the formal theory of digital libraries. On the one hand, there is most prominently the 5S model, which we already mentioned in Section 2. In contrast to the 5S model, the DELOS Reference Model [Agosti et al. 2006; Candela et al. 2007] models a digital library by using the following six main concepts: content, user, functionality, quality, policy, and architecture. The DELOS model is formulated as an entity-relationship model and the structure is mainly hierarchical. The aforementioned concepts represent high level containers, e.g. the "content" concept holds the *Resource* entity and the "quality" concept holds the *Quality Parameter* entity and these two entities are related: a quality parameter is evaluated on a resource. [Agosti et al. 2007] describes and compares the DELOS Reference Model to the 5S Framework. In particular, it compares the quality aspects for the 5S model described in [Gonçalves et al. 2007] with the quality aspects of the DELOS model [Candela et al. 2007].

[21] http://index.okfn.org, last accessed 2015-10-30

[22] http://www.opendatamonitor.eu/frontend/web/index.php, last accessed 2015-10-30

[23] http://metadata-census.com, last accessed 2015-03-06

In [Ferro and Silvello 2013] the authors propose “NESTOR”, a formal model for digital archives. The formally defined model is based on nested sets, where subset relations correspond to different hierarchies within an archive. Further, the authors use the proposed model to map and extend the 5S model [Gonçalves et al. 2004].

8. CONCLUSIONS

The Open Data movement enjoys great popularity and enthusiasm mostly driven by public administration, but also increasingly gaining attention in the private sector. However, there are metadata quality issues that could disrupt the success of Open Data: inadequate descriptions or classifications of datasets directly affect the usability and searchability of resources. In order to assess the severity of these issues, the present work has proposed a set of objective quality metrics (based on the W3C metadata schema DCAT) to monitor the quality of Open Data portals in a generic and automated way. Moreover, we have introduced a generic abstraction of web-based data portals for the purpose of integrating a large amount of existing data portals in an extensible manner: based on prior metadata homogenization approaches, we have mapped the metadata occurring on the main Open Data publishing systems (i.e. CKAN, Socrata and OpenDataSoft) to DCAT. Based on this mapping we have implemented and deployed an Open Data portal monitoring and quality assessment framework – the “Open Data Portal Watch” platform[24] – that monitors our metrics in weekly snapshots of the metadata from over 261 Open Data portals. Currently, our framework monitors 148 CKAN, 102 Socrata and 11 OpenDataSoft portals, consisting of a total of 110k datasets and 2.2M distinct resource URLs. Our core findings and conclusions can be summarized as follows:

- Several DCAT properties cannot yet be automatically mapped to the existing metadata models of the three portal frameworks. For instance, properties concerned with spatial or temporal information (represented in DCAT by the `dct:temporal` and `dcat:spatial` properties), which would be particularly important for searchability of relevant datasets for a specific local or temporal context, can not be mapped to existing default metadata keys. Furthermore, the existence of some specific metadata properties is highly portal dependent (extra keys): I.e., there is a lot of heterogeneity among such extra metadata keys, some of which – by the way – in fact describe the missing DCAT properties concerning spatial and temporal context, but agreement on these extra metadata keys among portals (even using the same software framework) seems still to be relatively low in some respects.
- Slightly alarming, we were able to perform HTTP HEAD lookups on only 78% of the resource URLs without any errors or restrictions – which seems to indicate deficits even in terms of proper implementation of standard HTTP features.
- We observed that there is a gap between the common definition (i.e., the Open Definition[25]) and the actual state of Open Data regarding the use of machine-readable Open formats and the existence of license specifications and compliance to actual Open licenses.
- Last, but not least, the majority of the datasets do not provide machine-readable contact information (e.g. in the form of a valid email address or URLs): missing provenance information – in our opinion – involves the risk of intransparency and impaired usability of datasets and prohibits valuable user feedback.

As for future work, in order to further improve our framework and quality metrics, we prioritize (i) the evaluation of more portals (including other software frameworks

[24] http://data.wu.ac.at/portalwatch/

[25] http://opendefinition.org/, last accessed 2016-03-09

and HTML-based data portals), (ii) the efficient monitoring of the actual resource content for the evaluation of metadata accuracy and finally (iii) further refinement of the openness metrics regarding various licenses and formats.

Our monitoring framework performs weekly runs, i.e. it retrieves and stores weekly snapshots of metadata from several portals over a longer period. The continuous monitoring of data catalogues allows us to estimate the development and growth-rate of Open Data. A planned application of this information is a more detailed evolution analysis of the retrieved data wrt. additions and deletions of datasets and their contents, as well as the effect of these changes on specific quality dimensions. Further, the continuous monitoring shall allow us to develop improvement methods and initiatives, e.g. we are able to report back to the portal and data providers the impact and distribution rate of metadata homogenization efforts or other improvements.

We shall further research solutions to improve and extend our proposed DCAT mapping in order to deal with "extra keys", i.e., addressing the high heterogeneity of metadata keys and values in and across portals, possibly, for instance, looking into methods from automated, instance-based ontology matching and alignment [Shvaiko and Euzenat 2013]. A first, simple approach is the manual extension of the static mapping by the most commonly used metadata keys, e.g., by mapping a set of conceptually similar CKAN keys to a corresponding DCAT property. Moreover, we plan to make our (evolving) RDFized DCAT metadata mappings publicly available.

A complementary, additional potential use of the acquired information in our system would be the (semi-)automated addition and correction of respectively missing and wrong metadata, e.g. by suggesting values for certain missing metadata fields, or by automatically checking the consistency of existing fields in comparison to actual values.

REFERENCES

Maristella Agosti, Leonardo Candela, Donatella Castelli, Nicola Ferro, Yannis Ioannidis, Georgia Koutrika, Carlo Meghini, Pasquale Pagano, Seamuss Ross, H. J. Schek, and H. Schuldt. 2006. *A Reference Model for DLMSs Interim Report*. Deliverable. DELOS.

Maristella Agosti, Nicola Ferro, Edward A Fox, and Marcos A Gonçalves. 2007. Modelling DL quality: A comparison between approaches: The DELOS reference model and the 5S model. In *Second DELOS Conference on Digital Libraries*. Tirrenia, Pisa, Italy, 5–7.

Keith Alexander, Richard Cyganiak, Michael Hausenblas, and Jun Zhao. 2011. Describing Linked Datasets with the VoID Vocabulary. https://www.w3.org/TR/void/. (March 2011).

Ahmad Assaf, Raphaël Troncy, and Aline Senart. 2015. HDL - Towards a harmonized dataset model for open data portals. In *PROFILES 2015, 2nd International Workshop on Dataset Profiling & Federated Search for Linked Data, Main conference ESWC15, 31 May-4 June 2015, Portoroz, Slovenia*. CEUR-WS.org. http://www.eurecom.fr/publication/4543

Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. 2009. Methodologies for Data Quality Assessment and Improvement. *ACM Comput. Surv.* 41, 3, Article 16 (July 2009), 52 pages. DOI:http://dx.doi.org/10.1145/1541880.1541883

John Carlo Bertot, Patrice McDermott, and Ted Smith. 2012. Measurement of Open Government: Metrics and Process. *2014 47th Hawaii International Conference on System Sciences* (2012), 2491–2499. DOI:http://dx.doi.org/10.1109/HICSS.2012.658

Christian Bizer and Richard Cyganiak. 2009. Quality-driven information filtering using the WIQA policy framework. *J. Web Sem.* 7, 1 (2009), 1–10. DOI:http://dx.doi.org/10.1016/j.websem.2008.02.005

Katrin Braunschweig, Julian Eberius, Maik Thiele, and Wolfgang Lehner. 2012. The State of Open Data - Limits of Current Open Data Platforms. In *Proceedings of the International World Wide Web Conference, WWW 2012, Lyon, France*. ACM.

Leonardo Candela, Donatella Castelli, Pasquale Pagano, Costantino Thanos, Yannis E. Ioannidis, Georgia Koutrika, Seamus Ross, Hans-Jörg Schek, and Heiko Schuldt. 2007. Setting the Foundations of Digital Libraries: The DELOS Manifesto. *D-Lib Magazine* 13, 3/4 (2007). DOI:http://dx.doi.org/10.1045/march1007-castelli

Muriel de Dona, Elie Sloïm, Laurent Denis, and Fabrice Bonny. 2012. *Qualité Web : Les bonnes pratiques pour amÈliorer vos sites*. Temesis. http://amazon.com/o/ASIN/2954303107/

Nicola Ferro and Gianmaria Silvello. 2013. NESTOR: A formal model for digital archives. *Inf. Process. Manage.* 49, 6 (2013), 1206–1240. DOI:http://dx.doi.org/10.1016/j.ipm.2013.05.001

Edward A. Fox, Marcos André Gonçalves, and Rao Shen. 2012. *Theoretical Foundations for Digital Libraries: The 5S (Societies, Scenarios, Spaces, Structures, Streams) Approach*. Morgan & Claypool Publishers. DOI:http://dx.doi.org/10.2200/S00434ED1V01Y201207ICR022

Marcos André Gonçalves, Edward A. Fox, Layne T. Watson, and Neill A. Kipp. 2004. Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries. *ACM Trans. Inf. Syst.* 22, 2 (2004), 270–312. DOI:http://dx.doi.org/10.1145/984321.984325

Marcos André Gonçalves, Bárbara Lagoeiro Moreira, Edward A. Fox, and Layne T. Watson. 2007. "What is a good digital library?" - A quality model for digital libraries. *Inf. Process. Manage.* 43, 5 (2007), 1416–1437. DOI:http://dx.doi.org/10.1016/j.ipm.2006.11.010

Jane Greenberg, Maria Cristina Pattuelli, Bijan Parsia, and W. Davenport Robertson. 2001. Author-generated Dublin Core Metadata for Web Resources: A Baseline Study in an Organization. In *DC-2001, Proceedings of the International Conference on Dublin Core and Metadata Applications 2001*. National Institute of Informatics, Tokyo, Japan, 38–46. http://www.nii.ac.jp/dc2001/proceedings/abst-06.html

Andreas Harth, Jürgen Umbrich, and Stefan Decker. 2006. MultiCrawler: A Pipelined Architecture for Crawling and Indexing Semantic Web Data. In *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*. 258–271. DOI:http://dx.doi.org/10.1007/11926078_19

Baden Hughes and Amol Kamat. 2005. A Metadata Search Engine for Digital Language Archives. *D-Lib Magazine* 11, 2 (2005). DOI:http://dx.doi.org/10.1045/february2005-hughes

Matthias Jarke and Yannis Vassiliou. 1997. Data Warehouse Quality: A Review of the DWQ Project. In *Second Conference on Information Quality (IQ 1997)*. MIT, 299–313.

Jan Kucera, Dusan Chlapek, and Martin Necaský. 2013. Open Government Data Catalogs: Current Approaches and Quality Perspective. In *Technology-Enabled Innovation for Democracy, Government and Governance - Second Joint International Conference on Electronic Government and the Information Systems Perspective, and Electronic Democracy, EGOVIS/EDEM 2013, Prague, Czech Republic, August 26-28, 2013, Proceedings*. 152–166. DOI:http://dx.doi.org/10.1007/978-3-642-40160-2_13

Fadi Maali and John Erickson. 2014. Data Catalog Vocabulary (DCAT). http://www.w3.org/TR/vocab-dcat/. (Jan. 2014).

Thomas Margaritopoulos, Merkourios Margaritopoulos, Ioannis Mavridis, and Athanasios Manitsaris. 2008. A Conceptual Framework for Metadata Quality Assessment. In *Proceedings of the 2008 International Conference on Dublin Core and Metadata Applications (DCMI '08)*. Dublin Core Metadata Initiative, 104–113. http://dl.acm.org/citation.cfm?id=1503418.1503429

Jerzy Michnik and Mei-Chen Lo. 2009. The assessment of the information quality with the aid of multiple criteria analysis. *European Journal of Operational Research* 195, 3 (2009), 850–856. DOI:http://dx.doi.org/10.1016/j.ejor.2007.11.017

William E. Moen, Erin L. Stewart, and Charles R. McClure. 1998. Assessing Metadata Quality: Findings and Methodological Considerations from an Evaluation of the U.S. Government Information Locator Service (GILS). In *Proceedings of the IEEE Forum on Reasearch and Technology Advances in Digital Libraries, IEEE ADL '98, Santa Barbara, California, USA, April 22-24, 1998*. 246–255. DOI:http://dx.doi.org/10.1109/ADL.1998.670425

Bárbara Lagoeiro Moreira, Marcos André Gonçalves, Alberto H. F. Laender, and Edward A. Fox. 2007. 5SQual: a quality assessment tool for digital libraries. In *ACM/IEEE Joint Conference on Digital Libraries, JCDL 2007, Vancouver, BC, Canada, June 18-23, 2007, Proceedings*. 513. DOI:http://dx.doi.org/10.1145/1255175.1255313

Jehad Najjar, Stefaan Ternier, and Erik Duval. 2003. The Actual Use of Metadata in ARIADNE: an Empirical Analysis. In *Proceedings of the 3rd ARIADNE Conference*. 1–6.

Marc Najork and Allan Heydon. 2002. High-Performance Web Crawling. In *Handbook of Massive Data Sets*. Massive Computing, Vol. 4. Springer US, 25–45. http://dx.doi.org/10.1007/978-1-4615-0005-6_2

Xavier Ochoa and Erik Duval. 2009. Automatic evaluation of metadata quality in digital repositories. *Int. J. on Digital Libraries* 10, 2-3 (2009), 67–91. DOI:http://dx.doi.org/10.1007/s00799-009-0054-4

Peter Orszag. 2009. Open Government Directive. https://www.whitehouse.gov/open/documents/open-government-directive. (2009). Memorandum for the Heads of Executive Departments and Agencies.

Leo Pipino, Yang W. Lee, and Richard Y. Wang. 2002. Data quality assessment. *Commun. ACM* 45, 4 (2002), 211–218. DOI:http://dx.doi.org/10.1145/505248.5060010

Konrad Johannes Reiche, Edzard Höfig, and Ina Schieferdecker. 2014. Assessment and Visualization of Metadata Quality for Open Government Data. In *Proceedings of the International Conference for E-Democracy and Open Government, CeDEM14, 2014, Krems, Austria, May 21-23, 2014*.

Pavel Shvaiko and Jérôme Euzenat. 2013. Ontology Matching: State of the Art and Future Challenges. *IEEE Trans. Knowl. Data Eng.* 25, 1 (2013), 158–176. DOI:http://dx.doi.org/10.1109/TKDE.2011.253

Manu Sporny, Gregg Kellogg, and Markus Lanthaler. 2014. JSON-LD 1.0A JSON-based Serialization for Linked Data. http://www.w3.org/TR/json-ld/. (Jan. 2014).

Diane M Strong, Yang W Lee, and Richard Y Wang. 1997. Data quality in context. *Commun. ACM* 40, 5 (1997), 103–110.

Jürgen Umbrich, Sebastian Neumaier, and Axel Polleres. 2015. Quality assessment & evolution of Open Data portals. In *The International Conference on Open and Big Data*. IEEE, Rome, Italy, 404–411. DOI:http://dx.doi.org/10.1109/FiCloud.2015.82

Richard Y. Wang and Diane M. Strong. 1996. Beyond Accuracy: What Data Quality Means to Data Consumers. *J. of Management Information Systems* 12, 4 (1996), 5–33. http://www.jmis-web.org/articles/1002

Stuart Weibel, John Kunze, Carl Lagoze, and Misha Wolf. 1998. *Dublin Core Metadata for Resource Discovery*. Technical Report. USA.

Amanda J. Wilson. 2007. Toward releasing the metadata bottleneck - A baseline evaluation of contributor-supplied metadata. *Library Resources & Technical Services* 51, 1 (Jan. 2007), 16–28. DOI:http://dx.doi.org/10.5860/lrts.51n1.16

World Wide Web Foundation. 2015. Open Data Barometer. (Jan. 2015).

Burcu Yildiz, Katharina Kaiser, and Silvia Miksch. 2005. pdf2table: A Method to Extract Table Information from PDF Files. In *Proceedings of the 2nd Indian International Conference on Artificial Intelligence, Pune, India, December 20-22, 2005*. 1773–1785.

Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. 2015. Quality Assessment for Linked Data: A Survey. *Semantic Web Journal* 7, 1 (March 2015), 63–93. DOI:http://dx.doi.org/10.3233/SW-150175

Hongwei Zhu, Stuart E Madnick, Yang W Lee, and Richard Y Wang. 2012. Data and Information Quality Research: Its Evolution and Future. In *Computing Handbook, Third Edition: Information Systems and Information Technology*. CRC Press, USA, 16: 1–20.

Anneke Zuiderwijk, Marijn Janssen, Sunil Choenni, Ronald Meijer, and Roexsana Sheikh Alibaks. 2012. Socio-technical Impediments of Open Data. *Electronic Journal of e-Government* 10, 2 (2012), 156–172.