Undefined 0 (2014) 1–0
IOS Press

LOD in a Box: The C-LOD Meta-Dataset

Laurens Rietveld, Wouter Beek and Stefan Schlobach
Department of Computer Science, VU University Amsterdam, The Netherlands
E-mail: {laurens.rietveld,w.g.j.beek,stefan.schlobach}@vu.nl

Abstract. This paper introduces the C-LOD (Clean & Linked Open Data) Meta-Dataset, a continuously updated Meta-Dataset of the LOD cloud, tightly connected to the (re)published corresponding datasets which are crawled and cleaned by the LOD Laundromat [2]. The C-LOD Meta-Dataset contains meta-data for over 12 billion triples (and growing). While traditionally dataset meta-data is often not provided, incomplete, or incomparable in the way they were generated, the C-LOD Meta-Dataset provides a wide variety of those properties using standardized vocabularies. This makes it a particularly useful dataset for data comparison and analytics, as well as for the global study of the Web of Data.

Keywords: Dataset Meta-data, Linked Data, Dataset Descriptions

1. Introduction

In this paper we present the Clean & Linked Open Data (C-LOD) Meta-Dataset, a uniform collection of structural metadata properties that describes very many (over 26,000) existing Linked Data Documents, containing over 12 billion triples. The C-LOD Meta-Dataset is unique in its scale (both in terms of datasets and number of meta-data properties), the consistent way in which meta-data properties are calculated, the explicit description of the computational processes used to calculate these properties, and the use cases it supports. With respect to the latter, the C-LOD Meta-Dataset uniquely facilitates the analysis and comparison of very many datasets, as well as supporting Big Data research scenarios in which algorithms make innovative use of meta-data values (e.g., in order to improve performance).

Analyzing, comparing, and using multiple Linked Open Datasets currently requires the hassle of finding a download location, hoping the downloaded data dumps are valid, and parsing the data in order to analyze or compare it based on some criterion. It is even more difficult to search for datasets based on characteristics that are relevant for machine-processing, such as syntactic conformance and structural properties such as the average outdegree of nodes. What is needed is a uniform representation of the *dataset* and a uniform representation of *dataset descriptions*.

The LOD Laundromat [2] already realizes the first: it (re)publishes the largest (collection of) dataset(s) on the Web of Data (over twelve billion triples and counting). We call this collection the C-LOD Collection. Every dataset is published in the same format that is fully conformant with Linked Open Data (LOD) publication standards for machine-processability. The purpose of the C-LOD Collection is to drastically simplify the task of data preprocessing for the data consumer.

However, the creation of meta-data describing the datasets is still left up to the original data publisher. We see that many data publishers do not publish a dataset description that can be found by automated means, and that those data descriptions that can be found do not always contain all (formally or de-facto) standardized meta-data. More importantly, the meta-data values are generally not comparable between datasets since different data publishers may interpret and calculate the same meta-data property differently. For instance, it is not generally the case that a dataset with a higher value for the `void:triples` property contains more triples. Because of such incompatibilities between existing dataset descriptions, it is difficult to reliably analyze and compare datasets on a large scale and use structural metadata properties in order to fine-tune Big Data algorithms.

Therefore, next to the uniform *dataset* representations that are published by the LOD Laundromat,

we need the same uniform representation for publishing *dataset meta-data*. In addition to uniformity, even straightforward meta-data should come with provenance annotations that describe how meta-data was generated. The here presented C-LOD Meta-Dataset brings exactly this: a collection of dataset descriptions, linked to the same canonical dataset representation, all modeled, created, and published in the same manner, and with provenance annotations that explain how meta-data was generated.

In section 2 we given an overview of comparable datasets. In section 3 we identify shortcomings in existing meta-data standards and collections, and formulate a set of requirements for a dataset that would allow large collections of datasets to be analyzed, compared, and used. Section 4 presents the meta-data we publish, the model that is used to publish in, the used external vocabularies, a discussion in the context of the five stars of Linked Data Vocabulary use, and clarification on how the C-LOD Meta-Dataset is generated and maintained. Section 5 shows the applications and use cases that the C-LOD Meta-Dataset and the C-LOD Collection offers. We conclude with section 6.

2. Comparable Datasets

[3] presents an overview of dataset descriptions that can be found by automated means, as surveyed by the SPARQL Endpoint Status service[1]. These results show that even the uptake of the core meta-data properties (such as the ones from the VoID specification) is still quite low: only 16.2% of the analyzed SPARQL endpoints contain self-describing VoID information. machine-readable. Because of this apparent lack of LOD meta-data, several initiatives tried to fill this gap by creating uniform metadata descriptions for multiple datasets.

Firstly, LODStats [1] provides statistical information for all Linked Open Datasets that are published in the CKAN-powered[2] Datahub[3] catalog. It offers a wide range of statistics, e.g., including the number of blank nodes in a dataset and the average outdegree of subject terms. Unfortunately, only a small subset of those statistics are themselves being published as Linked Data. Secondly, Sindice [8] provides statistical information similar to LODStats, but mostly analyzes smaller datasets that are crawled from Web pages. The meta-data provided by Sindice are similar to those in the VoID specification but they do not reuse a vocabulary as they are not encoded in a machine-readable format such as RDF.

Although Sindice and LODStats provide a step in the right direction by uniformly creating metadata descriptions for many Linked Datasets, they only support a subset of existing metadata properties, they do not publish exhaustive metadata descriptions as Linked Data, and they do not publish structural information on the meta-data generation procedure. Also, they are constrained to Linked Datasets that are published in only certain locations.

3. Meta-Data Requirements

In this section we present a requirements analysis for a dataset that satisfies our goal of supporting the meaningful analysis, comparison, and use, of very many datasets. We do so by explaining problems with respect to meta-data specifications (section 3.1), dataset descriptions (section 3.2) and collections of dataset descriptions (section 3.3). Finally, based on these considerations, we present our requirements (section 3.4).

3.1. Meta-data specifications

Existing dataset vocabularies include VoID[4], VoID-ext [7] and Bio2RDF [4]. VoID is a vocabulary for expressing metadata about Linked Datasets. It supports generic metadata (e.g., the homepage of a dataset), access metadata (e.g., which protocols are available), links to other datasets, exemplary resources, as well as dataset statistics (e.g., the number of triples). Only some of the VoID metadata properties can be automatically generated. Others can only be given by human authors, since they depend on interpretation. Bio2RDF presents a collection of dataset meta-data properties that extends the set of VoID properties and provides more detail. For example, Bio2RDF does not only extend the kinds of things that can be counted with, e.g., the number of classes, but also allows the full set of properties and classes to be represented by using VoID partitions. In addition, Bio2RDF adds entirely new properties, such as the number of unique entities that are linked to from each predicate. Also VoID-ext extends the set of meta-data properties that are found

[1] http://sparqles.okfn.org/
[2] http://ckan.org/
[3] http://datahub.io/
[4] See www.w3.org/TR/void/

in VoID. They include the in- and outdegree of entities, the number of blank nodes, the average string length of literals, and a partitioning of the literals and URIs based on string length.

We observe the following problems with these existing metadata specifications:

Firstly, several formal definitions of metadata properties are incorrect. As an example we take the VoID property `void:properties` which ought to denote the number of distinct RDF properties that occur in a dataset. The specification of this property conflates the semantic notion of an RDF property with the syntactic notion of an RDF term that appears in the predicate position of a triple. That these are quite different things is apparent from the following 4-line 'dataset' which contains 4 distinct predicate terms and 9 distinct RDF terms that denote an RDF property.

```
ex:p1 rdf:type rdf:Property ;
ex:p2 rdfs:subPropertyOf ex:p3 .
ex:p4 rdfs:domain ex:c1 .
ex:p5 rdfs:range ex:c2 .
```

As far as the number of distinct properties is concerned, this cannot be determined without the added assumption that the dataset is complete w.r.t. identity statements regarding property-denoting terms. Due to the Open World Assumption some other dataset may contain the following triple:

```
ex:p1 owl:sameAs ex:p5 .
```

Whether the data publisher uses the value 4 or 9 (or some other value) depends on her assessment of what the VoID authors may have intended to describe.

Secondly, some existing metadata properties are subjective. For example, `void:entities` is intended to denote a subset of the IRIs of a dataset based on "arbitrary additional requirements" imposed by the authors of the dataset description. Since different authors may impose different requirements, the number of entities of a dataset may vary between zero and the number of resources.

Thirdly, some existing metadata properties are defined in terms of undefined concepts. For example, LODStats specifies the set of vocabularies that are reused by a given dataset. The notion of a 'reused vocabulary' is itself not formally defined but depends on heuristics about whether or not an IRI belongs to another dataset. LODStats calculates this set by using relatively simple string operations according to which IRIs of the form `http://<authority>/<string>/<value>` are assumed to belong to the vocabulary denoted by `http://<authority>/<string>`. Although this is a fair attempt at identifying reused vocabularies, there is not always a bijective map between datasets and URI substrings that occur in datasets. The number of links to other datasets suffers from the same lack of a formal definition.

3.2. Dataset descriptions

We observe the following problems with existing dataset descriptions: Firstly, uptake of dataset descriptions that can be found by automated means is still quite low (section 2). Secondly, for reasons discussed above, the values of metadata properties that do not have a well-founded definition cannot be meaningful compared across datasets. E.g., if two dataset descriptions contain different values for the `void:entities` property it is not clear whether this denotes an interesting difference between the two datasets or whether this is due to the authors having different criteria for identifying the set of entities. Thirdly, even the values of well-defined metadata may have been calculated in different ways by different computational procedures. We observe that there are great discrepancies between meta-data which occurs *in* the original dataset description and those in the C-LOD Meta-Dataset.[5] Similar discrepancies exist between meta-data values that occur in different dataset description *collections*, e.g. between LODStats and the C-LOD Meta-Dataset.[6]

Since it is difficult to assess whether a computational procedure that generate meta-data is correct, we believe it is necessary that all generated meta-data is annotated with provenance information that describes the used computational procedure. Although relatively verbose, this approach circumvents the arduous discussion of which version of what tool is correct/incorrect for calculating a given meta-data value. We assume that there will always be multiple values for the same meta-data property. The fact that there are different values, and that these have been derived by different means, is something that has to be made transparent to the consumer of this meta-data. The onus is on the data consumer to trust one computational pro-

[5] E.g., the English Heritage Periods List (`http://purl.org/heritagedata/schemes/eh_tmt2`) contains 70,314 triples according to its original VoID description but 29 triples according to the C-LOD Meta-Dataset.

[6] E.g., according to LODStats the dataset located at `http://www.open-biomed.org.uk/open-biomed-data/sdgp-images-all-20110211.tar.gz` contains 1,080,060 triples while the C-LOD Meta-Dataset states 1,070,072.

cedure for calculating a specific meta-data value more than another.

3.3. Dataset description collections

We observe the following problems with existing collections of dataset descriptions: Firstly, even though the meta-data is calculated consistently within a collection, the computational procedure that is used is not described in a machine-processable format (if at all). This means that values can only be compared within the collection, but not with dataset descriptions external to the collection (e.g. occurring in other collections). Secondly, meta-data that is calculated within existing collections is not always published in a machine-interpretable format. (one example is LODStats)

3.4. Requirements

Based on the above considerations, we formulate the following requirements which allow multiple datasets to be meaningfully compared based on their meta-data:

1. The C-LOD Meta-Dataset must cover very many datasets in order to have a sufficiently wide scope.
2. The C-LOD Meta-Dataset must reuse official and de-facto meta-data standards in order to be compatible with other dataset descriptions, promoting reuse.
3. The C-LOD Meta-Dataset must be generated algorithmically in order to assure that values are calculated in the same way for every described dataset.
4. The meta-data must be calculated efficiently, because very many datasets are considered and some of them have quite peculiar properties that may not have been anticipated when the meta-data properties were first defined.
5. The C-LOD Meta-Dataset must contain provenance annotations that explain how and when the meta-data was calculated.
6. The C-LOD Meta-Dataset must be disseminated as LOD and must be accessible via a SPARQL endpoint.
7. The C-LOD Meta-Dataset must be able to support a wide range of real-world use cases that involve analyzing and/or comparing datasets as well as supporting Big Data algorithms that process LOD.

4. The C-LOD Meta-Dataset

In this section we present the meta-data we publish, the model we use, and how we generate this dataset.

4.1. Published Meta-Data

The C-LOD Meta-Dataset is generated in adherence to the requirements formulated in section 3. Since there are multiple ways in which these requirements can be prioritized and made concrete, we will now discuss the considerations that have guided the generation of the C-LOD Meta-Dataset.

Firstly, there is a conflict between requirements 2 and 3: since the C-LOD Meta-Dataset has to be constructed algorithmically, only well-defined meta-data properties can be included.

Secondly, there is a conflict between requirements 1 and 4 on the one hand, and requirement 2 on the other: since the C-LOD Meta-Dataset must describe very many datasets, some of which are relatively large, and we want calculations to be efficient, we chose to narrow down the set of meta-data properties to those that can be calculated by *streaming* the described datasets. This excludes properties that require loading (large parts of) a dataset into memory, e.g. in order to perform joins on triples.

Thirdly, because of the scale at which the C-LOD Meta-Dataset describes datasets, it is inevitable that some datasets will have atypical properties. This includes datasets with extremely long literals, or datasets where the number of unique predicate terms is close to the total number of predicate terms. It is only when meta-data is systematically generated on a large scale, that one finds such corner cases. These corner cases can make dataset descriptions impractically large, e.g., larger than the described dataset. This is especially true for meta-data properties that consist of enumerations. E.g., for some datasets the partition of all properties, as defined by VoID-ext and Bio2RDF, is only (roughly) a factor 3 smaller than the described dataset itself (and this is only one meta-data property). In order to keep data descriptions relatively small w.r.t. the dataset described, the C-LOD Meta-Dataset does not include properties whose values are dataset partitions.

Under these restrictions, the C-LOD Meta-Dataset is able to include a large number of datasets while still being relatively efficient to construct. Implementation-wise, the generation of the C-LOD Meta-Dataset takes into account the many advantages that come from the way in which LOD Laundromat (re)publishes C-LOD.

LOD Laundromat allows datasets to be opened as gzip-compressed streams of lexicographically sorted N-Triples. Since these streams are guaranteed to contain no syntax error nor any duplicate occurrences of triples, they can be processed on a line-by-line / triple-by-triple basis, making it convenient to generate meta-data for inclusion in the C-LOD Meta-Dataset. Table 1 gives an overview of the meta-data properties included in the C-LOD Meta-Dataset, together with those that are included in existing dataset description standards. As can be seen from the table, the only meta-data properties that are excluded from our dataset (because of computational issues) are the distinct number of classes that occur in either the subject, predicate, or object position, as specified in VoID-ext. These three meta-data properties cannot be calculated by streaming the data a single time. In addition, all meta-data properties whose values must be represented as partitions are excluded in order to preserve brevity for all dataset descriptions, and to maintain scalability.

Since we want the C-LOD Meta-Dataset to be maximally useful for a wide range of use cases (requirement 7), we have added several meta-data properties that do not occur in existing specifications:

1. Next to the number of distinct IRIs, blank nodes and literals (i.e., *types*), we also include the number of (possibly non-distinct) occurrences (i.e., *tokens*).
2. Existing vocabularies specify the number of properties and classes (although they do so incorrectly, see section 3). The C-LOD Meta-Dataset also includes the number of classes and properties that are *defined* in a dataset, such as `<prop> rdf:type rdf:Property`
3. Where existing dataset descriptions specify only the average of statistics such as the literal length, the C-LOD Meta-Dataset also includes the standard deviation, median, and minimum and maximum values.

Figure 1 illustrates one of the published meta-data properties: the maximum indegree of datasets. The figure illustrates our previous remark that analyzing very many datasets will inevitably include datasets with atypical properties or 'corner cases'. E.g., the dataset with the highest maximum indegree, has a value of more than 7 million. In other words, one single resource occurs in the object position of about 7 million triples, thereby strongly skewing the dataset distribution. Note, that generating the data behind this figure requires the following easy SPARQL query, illustrating the ease of use:

```
SELECT * {[] llm:degree/llm:mean ?mean}
```

Besides publishing the meta-data, and in line with requirement 5, the Meta-Dataset contains a provenance trail of how the meta-data was generated. The provenance trail includes a reference to the code that was used to generate the meta-data. For this we use a Git commit identifier in order to uniquely identify the exact version that was used. The provenance trail also includes all the steps that preceded the calculation of the meta-data:

1. Where the file was downloaded (either the original URL or the archive that contained the file).
2. When the file was downloaded (date and time).
3. Metadata on the download process, such as the status code and headers from the original HTTP reply. For archived data the applied compression techniques (possibly multiple ones) are enumerated as well.
4. Detailed metadata on the data preparation tasks performed by the LOD Laundromat in order to clean the data. This includes the number of bytes that were read (not necessarily the same as the value for `Content-Length` HTTP header) and syntax errors that were encountered (e.g., malformed syntax, unrecognized encoding, undefined prefixes).
5. The number of duplicate triples in the original dataset.
6. A reference to the online location where the cleaned (C-LOD) file is stored, and from which the C-LOD Meta-Dataset is derived.

4.2. Model

The meta-data is specified in the C-LOD Meta-Data Vocabulary[7]. Of the 26 meta-data properties that are included, 22 are linked to one or more other dataset description vocabularies. The referenced vocabularies are VoID, Bio2RDF, and VoID-ext. The C-LOD Meta-Data Vocabulary also includes meta-data about the vocabulary *itself*, such as the license (Creative Commons 3), last modification date, creators, and homepage. As such, it implements the first 4 of the 5 stars for vocabulary re-use [5]. The fifth star (re-use *by* other vocabularies) is not reached yet because the vocabulary is very recent. However, the C-LOD Meta-Data Vocabulary has been submitted to the Linked Open Vocabu-

[7] See `http://lodlaundromat.org/metrics/ontology/`

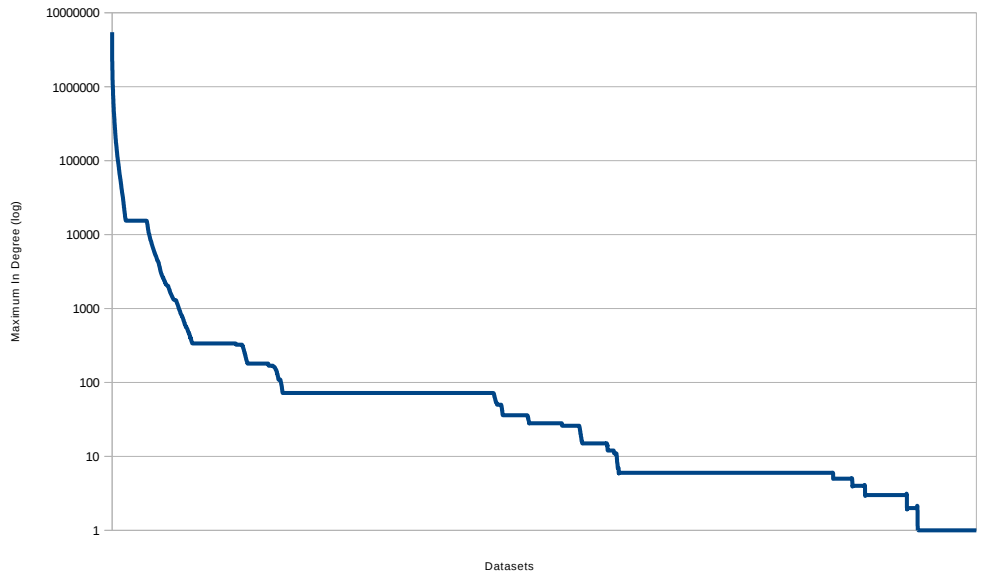

Fig. 1. Maximum indegree Distribution, log scale)

lary catalog [8], thereby hopefully supporting its re-use and findability.

The provenance information of datasets is described using the PROV-O standard [6]. As the LOD Laundromat cleaning process is part of the provenance trail, we model this part of the dataset by using separate vocabularies: Firstly, the LOD Laundromat vocabulary[9] describes the crawling and cleaning process of LOD Laundromat. Secondly, the HTTP vocabulary[10] describes HTTP status codes. Thirdly, the error ontology[11] models all exceptions and warnings, which is used by the LOD Laundromat vocabulary in order to represent errors that occur during the crawling and cleaning process. Each of these vocabularies are linked to other vocabularies as well. E.g., the HTTP vocabulary is an extension of the W3C HTTP in RDF vocabulary[12].

4.3. Dissemination

The LOD Laundromat [2] continuously crawls and analyses Linked Data dumps. In order to get a maximum coverage of the LOD Cloud, it searches both linked data catalogs and the C-LOD datasets themselves for references to datadumps. Because it does not claim to have a complete seed list that links to all LOD in the world, users have the option to manually or algorithmically add seed-points to the LOD Laundry Basket[13].

The code[14] used to generate the C-LOD Meta-Dataset runs on newly cleaned data in daily intervals. As a consequence, the dataset is regularly and automatically updated whenever new C-LOD data is crawled and cleaned by the LOD Laundromat. In line with requirement 6, each daily version of the C-LOD Meta-Dataset is made publicly available in data dump form[15], using a standardized RDF serialization. The data is also accessible via a public SPARQL endpoint[16].

5. Use Cases

The C-LOD Meta-Dataset is intended to support a wide array of non-trivial use cases. One of the first

[8] http://lov.okfn.org/
[9] http://lodlaundromat.org/ontology/
[10] http://lodlaundromat.org/http/ontology/
[11] http://lodlaundromat.org/errors/ontology/
[12] http://www.w3.org/2011/http
[13] http://lodlaundromat.org/basket/
[14] Publicly available at https://github.com/LODLaundry/LODAnalysis
[15] http://download.lodlaundromat.org/dump.nt.gz
[16] http://lodlaundromat.org/sparql

Meta-data Property	**VoID**	**Bio2RDF**	**VoID-ext**	**LOD Laundromat**
Triples	v	v	v	v
Entities	v	v	v	v
Distinct Classes	v	v	v	v
Distinct Properties	v	v	v	v
Distinct subject	v	v	v	v
Distinct objects	v	v	v	v
Distinct RDF Nodes			v	v
Distinct IRIs			v	v
IRIs				v
Distinct Blank Nodes			v	v
Blank Nodes				v
Distinct literals	v		v	v
Literals				v
Distinct URIs in subject position			v	v
Distinct Blank Nodes in subject position			v	v
Distinct URIs in object position			v	v
Distinct Blank Nodes in object position			v	v
Distinct literal data-types			v	v
Distinct literal languages			v	v
Length statistics of IRIs			v	v
Length statistics of IRIs in subject position			v	v
Length statistics of IRIs in predicate position			v	v
Length statistics of IRIs in object position			v	v
Length statistics of literals			v	v
Defined Classes				v
Defined Properties				v
Distinct classes occurring in the subject position			v	
Distinct classes occurring in the predicate position			v	
Distinct classes occurring in the object position			v	

Table 1

An overview of dataset meta-data properties, grouped by the vocabularies that define them and dataset description collections that include them. For brevity's sake, properties whose values are dataset partitions are excluded.

use cases that comes to mind is using this data to analyze and compare datasets, e.g., in order to create an overview of the state of the LOD Cloud at a given moment in time. As multiple versions of the C-LOD dataset are generated over time, the dynamics of the LOD Cloud can be studied as well.

Another use case, is the evaluation of SW algorithms. This use case combines the strength of both the collection of C-LOD datasets and the C-LOD Meta-Dataset: In contemporary SW research novel algorithms are usually evaluated against only a handful of – often the same – datasets (i.e., mainly DBpedia, Freebase, and Billion Triple Challenge). The risk of this practice is that – over time – SW algorithms will be optimized for datasets with specific distributions, but not for others. By using the C-LOD Meta-Dataset for relating datasets to their overall structural properties, and by using the cleaned C-LOD files themselves to access those datasets, evaluations in SW research can be performed on a much wider scale, leading to results that are more indicative of the *entire* LOD Cloud.

The C-LOD Meta-Dataset can also be used in order to provide extra information to SW algorithms. By using such meta-data, algorithms can fine-tune their heuristics. They can also be optimized by filtering datasets with the desired properties in an early stage,

i.e., without having to load and interpret them. An example of this is PrefLabel[17], an online service that returns a human-readable label for a given resource-denoting IRI. The index behind the PrefLabel Web service is populated streaming and analyzing C-LOD datasets for RDFS label statements in datasets. PrefLabel uses the C-LOD Meta-Dataset by pruning for datasets that do not contain RDF literals at all. This crude way of using the C-LOD Meta-Dataset already excludes 20% of all the triples that are in the C-LOD Collection today, thereby significantly optimizing the algorithm. [18]

Our last use case is about performing Big Data research on LOD. Suppose we need to calculate the PageRank of RDF graphs. Such calculations are often performed in parallel (either on a cluster or on a single server). A priori knowledge of the structural properties of datasets, such as their node degree or their overall size, can assist the algorithm in estimating the hardware costs of performing such calculations. Based on these estimates an algorithm can perform dynamic load balancing: datasets that are very large and/or have non-standard skewness are likely to consume (much) more system resources, so only a limited number of such datasets should be processed at any given time; on the other hand, datasets of small size and/or with more regular structural properties can be processed in parallel without running the risk of exhausting hardware resources.

6. Conclusion

The dataset presented in this paper offers access to a large set of uniformly represented datasets descriptions, acting as an enabler for large scale Linked Data research: finding or comparing linked datasets with certain structural properties is now as easy as executing a SPARQL query. And even better: because the dataset descriptions are linked to their uniform dataset representations, the access to the underlying data is extremely easy as well.

We are exploring the possibilities of storing snapshots of both the Meta-Dataset and the corresponding cleaned datasets, effectively creating snapshots of the state of the LOD Cloud. At this point, we consider this future work though.

Another future improvement we consider is to publish partitions of the datasets via more scalable and efficient ways than SPARQL. As explained in section 4.1, corner-cases in the LOD cloud might drastically increase some partition sizes. Therefore, an efficient and scalable method is required for hosting such partitions. We consider publishing a selection of such partitions using non-SPARQL APIs with a stronger focus on scalability and efficiency (e.g. mongoDB).

[17] `http://preflabel.org`

[18] This 10-line bash script used by PrefLabel shows how easy it is to use the C-LOD Collection together with the C-LOD Meta-Dataset to crawl the entire LOD cloud: `https://github.com/Data2Semantics/prefLabel/blob/develop/scripts/load_lodlaundromat.sh`

Acknowledgements

This work was supported by the Dutch national program COMMIT.

References

[1] Sören Auer, Jan Demter, Michael Martin, and Jens Lehmann. Lodstats–an extensible framework for high-performance dataset analytics. In *Knowledge Engineering and Knowledge Management*, pages 353–362. Springer, 2012.

[2] Wouter Beek, Laurens Rietveld, Hamid R Bazoobandi, Jan Wielemaker, and Stefan Schlobach. Lod laundromat: A uniform way of publishing other people's dirty data. In *Proceedings of the International Semantic Web Conference (ISWC)*, 2014.

[3] Carlos Buil-Aranda, Aidan Hogan, Jürgen Umbrich, and Pierre-Yves Vandenbussche. Sparql web-querying infrastructure: Ready for action? In *The Semantic Web–ISWC 2013*, pages 277–293. Springer, 2013.

[4] Alison Callahan, José Cruz-Toledo, Peter Ansell, and Michel Dumontier. Bio2rdf Release 2: Improved Coverage, Interoperability and Provenance of Life Science Linked Data . In *The Semantic Web: Semantics and Big Data*, pages 200–212. Springer, 2013.

[5] Krzysztof Janowicz, Pascal Hitzler, Benjamin Adams, Dave Kolas, and Charles Vardeman II. Five stars of linked data vocabulary use. *Semantic Web*, 5(3):173–176, 2014.

[6] Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. Prov-o: The prov ontology. *W3C Recommendation, 30th April*, 2013.

[7] Eetu Mäkelä. Aether – generating and viewing extended void statistical descriptions of rdf datasets. In *Proceedings of the ESWC 2014 demo track, Springer-Verlag*, 2014.

[8] Eyal Oren, Renaud Delbru, Michele Catasta, Richard Cyganiak, Holger Stenzhorn, and Giovanni Tummarello. Sindice.com: a Document-Oriented Lookup Index for Open Linked Data . *International Journal of Metadata, Semantics and Ontologies*, 3(1):37–52, 2008.