Assessing Quantity and Quality of Links Between Linked Data Datasets

Ciro Baron Neto Dimitris Kontokostas Sebastian Hellmann
Kay Müller Martin Brümmer
Leipzig University, AKSW/KILT, http://aksw.org/Groups/KILT
Leipzig (Germany)
(cbaron|kontokostas|hellmann|kay.mueller|bruemmer)@informatik.uni-leipzig.de

ABSTRACT

The Linked Data Web is growing and it becomes increasingly necessary to analyze the relationship between datasets to exploit its full value. LOD datasets can range from datasets with low cohesion – containing data from different Fully Qualified Domain Names (FQDN) and namespaces – to highly cohesive datasets. This paper evaluates the quantity and quality of links between distributions, datasets and ontologies categorizing and defining different types of links. We streamed and indexed 2.5 billion triples and extracted 0.5 billion links using probabilistic data structures. Our results show the analysis of datasets w.r.t. valid links, dead links, and number of namespaces described by distributions and datasets. Our results indicate that 7.9% of the links we indexed and verified are actually dead.

Keywords

Linked Open Data, Linksets, Dead Links, RDF

1. INTRODUCTION

Since the beginning of the Linked Open Data (LOD) cloud we experienced an exponential growth in the amount of published datasets.[1] A key component in the LOD cloud are the links between the different datasets. Links play an essential role, allowing users to navigate between data entries, integrate data and perform large scale inference. The quantity and quality of these links can play a crucial factor in the evolution of the data web. Previous studies[4] show that per week one out of every 200 links become inaccessible in the Internet, and in social networks 11% of the shared links might disappear[10] in less than one year. Thus, it's necessary to secure a minimum quality of links, as more and more applications build on top of aggregated datasets and erroneous links can propagate errors breaking applications.

Another frequent problem in the area of Linked Data Analysis, is to provide fast and accurate methods to detect and extract links between datasets. Existing approaches mainly rely on counting *fully qualified domain name* (FQDN) in the Linked Data space. However, this method is not accurate, since for large datasets, it's unfeasible to check if all links are truly being described in the *source* and in the *target* dataset. Crawling the Linked Data datasets providing up-to-date data, is computationally expensive and requires powerful hardware to scale up indexing with the growing cloud.

In this paper, we present a thorough analysis of the links between the datasets participating in the 2014 LOD cloud[11] and Linked Open Vocabularies[2]. The aim of this paper was to evaluate the quality of links between these knowledge bases. The analysis was conducted with the engine of LODVader[3], a real-time LOD Visualisation, Analytics and DiscovEry tool. Our novel approach based on Bloom-filters allows us to accurately measure the exact number of links between datasets and distributions, as well as identify dead and unverified links (cf. section 2) between datasets.

The remainder of this work is structured as follows: We provide a description of metadata vocabularies, link granularity and *linksets* in Section 2, followed by the methodology used details in Section 3. Section 4 describes the results of our analysis and in Section 5 we present the related works.

[1] http://lod-cloud.net/#history
[2] http://lov.okfn.org/dataset/lov/
[3] For the interface see http://svn.aksw.org/papers/2016/WWW_LODVader_DEMO/public.pdf

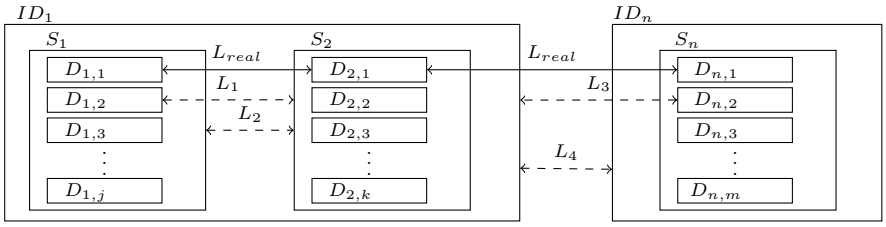

Figure 1: The full arrows (L_{real}) represent measurable links between distributions. The dotted arrows are inferred linksets between: L_1 distribution to subset, L_2 subset to subset , L_3 distribution to dataset, L_4 dataset to dataset

Finally, in Section 6 we present the future works and our conclusions.

2. BACKGROUND

2.1 Dataset Metadata Vocabularies

In order to identify which resources should be streamed and analyzed, this work relies on vocabularies such as DCAT [8], VoID [4] and DataID [2]. These vocabularies are used to represent metadata descriptions of datasets. They provide information about multiple properties of a dataset, including subsets and distributions. A subset is a distinct part of a dataset that can be differentiated for a number of reasons, such as differences in provenance, publication dates, accessibility or language[5]. Distributions describe the specific files or resources by which the datasets might be accessed or acquired[6].

2.2 Linkset Definition

Linksets are RDF descriptions of relations between datasets or distributions, represented by links. We adopted the DCAT and VoID vocabulary to describe the number of links, as well as source and target datasets. In order to clarify the definition of the existing variables for a *linkset*, a brief explanation is given.

- ID: a dataset, described by `void:Dataset` or `dcat:Dataset`;
- S_{ID}: the set of subsets, described by `void:subset` of given dataset ID
- $< s, p, o >$: the RDF triple which represents the subject s, predicate p and object o for a given relation
- d_n: the n-th distribution consisting of a set of RDF triples.
- D_{ID}: the set of distributions, described by `dcat:distributions`, of the dataset ID
- $D_{S_{ID}}$: the set of distributions of subset S of dataset ID
- $L_{d_s \to d_t}$: the set of existing links between two distributions, having d_s as source distribution and d_t as target distribution. We define that a link occurs from a distribution d_s to a distribution d_t whenever d_s contains $< s_s, p_s, o_s >$ and d_t contains $< s_t, p_t, o_t >$ where $o_s = s_t$. We then call the triple $< s_s, p_s, o_s >$ in the source distribution a link (regardless of the used property) and say that the distributions are linked with each other (cf. Section 2.4). From this definition it easily follows that *linksets* between distributions (subsets or datasets) can be aggregated in a straightforward manner. Consequently, a dataset ID_s is linked to another dataset ID_t, if a non-empty *linkset* from any distribution $D_{S_{ID_s}}$ to $D_{S_{ID_t}}$ exists.

Furthermore, we define the following notions in order to describe dead or unverified links. A dead link on the WWW is generally associated with a HTTP 404 Not Found response message. Analogously, we define "Not Found" between a distribution and a dataset:

- $NS_n(uri)$: The namespace of a URI, whereas NS_0 refers to FDQN (incl. subdomain), NS_x refers to FDQN plus the URI path of length n and NS_* refers to the FDQN plus the path until and including the last '/' or '#'. In this paper, we work with NS_* or simply NS only, although other research would be interesting.
- $SNSS(D)$ the set of $NS_*(s_t)$ for all the subjects in all distributions of dataset D.
- A *partial dead link* $< s_1, p_1, o_1 >$ between a distribution d_1 and a dataset D exists if $NS(o_1) \in SNSS(D)$ and $\nexists$ triple $t \in D \mid o_1 = s_t$. Note that this definition is based on the assumption that namespaces are unique to datasets. Given that there are several datasets with applicable namespaces, a *total dead link* or just *dead link* means that the respective object is not found as subject in any –already indexed– dataset with overlapping namespaces.
- An *unverified link* $< s_1, p_1, o_1 >$ exists if $NS(o_1)$ can not be found in any indexed dataset, i.e. there are no overlapping namespaces. As we are not investigating HTTP resolution, we have to assume bona fide that we just have not indexed the target dataset yet.

2.3 Link Granularity

The LOD cloud diagram[11] assumes as the basis for a dataset definition the *Pay-Level Domain (PLD)* [7]. It consequently only depicts inter-dataset relations as links. *LODVader* also offers visualisation and analysis of intra-dataset relationships, for example between subsets and distributions, featuring a higher *link granularity*. Figure 1 shows an overview of links at different levels of granularity regarding a *linkset* representation. Datasets are represented by ID_n, subsets are represented by S_n and distributions are represented by D_n. L_{real} is a *linkset* containing links between two distri-

[4] `http://www.w3.org/TR/void/`
[5] `http://www.w3.org/TR/void/#subset`
[6] `http://www.w3.org/TR/vocab-dcat/#class-distribution`

butions which are measured on the intersection of subjects and objects (cf. Section 2.2). The *linksets* L_1 to L_4 can be generated by calculating the union of the *linksets* between all distributions of the respective subsets and datasets.

2.4 Linking Predicates

Common approaches for linking analysis rely on the inspection of the predicates. `owl:sameAs` has well-defined formal semantics and is the predicate which is closest to traditional deduplication. For record linkage or object reconciliation in the database area, counting `owl:sameAs` links exclusively provides a very limited view of the Web of Data and does not provide a reliable model [6].

Several other properties have been proposed with `rdfs:seeAlso` and `skos: { exact | close | broad | narrow | related} Match` being the most common. In our work, we are tolerant and consider all predicates for linking. While for crawling link direction is important – although DBpedia is the largest authority [11], no backlinks are included – we argue that linking properties is often either symmetric (and highly unlikely to be asymmetric) or it is feasible to assume that an inverse property exists or could be easily created, i.e., following a `birthplace`↔`isBirthplaceOf` pattern or simply `birthplace`$^{-1}$.

To the best of our knowledge, we have not encountered predicates expressing negative links yet (i.e. `notLinkedTo`).

Vocabulary Links. another aspect of linking properties that is often neglected are links to vocabularies and links between vocabularies. Especially, the linkage via `rdf:type` has not yet been visualized in a cloud diagram and is often not included in link analysis.

3. METHODOLOGY

We parsed description files from Linked Open Vocabularies[7], DBpedia datasets and from the LOD cloud searching for instances of `dcat:Distribution`, henceforth called *source distribution*. The application then fetches the `dcat:downloadURL` or `void:dataDump` object. Before the download of the *source distribution* is started, it is checked whether the dataset has already been imported into the system. If the dataset is known, the system reads the *Last-Modified date* and *Content-Length* in the HTTP header to verify whether the dataset has not been changed. If there are modifications, the old data is moved to an archive, in order to use it for versioning reasons. Once the streaming starts, we detect the serialization type, possibly decompress the stream and parse the RDF triples. It's important to emphasize that since LODVader is publicly available, more and more datasets are added and analyzed.

The process of Link Discovery is made on the fly for each distributions streamed. For every triple, the *Linking Analytics* modules discards the predicate and takes only the subject and the object as input ($< s, o >$). If the object is a literal or a blank node the tuple is discarded. As a final filtering step, we reject tuples with malformed IRIs. The tuples that pass the filtering step, enter a processing pipeline:

1. **Tuple splitting**. *subjects* and *objects* of each tuple are separated and saved in two queues. The queues contain resources which will be compared with Bloom filters (BFs).
2. **BF Fetching**. we extract the namespace of each resource to compare and assign the resource to a respective BF which will represent a *target distribution*. For every namespace we encounter, we fetch all the existing BFs that are processed and stored in a cache memory.
3. **Link Extraction**. objects and subjects of the *source distribution* are compared with the in-memory BFs of the *target distributions*. If an *object* of the *source distribution* exists in the BF of the target distribution as a *subject* we count one link between the *source distribution* and the *target distribution*. If the opposite way happens, i.e. if *subject* of the *source distribution* exists in the BF of the target distribution as an *object* we count one link between the *target distribution* and the *source distribution*. The non-existence of link between a *source distribution* and a *target dataset* is counted as a dead link between the *source distribution* and the *target dataset*.

At the end of the pipeline two sets of BFs are created. A set containing all subjects and a second set containing all objects of the *source distribution*. These BFs will represent the current distribution and might be used later when other *sources distributions* are streamed.

It is important to stress that, although our model reads and retrieves RDF data, it does not store any RDF. Our implementation creates RDF on the fly reading documents from MongoDB and using Apache Jena to create RDF models. All BF stored have the same size (each BF describes 5000 resources), making the time to query any resource from any distribution be quasi-linear time complexity. For big distributions with more than 5000 triples, multiple BFs are created. In addition, the BFs are not stored directly to the file system, but using GridFS[8] to manage the BF files. A more detailed documentation in regard to the implementation can be found on the LODVader GitHub[9] repository.

4. RESULTS

In order to make a general analysis of quantity and quality of Linked Data datasets, we streamed all datasets found in the metadata description file of the The Linking Open Data cloud diagram 2014[10], the DBpedia Core[11] distributions and all vocabularies found on Linked Open Vocabularies[12]. At the time of writing, we discovered[13] 185 million verified links (out of 0.5 billion links in total) among 1408 datasets and 395 vocabularies, totalizing more than 2.5 billion triples. These numbers grow, since more users start to provide good metadata and it's possible for users to submit their datasets to our analysis.

[7]http://lov.okfn.org/dataset/lov/

[8]https://docs.mongodb.org/manual/core/gridfs/
[9]https://github.com/AKSW/LODVader
[10]http://data.dws.informatik.uni-mannheim.de/lodcloud/2014/ISWC-RDB/
[11]http://downloads.dbpedia.org/current/core/
[12]http://lov.okfn.org/dataset/lov/
[13]http://lodvader.aksw.org/#/stats

Name	$\#NS_*$	%
Educational programs - SISVU	6	99.97%
statistics.data.gov.uk	3	99.96%
Farmers Markets Geo. Data (U.S.)	8	99.95%
VIVO Weill Cornell Medical College	3	99.82%
VIVO WUSTL	5	99.62%
...	...	...
...	...	...
eagle-i @ Dartmouth College	13	72.96%
TaxonConcept Knowledge Base	9	59.20%
eagle-i @ Montana State University	12	47.63%
The Living LOD Cloud	741	31.91%
Ontos News Portal	472	10.35%

Table 1: Distinct namespaces per dataset and percentage of predominant namespace

Target Dataset	Indegree	Links
DBpedia Core	38	142,951,692
eagle-i @ University of Hawaii	44	573,835
eagle-i @ University of Texas	43	389,449
TaxonConcept Knowledge Base	27	143,668
The Living LOD Cloud	79	121,770

Table 2: Highest number of related datasets (indegree) pointing to the target dataset

Our result analysis consists of three steps. First, in order to know whether a dataset is suitable or not to describe certain resource (e.g., *subjects* or *objects*), we extracted all namespaces with their respective proportion on the datasets. Following, we calculated the number of indegree and outdegree per datasets, and finally, we calculated the indegree and outdegree of dead links among datasets. Our metric for indegree and outdegree are the number of datasets which contains one or more link to or from the current dataset.

Several datasets describe a single namespace, however more than 70% of datasets describes two or more. Table 1 shows datasets with the biggest and smallest proportions of described namespaces. The column "$\#\ NS_*$" contains the number of distinct namespaces for the dataset, and the last column shows the proportion of the predominant namespace. The top 5 rows show datasets with highly predominant namespaces, and the last 5 rows show the datasets with completely mixed namespaces.

Table 2 and Table 3 show the top 5 datasets in terms of number of indegree and outdegree. DBpedia is the most linked dataset in both cases followed by eagle-i datasets which describes biomedical data and is heavily interconnected.

Table 4 and Table 5 shows the top 5 datasets with dead indegree links, and top 5 datasets with dead outdegree links. Dead indegree means that external datasets link to non-existing resources of a dataset. Dead outdegree refers to dataset that link to external dead links. The in and out degree is aggregated at the dataset level and the *links* provides the total number of dead links.

Source Dataset	Outdegree	Links
DBpedia Core	39	142,963,603
eagle-i @ Dartmouth College	165	416,858
eagle-i @ Uni. Alaska	144	386,797
eagle-i @ Charles R. Drew Uni.	140	320,099
TaxonConcept Knowledge Base	241	241,817

Table 3: Source datasets that point to the highest number of related datasets (outdegree)

Target Dataset	Indegree	Links
The Living LOD Cloud	89	10,315,736
TaxonConcept Knowledge Base	81	10,001,141
VIVO Cornell	58	226,740
eagle-i @ Jackson State University	42	195,298
Traditional Korean Medicine Ont.	68	134,386

Table 4: Highest Indegree Dead links

Source Dataset	Outdegree	Links
eagle-i @ Ponce - School of Medicine	13	61,402
Rådata nå!	13	49,861
eagle-i @ Vanderbilt University	21	42,104
I-Choose	41	9,435
The Cancer Genome Atlas	1	8,428

Table 5: Highest Outdegree Dead links

Target Dataset	Indegree	Links
The Media RDF Vocabulary	75	217
Document Availability Information Ont	36	190
VIVO Core Ontology	4	166
An Ontology for vCards	4	57
Conversion Ontology	10	55

Table 6: Highest Indegree Dead links (ontologies)

Table 6 depicts the top 5 ontologies with the highest number of dead links indegree. This table actually reveals accidental or intentional ontology IRI typos. We also measure outdegree of dead links from ontologies, but didn't include a table. These numbers are very small compared to Table 3, however, linking to a misspelled or non-existing external class via `owl:equivalentClass` has a much higher impact on the overall quality of the Web of Data.

Finally, Figure 2 provides an overview of the total correct links, dead links and unverified links. In total, we have found 302,855,189 unverified links, 12,430,800 dead links and 172,254,731 links. The large number of unverified links is due the fact that our coverage is not so broad, and it's still getting wider since new datasets are added. It is worth noting though that 7.9% of the verified links are dead links.

5. RELATED WORK

Most LD (link discovery) frameworks can only determine links based on `owl:sameAs` or equivalent instances. However, RDF-AI[5] is a framework which takes two datasets as input, and as outcome generates a new dataset where the content is a list of correspondences between equivalent resources of the input datasets. The system is composed of five modules

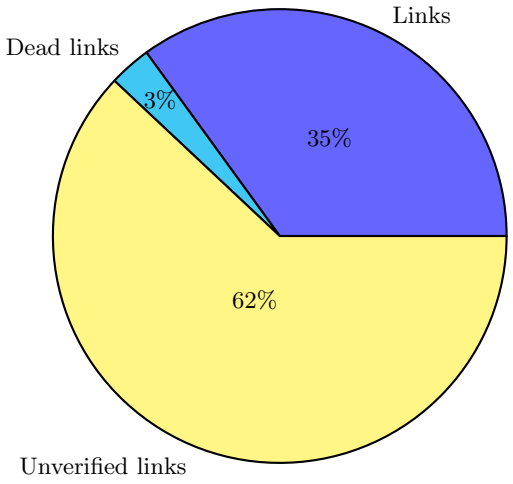

Figure 2: Links, Dead links and Unverified links

which allows pre-process, match, fusion, inter-link and post-process RDF datasets.

Due to strong growth of the LOD cloud it is obvious that there is a demand for LOD cloud analytical frameworks. Some statistical information can be found together with the LOD cloud diagram [11] [3]. Unfortunately the statistical information are also static.

Another good example is Aether [9]. It supplies the user with many different statistical information for datasets when supplied with a SPARQL endpoint address. It is even possible to compare different SPARQL endpoints, which can be useful if two different endpoints should be analyzed. Although this framework supplies the user with great statistical information and pie charts, it is only developed for comparing the content between two SPARQL endpoints.

LOD-Laundromat[1] provides an uniform way to publish and clean datasets. Different statistical data is published, like duplicated triples, amount of triples, dataset size and other. The LOD-Laundromat contains over 38 billion triples, however the issue is that they do not provide metadata regarding dataset labels, name or title, making the whole graph visualization a hard task.

6. CONCLUSIONS AND FUTURE WORK

This paper classified and evaluated links among more than 1,200 datasets w.r.t. dataset indegree and outdegree for different types of links. We discovered a total of 0.5 billion links out of which 12.5M were dead and we could not verify 302M links. This suggests that around 7.9% of the verified LOD links we indexed are dead. This number is based on current coverage of indexed datasets of our analysis. Indexing new datasets can raise this number (if more dead links are discovered) as well as lower it (if a dataset is indexed that contains link targets). However, we already invested a lot of effort into discovering as many datasets as possible and assume that an average linked data consumer would not go to such lengths to retrieve data.

In order to expand the coverage of our analysis, we expect to work in collaboration with other approaches such as LOD-Laundromat[1]. We believe that at least the amount of unverified links might be reduced as more dataset will be added.

An area we would like to reasearch on is to identify authoritative namespaces for datasets. This would make it easier to identify if a resource is described in an authoritative dataset or a dataset hijacks a namespace. This could provide ways to further analyze the quality of links and would also help to define best practices based on de-facto linking.

Acknowledgement. This paper's research activities were funded by grants from the FP7 & H2020 EU projects LIDER (GA-610782) and ALIGNED (GA 644055), FREME (GA-644771), Smart Data Web (GA-01MD15010B) and CAPES foundation - Ministry of Education of Brazil (13204/13-0).

7. REFERENCES

[1] W. Beek, L. Rietveld, H. Bazoobandi, J. Wielemaker, and S. Schlobach. Lod laundromat: A uniform way of publishing other people's dirty data. In *ISWC 2014*, Lecture Notes in Computer Science, pages 213–228. Springer International Publishing, 2014.

[2] M. Brümmer, C. Baron, I. Ermilov, M. Freudenberg, D. Kontokostas, and S. Hellmann. DataID: Towards Semantically Rich Metadata for Complex Datasets. In *Proceedings of the 10th International Conference on Semantic Systems*, SEM '14, pages 84–91. ACM, 2014.

[3] A. J. Chris Bizer and R. Cyganiak. State of the lod cloud., 2011.

[4] D. Fetterly, M. Manasse, M. Najork, and J. Wiener. A large-scale study of the evolution of web pages. In *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pages 669–678, New York, NY, USA, 2003. ACM.

[5] C. Z. Francois Scharffe, Yanbin Liu. Rdf-ai: an architecture for rdf datasets matching, fusion and interlink. *IJCAI*, 2009.

[6] H. Halpin, P. J. Hayes, J. P. McCusker, D. L. McGuinness, and H. S. Thompson. When OWL: sameAs Isn't the Same: An Analysis of Identity in Linked Data. In *ISWC*, pages 305–320. Springer, 2010.

[7] O. Lehmberg, R. Meusel, and C. Bizer. Graph Structure in the Web: Aggregated by Pay-level Domain. In *Proceedings of the 2014 ACM Conference on Web Science*, WebSci '14, pages 119–128, New York, NY, USA, 2014. ACM.

[8] F. Maali and J. Erickson. Data Catalog Vocabulary (DCAT). W3C recommendation, W3C, Jan. 2014.

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

[10] H. SalahEldeen and M. L. Nelson. Losing my revolution: How many resources shared on social media have been lost? *CoRR*, abs/1209.3026, 2012.

[11] M. Schmachtenberg, C. Bizer, and H. Paulheim. Adoption of the Linked Data Best Practices in Different Topical Domains. In *ISWC 2014*, pages 245–260, 2014.