Profiling the Web of Data

Anja Jentzsch
supervised by Prof. Dr. Felix Naumann
anja.jentzsch@hpi.uni-potsdam.de

Hasso-Plattner-Institute, Potsdam, Germany

Abstract. The Web of Data contains a large number of openly-available datasets covering a wide variety of topics. In order to benefit from this massive amount of open data such external datasets must be analyzed and understood already at the basic level of data types, constraints, value patterns, etc.
For Linked Datasets such meta information is currently very limited or not available at all. Data profiling techniques are needed to compute respective statistics and meta information. However, current state of the art approaches can either not be applied to Linked Data, or exhibit considerable performance problems. This paper presents my doctoral research which tackles these problems.

1 Problem Statement

Over the past years, an increasingly large number of data sources has been published as part of the Web of Data[1]. At the time of writing the Web of Data comprised already roughly 1,000 datasets totaling more than 82 billion triples[2], including prominent examples, such as DBpedia, YAGO, and DBLP. Furthermore, more than 17 billion triples are available as RDFa, Microdata and Microformats in HTML pages[3]. This trend, together with the inherent heterogeneity of Linked Datasets and their schemata, makes it increasingly time-consuming to find and understand datasets that are relevant for integration. Metadata gives consumers of the data clarity about the content and variety of a dataset and the terms under which it can be reused, thus encouraging its reuse.

A Linked Dataset is represented in the Resource Description Framework (RDF). In comparison to other data models, e.g., the relational model, RDF lacks explicit schema information that precisely defines the types of entities and their attributes. Therefore, many datasets provide ontologies that categorize entities and define data types and semantics of properties. However, ontology information is not always available or may be incomplete. Furthermore, Linked Datasets are often inconsistent and lack even basic metadata. Algorithms and tools are needed that profile the dataset to retrieve relevant and interesting metadata analyzing the entire dataset.

[1] The Linked Open Data Cloud nicely visualizes this trend: http://lod-cloud.net
[2] http://datahub.io/dataset?tags=lod
[3] http://webdatacommons.org

Data profiling is an umbrella term for methods that compute metadata for describing datasets. Traditional data profiling tools for relational databases have a wide range of features ranging from the computation of cardinalities, such as the number of values in a column, to the calculation of inclusion dependencies; they determine value patterns, gather information on used data types, determine unique column combinations, and find keys.

Use cases for data profiling can be found in various areas concerned with data processing and data management [12]:

Query optimization is concerned with finding optimal execution plans for database queries. Cardinalities and value histograms can help to estimate the costs of such execution plans. Such metadata can also be used in the area of Linked Data, e.g., for optimizing SPARQL queries.

Data cleansing can benefit from discovered value patterns. Violations of detected patterns can reveal data errors, and respective statistics help measure and monitor the quality of a dataset. For Linked Data, data profiling techniques help validate datasets against vocabularies and schema properties.

Data integration is often hindered by the lack of information on new datasets. Data profiling metrics reveal information on, e.g., size, schema, semantics, and dependencies of unknown datasets. This is a highly relevant use case for Linked Data, because for many openly available datasets only little information is available.

Schema induction: Raw data, e.g., data gathered during scientific experiments, often does not have a known schema at first; data profiling techniques need to determine adequate schemata, which are required before data can be inserted into a traditional DBMS. For the field of Linked Data, this applies when working with datasets that have no dereferencable vocabulary. Data profiling can help induce a schema from the data, which then can be used to find a matching vocabulary or create a new one.

Data Mining: Finally, data profiling is an essential preprocessing step to almost any statistical analysis or data mining task. While data profiling focuses on gathering structural metadata about a dataset, data mining is usually more concerned with gaining new insights about data.

2 Relevancy

There are many commercial tools, such as IBM's Information Analyzer, Microsoft's SQL Server Integration Services (SSIS), or others for profiling relational datasets. However these tool were designed to profile relational data. Linked Data has a very different nature and calls for specific profiling and mining techniques.

Finding information about Linked Datasets is an open issue on the constantly growing Web of Data due to the use cases mentioned above. While most of the Linked Datasets are listed in registries as for instance at the Data Hub (`datahub.io`), these registries usually are manually curated, and thus incomplete or outdated. Furthermore, existing means and standards for describing datasets

are often limited in their depth of information. VoiDand Semantic Sitemapscover basic details of a dataset, but do not cover detailed information on the dataset's content, such as their main classes or number of entities. More detailed descriptions, e.g., information on a dataset's RDF graph structure, topics etc., is usually not available. Data profiling techniques can help to fulfil the need for information about, e.g., classes and property types, value distributions, or entity interlinking.

3 Related Work

While many general tools and algorithms already exist for data profiling, most of them cannot be used for graph datasets, because they assume a relational data structure, a well-defined schema, or simply cannot deal with very large datasets. Nonetheless, some Linked Data profiling tools already exist. Most of them focus on solving specific use cases instead of data profiling in general.

One relevant use case is schema induction, because the lack of a fixed and well-defined schema is a common problem with Linked Datasets. One example for this field of research is the ExpLOD tool [9]. ExpLOD creates summaries for RDF graphs based on class and property usage as well as statistics on the interlinking between datasets based on `owl:sameAs` links.

Li describes a tool that can induce the actual schema of an RDF dataset [11]. It gathers schema-relevant statistics like cardinalities for class and property usage, and presents the induced schema in a UML-based visualization. Its implementation is based on the execution of SPARQL queries against a local database. Like ExpLOD, the approach is not parallelized. Both solutions still take approximately 10h to process a 10 million triples dataset with 13 classes and 90 properties. These results illustrate that performance is a common problem with large Linked Datasets.

An example for the query optimization use-case is presented in [10]. The authors present RDFStats, which uses Jena's SPARQL processor to collect statistics on Linked Datasets. These statistics include histograms for subjects (URIs, blank nodes) and histograms for properties and associated ranges.

Others have worked more generally on generating statistics that describe datasets on the Web of Data and thereby help understanding them. LODStats computes statistical information for datasets from the Data Hub [2]. It calculates 32 simple statistical criteria, e.g., cardinalities for different schema elements and types of literal values (e.g., languages, value data types).

In [4] the authors automatically create VoID descriptions for large datasets using MapReduce. They manage to profile the BTC2010 dataset in about an hour on Amazon's EC2 cloud, showing that parallelization can be an effective approach to improve runtime when profiling large amounts of data.

Finally, the ProLOD++ tool allows to navigate an RDF dataset via an automatically computed hierarchical clustering [5] and along its ontology class tree [1]. Data profiling tasks are performed on each cluster or class dynamically and independently to improve efficiency.

4 Challenges

This section describes selected challenges that I identified as specific to profiling Linked Data and web data, as opposed to profiling relational tables.

Profiling along hierarchies

Vocabularies define classes and their relationships. Ontology classes usually are arranged in a taxonomic (subclass–superclass) hierarchy. While the Web of Data spans a global distributed data graph, its ontology classes build a tree with `owl:Thing` as its root. Analyzing datasets along the vocabulary-defined taxonomic hierarchies yield further insights, such as the data distribution at different hierarchy levels, or possible mappings betweens vocabularies or datasets.

Keys are clearly of vital importance to many applications in order to uniquely identify individuals of a given class by values of (a set of) key properties. In OWL 2 a collection of properties can be assigned as a key to a class using the `owl:hasKey` statement [8].

Nevertheless it has not yet fully arrived on the Web of Data: only one Linked Dataset uses `owl:hasKey` [7]. Thus, actually analyzing and profiling Linked Datasets requires manual, time consuming inspection or the help of tools.

Many languages have a so-called "unique names" assumption. On the web, such an assumption is not possible as real-world entities can be referred to with different URI references.

Heterogeneity

A common practice in the Linked Data community is to reuse terms from widely deployed vocabularies whenever possible, in order to increase homogeneity of descriptions and, consequently, easing the understanding of these descriptions. There are at least 416 different vocabularies to be found on the Web of Data[4]. Some datasets, however, also exist without any defined or dereferencable vocabularies. And even if common vocabularies are used, there is no guarantee that the specifications and constraints are followed correctly.

Nearly all datasets on the Web of Data use terms from the W3C base vocabularies RDF, RDF Schema, and OWL. In addition, 191 (64.75 %) of the 295 datasets in the Linked Open Data Cloud Catalogue use terms from other widely deployed vocabularies [3].

As Linked Datasets cover a wide variety of topics, widely deployed vocabularies that cover all aspects of these topics may not exist yet. Thus, data providers often define proprietary terms that are used in addition to terms from widely deployed vocabularies in order to cover the more specific aspects and to publish the complete content of a dataset on the Web. Currently 190 (64.41 %) out of the 295 datasets use proprietary vocabulary terms with 83.68 % making the term URIs dereferenceable.

Topical profiling

The Web of Data covers not only a wide range of topics, it also contains a number of topically overlapping data sources. Since it provides for data-

[4] `http://lov.okfn.org/`

coexistence, everyone can publish data to it, express their view on things, and use the vocabularies of their choice. Integrating topically relevant datasets requires knowledge on the datasets' content and structure.

The State of the LOD Cloud document **??** gives an overview of the Linked Datasets for each topical domain but there is no fine-grained topical clustering for Linked Datasets. With 504 million inter-dataset links the Web of Data is highly interlinked; 1.6% of all triples are links stating the relationship between the real-world entities in different datasets. Thus a huge topical overlap amongst the datasets is given.

Large scale profiling

With more than 82 billion triples distributed among roughly 1,000 Linked Datasets and more than 17 billion triples available as RDFa, Microdata and Microformats, the need for efficient profiling methods and tools is apparent.

The runtime of profiling tasks as presented in Sec. 7 takes up to hours, e.g., for determining property co-occurrences [6]. Profiling tasks often have the same preprocessing steps, e.g., filtering or grouping the dataset. Thus there is a large incentive and potential to optimize the execution of multiple scripts.

5 Research Questions

The main question in my doctoral research is:

What are the challenges that are specific to profiling Linked Data and web data, as opposed to profiling relational tables?

After identifying four selected challenges, the following questions arise:

Profiling along hierarchies *Does analyzing Linked Datasets along the vocabulary-defined taxonomic hierarchies, such as the data distribution at different hierarchy levels, yield further insights?*

Heterogeneity *How does profiling help analyzing the heterogeneity on the Web of Data?*

Topical profiling *How can topical clusterings for unknown datasets on the constantly growing Web of Data be derived efficently?*

Large scale profiling *How can these huge amounts of Linked Data be profiled efficiently?*

6 Approach

My approach to address the research questions is to tackle each of the identified challenges. The main goal is to reuse existing profiling techniques and adapt them to the Linked Data world.

This section presents possible and if available developed solutions by me to the presented challenges.

Profiling along hierarchies

One example of profiling tasks along the class hierarchy is determining the *uniqueness* of properties as well as the unique property combinations, which can bring insights into the property distribution inside the dataset. It allows for finding relevant (key-candidate) properties for each level in the class hierarchy and see if the relevance is increasing or decreasing along hierarchy.

As I have found, due to the sparsity on the Web of Data, usually neither full key-candidates of properties nor unique property combinations can be retrieved using traditional techniques. Thus I defined the concept of *keyness* as the Harmonic Mean of uniqueness and density of a property[5], allowing to find potential key candidates.

Heterogeneity

Data profiling can be used to provide metadata describing the characteristics of a dataset, for instance its topic and more detailed statistics, like the main classes and properties. Furthermore, data profiling can not only determine the usage of vocabularies but also the help understanding and reusing existing vocabularies. Additionally, it can assist when mapping vocabulary terms.

Topical profiling

The first profiling task is, of course, to discover (and possibly label) these topical clusters. The discovery of which topics an unknown dataset is even about, is already a very helpful insight. Next, any profiling task can be executed on data of a particular topic and compared against the metadata of other topics.

Large scale profiling

The runtime of the profiling tasks takes up to hours already on 1 million triples, e.g., for determining property co-occurrences [6]. A number of different approaches can be chosen when trying to optimize the execution time of algorithms dealing with RDF data in general and data profiling tasks in particular.
Algorithmic optimization: Profiling tasks that have high computational complexity cannot be computed naïvely, e.g., it is infeasible to detect property co-occurrence by considering all possible property combinations. Such metrics require innovative algorithms for efficiently computing the targeted result. If such an algorithm can not be found, approximation techniques (e.g., sampling) may be required. Because these algorithms are often highly specialized for a specific profiling task, they usually do not benefit other tasks.
Parallelization: When dealing with large datasets, a good approach for improving performance is to perform calculations in parallel when possible [12]. This can be done on different levels: dataset, profiling run, profiling task and triples. Cluster-based parallelization based on MapReduce is a reasonable choice when working with Linked Data.
Multi-Query Optimization: A data profiling run usually consists of a number of different tasks, which all have to be computed on the same dataset. Depending on the set of data profiling tasks, different tasks may require the same prepro-

[5] We define the *uniqueness* of a property as the number of unique values per number of total values for a given property; and the *density* of a property as the ratio of non-NULL values to the number of entities.

cessing steps, or perform similar computation steps. Overall execution time can be reduced by avoiding duplicate computations. Similar computation steps may be interweaved to reduce runtime and I/O costs. If different tasks require similar intermediate results, these can be stored in materialized views.

7 Preliminary Results

Initially, I have defined a set of 56 useful data profiling tasks along various groupings to profile Linked Datasets. The have been implemented as Apache Pig scripts and are available online[6].
Furthermore, I illustrated the Web of Data's diversity with the results for four different Linked Datasets [6].

Profiling along hierarchies

When analyzing the uniqueness in the class hierarchy for DBpedia, I found that there are properties that become more specific by class level, thus their uniqueness gets higher for subclasses. For instance, `dbpedia:team` becomes more unique for athletes than it is for all persons. I also found properties that are generic, their uniqueness stays constant throughout the class hierarchy. For instance, `dbpedia:birthDate` is not specific to persons or their subclasses.

Furthermore, I have defined the concept of *keyness* of the property to gap the sparsity on the Web of Data and thus the possibility to find potential key candidates where traditional approaches fail.

Large scale profiling

We have addressed the different approaches to improve Linked Data profiling performance and not only developed LODOP, a system for executing, benchmarking and optimizing Linked Data profiling scripts on Hadoop but also developed and evaluated 3 multi-query optimization rules [6]. We experimentally demonstrated that they achieve their respective goals of optimizing the amount of MapReduce jobs or the amount of data materialized between jobs, thus reducing the profiling tasks runtimes by 70%.

8 Evaluation Plan

For the evaluation, there are three main lines of interest.

Metadata The main goal is to provide comprehensive dataset metadata that helps analyzing the datasets. The metadata can be evaluated on quantity and quality wrt existing metadata on the Data Hub, VoiD and Semantic Sitemaps.

Usability Tools and techniques should have a high usability in terms of results being presented in both human and machine readable ways to achieve better decision making when working with datasets.

Performance evaluation Various aspects of the developed tools should be tested for performance, especially the for huge amounts of data as it is present on the Web of Data.

[6] `http://github.com/bforchhammer/lodop/`

9 Reflections and Conclusion

The main difference in my approach with existing work on Linked Data profiling is to address the shortcomings mentioned in Sec. 3, in particular gathering comprehensive metadata in an efficient way. Within my research I am building on existing profiling techniques for relational data and adapting them according to the different nature of Linked Datasets.

This paper has presented the outline and preliminary results of my doctoral research, in which I am focussing on profiling the Web of Data.

So far I have specified and implemented a comprehensive set of Linked Data profiling tasks and illustrated the Web of Data's diversity with the results for four different Linked Datasets. Furthermore I introduced three common techniques for improving performance of Linked Data profiling and implemented three multi-query optimization rules, reducing profiling taskruntimes by 70%.

References

1. Z. Abedjan, T. Grütze, A. Jentzsch, and F. Naumann. Mining and profiling RDF data with ProLOD++. In *Proceedings of the International Conference on Data Engineering (ICDE)*, 2014. Demo.
2. S. Auer, J. Demter, M. Martin, and J. Lehmann. LODStats – an extensible framework for high-performance dataset analytics. In *Proceedings of the Int. Conf. on Knowledge Engineering and Knowledge Management (EKAW)*, 2012.
3. C. Bizer, A. Jentzsch, and R. Cyganiak. State of the LOD Cloud, 2011.
4. C. Böhm, J. Lorey, and F. Naumann. Creating VoiD descriptions for web-scale data. *Journal of Web Semantics*, 9(3):339–345, 2011.
5. C. Böhm, F. Naumann, Z. Abedjan, D. Fenz, T. Grütze, D. Hefenbrock, M. Pohl, and D. Sonnabend. Profiling Linked Open Data with ProLOD. In *Proceedings of the International Workshop on New Trends in Information Integration (NTII)*, 2010.
6. B. Forchhammer, A. Jentzsch, and F. Naumann. LODOP - Multi-Query Optimization for Linked Data Profiling Queries. In *ESWC Workshop on Profiling & Federated Search for Linked Data (PROFILES)*, 2014.
7. B. Glimm, A. Hogan, M. Krötzsch, and A. Polleres. OWL: Yet to arrive on the Web of Data? In *WWW Workshop on Linked Data on the Web (LDOW)*, 2012.
8. P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, editors. *OWL 2 Web Ontology Language: Primer*. W3C Recommendation, 2009.
9. S. Khatchadourian and M. P. Consens. ExpLOD: Summary-based exploration of interlinking and RDF usage in the linked open data cloud. In *Proceedings of the Extended Semantic Web Conference (ESWC)*, Heraklion, Greece, 2010.
10. A. Langegger and W. Wöß. RDFStats – an extensible RDF statistics generator and library. In *Proceedings of the International Workshop on Database and Expert Systems Applications (DEXA)*, pages 79–83, Los Alamitos, CA, USA, 2009.
11. H. Li. Data Profiling for Semantic Web Data. In *Proceedings of the International Conference on Web Information Systems and Mining (WISM)*, 2012.
12. F. Naumann. Data profiling revisited. *SIGMOD Record*, 42(4), 2013.