Distributed and Parallel Databases manuscript No.
(will be inserted by the editor)

Analysis of Schema Structures in the Linked Open Data Graph based on Unique Subject URIs, Pay-level Domains, and Vocabulary Usage

Thomas Gottron · Malte Knauf · Ansgar Scherp

Received: date / Accepted: date

Abstract The Linked Open Data (LOD) graph represents a web-scale distributed knowledge graph interlinking information about entities across various domains. A core concept is the lack of pre-defined schema which actually allows for flexibly modelling data from all kinds of domains. However, Linked Data does exhibit schema information in a twofold way: by explicitly attaching RDF types to the entities and implicitly by using domain-specific properties to describe the entities. In this paper, we present and apply different techniques for investigating the schematic information encoded in the LOD graph at different levels of granularity. We investigate different information theoretic properties of so-called Unique Subject URIs (USUs) and measure the correlation between the properties and types that can be observed for USUs on a large-scale semantic graph data set. Our analysis provides insights into the information encoded in the different schema characteristics. Two major findings are that implicit schema information is far more discriminative and that applications involving schema information based on either types or properties alone will only capture

Thomas Gottron
WeST – Institute for Web Science and Technologies
University of Koblenz-Landau
56070 Koblenz, Germany
Tel.: +49-261-2872862
Fax: +49-261-2871002862
E-mail: gottron@uni-koblenz.de

Malte Knauf
WeST – Institute for Web Science and Technologies
University of Koblenz-Landau
56070 Koblenz, Germany
E-mail: mknauf@uni-koblenz.de

Ansgar Scherp
Kiel University, 24118 Kiel, Germany
Leibniz Information Centre for Economics (ZBW), 24105 Kiel, Germany
Tel.: +49-431-8814-1
Fax: +49-431-8814-520
E-mail: mail@ansgarscherp.net

between 63.5% and 88.1% of the schema information contained in the data. As the level of discrimination depends on how data providers model and publish their data, we have conducted in a second step an investigation based on pay-level domains (PLDs) as well as the semantic level of vocabularies. Overall, we observe that most data providers combine up to 10 vocabularies to model their data and that every fifth PLD uses a highly structured schema.

Keywords Linked Open Data · Schema Analysis · Information · Entropy

CR Subject Classification Information Systems · RDF

1 Introduction

Since its advent in 2007, the Linked Open Data (LOD) movement fosters Tim-Berner Lee's vision of a web where data is published from various sources and interlinked to form a huge knowledge graph. This web of data is also organized in a decentralized and distributed fashion, just like the web of documents. Everyone can contribute knowledge in the form of publishing semantic data and connecting it with the existing graph by establishing links to other data sets. The technological basis for Linked Data roots back to the foundations of the Web as well: URIs, the HTTP protocol and an open format for data exchange. This is summarized to what is nowadays referred to as the four Linked Data principles[1]: (a) use URIs for identifying *entities* (i. e., physical objects as well as intangible objects), (b) allow users to look up entities by using HTTP URIs, (c) when dereferencing a URI (i. e., looking up an entity), provide useful information in a standardized way using the Resource Description Framework (RDF)[2], (d) provide linkage between URIs such that users can discover more entities and information about them. While from a technological perspective the concept of Linked Data is quite simple, this simplicity is probably one of the key success factors of the Linked Open Data movement today. In recent years, both the number of independent data contributors as well as the sheer volume of available Linked Data on the web has increased tremendously [4,17]. Major industries (e. g., the New York Times, BBC, Facebook, and Google), academia (e. g., the DBPedia project, DBLP, etc.), and governmental institutions (e. g., UK and US government agencies) have joined the movement and provide interlinked data publicly available on the web. The result is the so called *LOD cloud*[3]: the entirety of all the data published on the web following the Linked Data principles. Effectively, this LOD cloud represents a huge, distributed, semantic graph on the web.

The data on this web-scale knowledge graph does not follow any particular schema structure. Data publishers are free to assign any number of conceptual types to the entities they want to describe. They can also describe and annotate the entities with properties as they see it necessary. The properties and types are provided by vocabularies specified in the RDF syntax. Essentially, a vocabulary defines types and

[1] `http://www.w3.org/DesignIssues/LinkedData.html`, accessed: 23 March, 2013
[2] `http://www.w3.org/TR/rdf-syntax-grammar/`, accessed: 23 March, 2013
[3] `http://www.lod-cloud.net/`, accessed: 23 March, 2013

properties for entities in a certain domain. There are numerous vocabularies providing types and properties to describe resources—as entities are called in the context of RDF. The data engineers are in theory free to combine any number of vocabularies. Furthermore, they can extend existing vocabularies or introduce new types and properties if necessary. This flexibility allows for modelling knowledge of a wide range of different domains. However, the conceptual flexibility of the LOD cloud poses several challenges and questions.

First of all, in order to be able to make use of the data it needs to be practically accessible, in the sense that users need to be able to find information of specific structure. For instance, assume a user looking for data about scientists and their publications. While some data providers like DBLP or ACM are obvious sources for this kind of information, there might be many other contributors which publish data of this type. Just like web search engines represent gateways and central access points for the web of documents, the LOD cloud needs semantic search engines to direct users with specific information needs to the relevant sources of Linked Data [5,9, 14]. This is of importance not only for end users, but also in query federation scenarios where formal SPARQL queries need to be distributed to several relevant data providers on the LOD cloud. The question arising in this context is which structural information needs to be captured in an appropriate index [11]. The schema structure of the graph data on the LOD cloud is of importance also in other settings: consumption of Linked Data in applications requires for a programmable interface to the LOD cloud. Such interfaces should represent the structure of certain types of data in a relatively stable fashion while at the same time being capable of handling the flexibility of Linked Data [27]. To continue the above example, an interface to Linked Data about scientists should be based on typical properties scientists exhibit on the cloud and should consider the properties that link scientist to other objects such as their publications or institutions. This requires for the detection of typical schema patterns on the LOD cloud. A third scenario where schema structure is of interest is settled in the context of publishing Linked Data. Assume a data engineer who wants to publish his own data set about scientists and publications on the LOD cloud. When modelling the data and considering the possible choices of vocabularies, he might be interested in what are common approaches, best practices, and established models for this kind of information [26]. The benefit of aligning a new data set with established models is the re-usability of the data and the better integration in the LOD cloud. Accordingly, what the data engineer is interested in is which vocabularies are used together to describe the domain he is working on.

All of these illustrative examples motivate the need to take a closer look at schema structures on the LOD cloud. The underlying mechanisms to model schema structures in Linked Data implicitly via properties and explicitly by using types are manifested in the Resource Description Framework. A data publisher can explicitly state the type of the entities he models. This is done by linking a resource via an rdf:type property to concept classes. For instance, he can express the fact that a resource x is a person by stating that x is of type foaf:Person[4]. To implicitly describe a resource via its

[4] Taken from the famous Friend-of-a-Friend (FOAF) vocabulary for describing people and their relations. See: `http://www.foaf-project.org/`, accessed: 23 March, 2013

properties, a data publisher can use numerous vocabularies or define new properties with specific semantics. For instance, he can describe resource x to have a name by attaching a property foaf:name with a literal value of *"Mr. X"*. Also the relation to other resources is modelled in this way. He can state that x is the author of a resource y by connecting them with the property foaf:made. These two manifestations of schema information are to a certain extent redundant, i. e., certain resource types entail typical properties and certain properties occur mainly in the context of particular types. For instance, we would expect a resource of type foaf:Person to have the properties foaf:name or foaf:age. Likewise, we can assume a resource with the property skos:prefLabel to be of type skos:Concept. Vocabularies also establish and refine hierarchies of concept classes or properties. A vocabulary is typically published on the web and can be referred to by its URI. Commonly used name space definitions provide a short abbreviation of the URI and serve as human readable labels for a specific vocabulary. For instance, rdf: is commonly used to refer to the basic RDF vocabulary, rdfs: for the RDF Schema vocabulary or foaf: and skos: for popular application specific vocabularies. Vocabularies can be combined and mixed to describe data. The general recommendation for designing Linked Data is to re-use existing vocabularies instead of re-inventing own descriptions. Thus, the choice of which vocabularies to use and how to combine them affects the data model on the schema level.

To summarize, we have a decentralized setting where anyone can contribute data and has the freedom to model data as needed and as preferred. This implies the need to understand the schema structures on the LOD cloud for several applications. The requirement of understanding schema structures and the lack of central control or curation of these structures motivate the need for an analysis of schema structures on the LOD cloud. We formulate the following research questions we want to address with such an analysis:

1. How are type and property structures used on the LOD cloud at global scale? Which structures are more informative in describing the data? To which degree are the two structures redundant? Do we need to capture both of these two schema structures or is one of these sufficient to explain the other?
2. How do data providers model their Linked Data? Is data provided by a single authority rather strongly or weakly structured with respect to the schema structure they use?
3. How are vocabularies used and correlated? Can we observe patterns in the use of vocabularies?

Answering these questions is not a trivial task and implies several requirements. First of all, we need a method to reliably observe and extract schema structures on large, distributed RDF graphs as they occur on the LOD cloud. Second, we need to provide suitable metrics capable of measuring the information, redundancy, and structure encoded in these schema structures. Finally, we need to implement this analysis on a large-scale excerpt of the LOD cloud. In this paper, we address all three requirements and present a method and metrics for performing such an analysis. We base our method on a scalable approach for extracting schema information from Linked Data [21]. The metrics are based on information theoretic concepts such as entropy

and mutual information. To address the impact of different data providers, we propose to use a decomposition of a large-scale Linked Data set based on the concept of pay-level domains (PLD). This allows for the observation of local effects as well as individual approaches for modelling Linked Data on a schema level. As data set, we use the well established data provided for the Billion Triple Challenge (BTC) track of the Semantic Web Challenge[5] carried out annually with the International Semantic Web Conference.

Our analysis provides insights into the information encoded in the different schema characteristics: We observe that implicit schema information is far more discriminative and that applications involving schema information based on either types or properties alone will only capture between 63.5% and 88.1% of the schema information contained in the data. Investigating the distribution of the normalized mutual information across single PLDs, we have found three unusual accumulations of PLDs sharing the same value of normalized mutual information. Overall, 163 PLDs (19.4%) share a common value of normalized mutual information of 0.99 or higher. Finally, we have observed different approaches for modelling linked data w.r.t. the use and combination of vocabularies. One approach is to follow a strong schematic design where few vocabularies are consistently used to model nearly all of the data. The other approach is to use a mix of several vocabularies and applying them as needed to model the available information.

Existing analyses of the LOD cloud typically focus either on obtaining statistical information of the characteristics of the graph (e. g., [3,8,6,10,29]), investigate the compliance of the data to the LOD principles (e. g., [19,23,22]), or apply information about the graph structure for query optimization (e. g., [25,24,15]). In this paper, we focus on obtaining new statistics and insights into the nature of LOD. However, we also have concrete applications for using the obtained insights in mind. These possible applications range from query recommendations when searching for LOD sources [14], providing programmatic access to the LOD cloud [27], and supporting the data engineer when modeling data using LOD vocabularies [26]. We extend the analyses conducted in earlier studies in various different ways: Existing works only consider one part of the schema information and ignore the other. For example, Neumann and Moerkotte consider outgoing property sets for their query optimizations but ignore the types [25]. Others apply their measures on both the properties and the types. However, they treat them as equal kind of information such as [8,23]. So far, there is no analysis on the combined use of property sets and type sets for describing the entities. Existing works like [3] only investigate the co-occurrence of a single property with a single type and thus miss the combined use of multiple properties and multiple types. Other works only considered specific relations in the LOD cloud like the OWL sameAs network [10], while we consider all kinds of edges. Finally, we analyze the strengths of the vocabularies in the pay-level domains, i. e., determine how dominating a vocabulary is for describing the data. We also compute frequent item sets to investigate which vocabulary combinations of properties and types appear very often and cannot be explained as random co-occurrences.

[5] `http://challenge.semanticweb.org/`, accessed: 23 March, 2013

The remainder of the paper is organized as follows: In the subsequent section, we describe the setup phase to prepare for our extensive analysis of Linked Open Data. The setup comprises the definition of the information to be collected, introduces a probabilistic distribution model for RDF types and properties, and presents how we estimate the probabilities based on a schema level index for LOD [21]. In this section, we also describe the data sets we use for our experiments and analysis. In Section 3, we conduct the first step of our analysis. We introduce various information theoretic measures and apply them on the schema structures observed in the data set. Subsequently, Section 4 presents the second step of our analysis. It provides insights into selected measures on the level of single data providers. In this part, we focus on the distribution of observed values rather then specific values. Finally, Section 5 presents our investigations of vocabulary usage and comprise the third and last step of our analysis. We compute the strength of vocabularies in the data published by specific data providers as well as determine the dominating vocabularies in the data. The related work in the areas of statistical analysis of Linked Open Data, compliance to Linked Open Data principles, as well as analysis for the purpose of query optimization on Linked Open Data is presented in Section 6, before we conclude the paper with a summary of our findings.

2 Obtaining Statistics about Schema Structures of Linked Open Data

In the introduction, we have identified RDF types and properties as observable schema information provided by Linked Data. Knowledge about the use of specific types and properties in any given data set is key to understanding and making use of the information contained in the data set. This importance is also reflected in the implementation of the VoID vocabulary [2], which can be used to describe the presence of type and property combinations in an RDF data set. However, VoID itself is not available for all data sources, neither it obliges the provision of *all* used vocabularies or combinations of types and properties. Thus, we cannot use VoID as basis for the intended analysis but rather need to collect and process the schema structures from raw data.

2.1 Information to be Collected

We are interested in the schema structure of Linked Data. As there is no ready available schema defined for the data, we need to extract such a schema from the data itself. To this end, we consider the entities described on the LOD cloud and the schema information they actually exhibit. This means, we observe the types and properties actually used to describe real data on the web. In this way, we also observe which vocabularies are used on the LOD cloud as the types and properties are specifically defined in the context of vocabularies.

In general, an entity is technically represented in RDF by a URI (uniform resource identifier). The schema structure of an entity is expressed by its types and properties, which fall into specific vocabularies. To aggregate all this information for one entity, we need to consider all RDF triples with the same *unique subject URI* (USU) (cf.

notion of Semantic Web terms in Ding and Finin [8]). For the purpose of observing the redundancy, entropy, or patterns in this data we furthermore need to aggregate all entities with the same schema structure. Only in this way, we can obtain count statistics for computing metrics of higher order. More formal, we identify for each USU x the set t of types defined via RDF triples of the form x rdf:type t, the set r of all properties or relations defined via RDF triples x property y for arbitrary resources y and property $\neq$ rdf:type, as well as the set V of vocabularies used in t and r.[6]

2.2 Probabilistic Distribution Model for Type and Property Sets

We are interested in the combinations of RDF types and properties attached to resources. The space of all possible type combinations therefore is the power set $\mathcal{P}(\mathit{Classes})$ of all class types defined in the data. While the power set itself is huge (of size $2^{|\mathit{Classes}|}$), we can restrict ourself to the subset $TS \subset \mathcal{P}(\mathit{Classes})$ of actually observed combinations of RDF types in the LOD cloud. For a given resource, we can now observe $t \in TS$ which corresponds to a set of types (e.g., the set {foaf:Person, dbpedia:Politician}).

Likewise, the properties observed for a resource is a combination of all possible properties. Accordingly we deal with an element from the power set $\mathcal{P}(\mathit{Properties})$ over all properties defined in the data. Again, we only need to consider the subset PS of actually observed property sets. For an individual resource, we observe $r \in PS$ which corresponds to the set of its properties (e. g., the set {foaf:familyName, foaf:givenName, dbpedia:spouse}).

To model the joint distribution of type sets and property sets, we introduce two random variables T and R. The range of these two random variables are the elements in TS and PS, respectively. Both random variables are of discrete nature and their joint distribution can be characterized by:

$$P(T = t, R = r) = p(t, r) \tag{1}$$

where $p(t, r)$ is the probability to observe the concrete set t of types and the set r of properties for a randomly chosen entity, i. e., unique subject URIs. Based on this joint distribution, we can also identify the marginal distributions of T and R:

$$P(T = t) = \sum_{r \in PS} p(t, r) \tag{2}$$

$$P(R = r) = \sum_{t \in TS} p(t, r). \tag{3}$$

[6] Please note, we use the letter r for sets of properties (inspired by the term relation), as p will be used to denote probabilities.

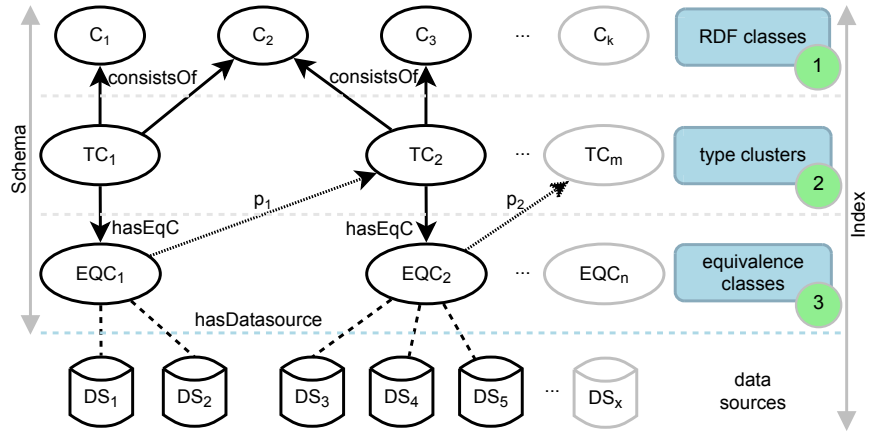


Figure 1 SchemEX index structure with three layers leveraging RDF typings and property sets

2.3 Using a Schema Level Index to Estimate Probabilities

SchemEX is a schema-level index for distributed, web scale RDF graphs such as the LOD cloud. The purpose of SchemEX [21, 20, 13] is to link schema structures to meta data about entities which conform to this structure. The meta data can be, for instance, volume information on how many entities comply with this schema structure or data sources which provide such entities[7]. The central schema elements of SchemEX are type clusters (TC) and equivalence classes (EQC). A TC represents all entities which conform to a well defined set of types. The EQC further subdivide the entities represented in a TC into disjoint subsets, defined by the set of properties the entities have and in which TC the object of the triple lies. An overview of the schema information contained in a SchemEX index providing data source information is shown in Figure 1.

The TC elements in SchemEX [21] correspond directly to the notion of types sets in TS given in Section 2.2. The equivalence classes in SchemEX subdivide the type clusters and are defined by the set of properties the triples have as well as the type cluster the object of the triple lies in. Hence, they are more fine-grained than the property sets we are interested in. However, aggregating the equivalence classes defined by the same set of properties over all attached type clusters, we obtain exactly the property sets PS introduced in Section 2.2. In this way, we can easily construct the set PS from a SchemEX index.

If we denote with SchemEX(t, r) the set of entities represented in the SchemEX index that correspond to the resources with types t and properties r, we can estimate the above probability of observing a resource to have a particular type and property set by

$$\hat{p}(t, r) = \frac{|\text{SchemEX}(t, r)|}{N}$$

[7] Data sources are, e. g., static RDF documents and SPARQL endpoints [17].

where N is the number of all entities observed when building the SchemEX index.

The estimates for the probabilities $p(t, r)$ above are central to all relevant metrics and effectively only need to be aggregated and normalized accordingly. However, the number of observed type sets and property sets indicates the high number of possible combinations (i. e., $|TS| \times |PS|$). The pragmatic solution to this quadratic development of combinations is not to compute all of the probabilities, but only those which actually have a non zero value. This does not affect the results of the computed metrics, as zero probabilities do not affect their overall values.

Another noteworthy feature of SchemEX indices is that they can be computed very efficiently and for large RDF graphs using a stream-based approach. In this case, the analytical component is operating in a single pass fashion over a set of RDF triples. By using a windowing technique, it is possible to obtain a very accurate schema of the processed data using commodity hardware. However, the windowing technique entails a certain loss of accuracy w.r.t. schema information. The extent of this loss has been analyzed in detail in [12]. The type of schema information and the metrics we use in the context of this paper are relatively stable. Deviations typically range up to 5%, in single cases differences of up to 10% have been observed in an empirical evaluation.

Thus, the combination of the efficient computation mode for a SchemEX index with the compact representation of the joint distribution allows for the analysis of schema patterns on web scale RDF graphs.

2.4 A Large-scale Data Set Suitable for Schema Analysis

For our empirical analysis, we use the different segments of the data set provided for the Billion Triple Challenge (BTC) 2012. The BTC data set has been crawled from the web in a typical web spider fashion and contains about 1.44 billion triples. Thus, it covers Linked Data of different origin and various quality. It is divided into five segments according to the set of URLs used as seed for the crawling process: Datahub, DBPedia, Freebase, Rest, and TimBL. Details about the different segments and the crawling strategies used for collecting the data are described on the website of the BTC 2012 data set[8].

As the efficient stream-based computation of a SchemEX index entails a certain loss of accuracy regarding the schema, we have to check that these inaccuracies do not affect the overall results. To this end, we have used smaller data sets to compute the schema once with our stream-based approach and once using a lossless approach and compared the values of our metrics on these two schemas. As the computation of a gold standard schema has high requirements regarding the hardware resources, we were limited to derive lossless schema for data sets of approximately 20 million triples. As small data sets, we used (A) the full *Rest* subset (22,328,242 triples) of the BTC dataset, (B) an extract of the *Datahub* subset (20,505,209 triples), and (C) an extract of the *TimBL* subset (9,897,795 triples). The extracts correspond to the data

[8] BTC 2012 data set: `http://km.aifb.kit.edu/projects/btc-2012/`, accessed: 25 March, 2013

Table 1 Size of the data sets based on the BTC segments.

Smaller data sets		**Number of Triples**
(A)	Rest (full)	22,328,242
(B)	Datahub (extract)	20,505,209
(C)	TimBL (extract)	9,897,795
Larger data sets		**Number of Triples**
(D)	Datahub (full)	910,078,982
(E)	DBPedia	198,090,024
(F)	Freebase	101,241,556
(G)	TimBL (full)	204,806,741

sets that would have been obtained by stopping the crawling process after two hops from the Datahub URI seed set and four hops from the TimBL URI seed set. We did not produce extracts for DBpedia and Freebase as the hop information is not provided for these BTC segments.

The stream-based approach for computing a SchemEX index is also applicable to the full data crawls of (D) *Datahub* (910,078,982 triples), (E) *DBPedia* (198,090,024 triples), (F) *Freebase* (101,241,556 triples) and (G) *TimBL* (204,806,741 triples). For this efficient SchemEX computation we have used the same parameter settings as in [21], i. e., a window size of 50,000 instances for schema extraction. While the smaller data sets serve the purpose of confirming the stability of the stream-based approach, the larger data sets are used for the actual analysis of explicit and implicit schema information on the LOD cloud. We consider the data sets particularly useful as they span different aspects of the LOD cloud. With *Datahub*, we have got a sample of several publicly available linked RDF data sources registered in a central location. *DBpedia* is interesting as it is one of the central and most connected resources in the LOD cloud extracted from the collaboratively curated Wikipedia. *Freebase*, instead, is also a collaborative knowledge base, but here a selected set of data sources have been merged with high effort. The *TimBL* data set is a crawl starting at the FOAF profile of Tim Berners-Lee (thus, the name). Hence, it provides a snapshot from yet a different part of the LOD cloud, namely starting at small, manually maintained RDF files. Table 1 gives an overview of the sizes of the different data sets obtained from the BTC data set.

2.5 Identification of Subsets Controlled by Individual Data Providers

While the segments of the BTC data set can provide us with a rather global view on the schema structure for several parts of the LOD cloud, they do not describe the use of schema information on the level of single data providers. To be able to investigate the behaviour and style of single data providers when modelling the schema of Linked Data, we split the full BTC data set along the pay-level domains (PLD). The pay-level domain is defined as the part of a domain name, which can typically be registered by companies, organisations, or private end users. Depending on the country, the PLD can start directly before the top-level domain (e. g., in Germany

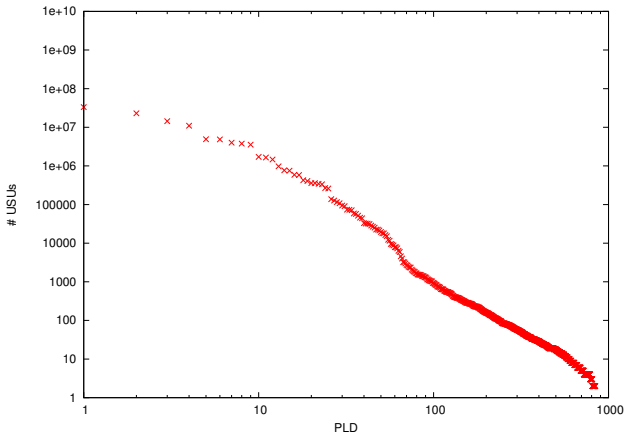


Figure 2 Number of USU per PLD (in descending order of the # of USUs in the PLDs).

before the .de), or before some administrative second level domains (e. g., in Great Britain the domain name suffix .co.uk). Sub-domain names are never considered in a PLD. For instance, data published under west.uni-koblenz.de would be assigned to the PLD uni-koblenz.de. A pay-level domain is considered a good estimate for which fragments of data on the LOD cloud are controlled by the same authorities [8].

Splitting the BTC data set along the PLDs provides us with a total of 840 smaller data sets. The sizes of these data sets vary very much. The size depends on the actual amount of Linked Data available at the data providers as well as the crawling strategy implemented for the creation of the entire BTC data set. The plots in Figure 2 and Figure 3 show the distribution of the sizes of the PLDs in terms of USUs and triples, respectively. We can see that both seem to follow a Zipf distribution. This needs to be kept in mind, as especially for the PLDs with very little data it is difficult to obtain a reliable estimation of the joint distribution of type and property sets. The lack of data can lead to a skewed and biased distribution and, as a consequence, to deviations in our metrics. Accordingly, we will perform our analysis on both: the entire collection of PLD data sets as well as a subset of the 100 largest PLDs in terms of size. An interesting observation is that each of these 100 PLDs provide more than 1,000 USUs, i. e., describe more than 1,000 entities. The number of triples per PLD is about a magnitude of order higher than the number of USUs per PLD. Precisely, we observed on average about 9.862 triples per USU. Overall, the largest 100 PLDs make about 99,84% of the overall USUs observed in our data set and 99,96% of all triples. Accordingly, we can consider the largest PLDs to cover by far the major part of the data contained in the BTC data set.

Comparing the number of USUs and triples as shown in Figure 4, we can furthermore observe a linear correlation. In fact, after cleaning data from three strong outliers, we have computed Pearson's r and observe a very large positive correlation of 0.978 between the number of USUs and triples in our data. This means, we do not have to distinguish further between the data sets with most triples and most USUs, they are essentially identical.

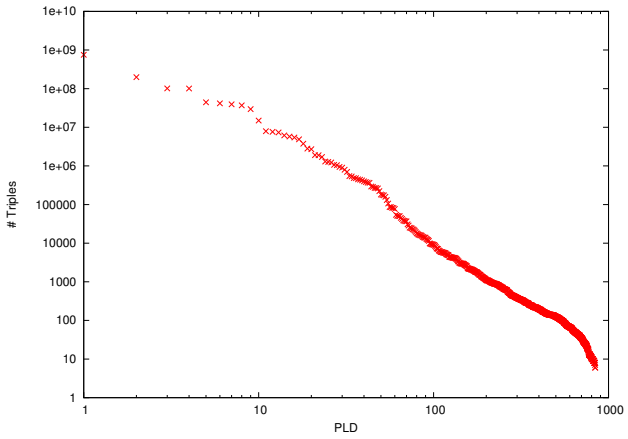


Figure 3 Number of triples per PLD (in descending order of the # of triples in the PLDs).

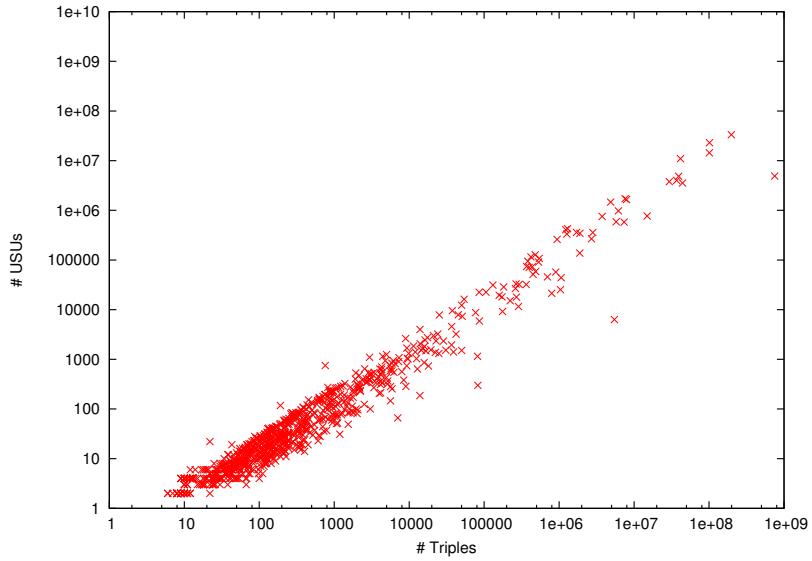


Figure 4 Scatter plot of triple count and USU count. Each cross represents a single PLD.

3 Information Theoretic Analysis of Schema Structures over Unique Subject URIs

For analyzing the LOD cloud, we are interested in several characteristics of the joint distribution $P(T, R)$ introduced above. The main questions that we want to answer are:

(a) How much information is encoded in the type set or property set of a resource on a global scale?

(b) How much information is still contained in the properties, once we know the types of a resource?
(c) How much information is still contained in the types, once we know the properties of a resource?
(d) To which degree can one information (either properties or types) explain the respective other?

To answer these questions, we first introduce appropriate measures that can be applied to the joint distribution of type sets and property sets. In Section 3.2, we apply these measures on the segments of the BTC data set and discuss our findings in Section 3.3.

3.1 Information Theoretic Measures

All our measures are based on the *entropy* of probabilistic distributions [28], the standard concept to measure information.

3.1.1 Entropy of the Marginal Distributions

To answer question (a) how much information is encoded in the type or property set of a resource, we need to look at the marginal distributions. These provide us with the probability of a certain resource to show a particular set of types or properties. The entropy of the marginal distributions of T and R is defined as:

$$H(T) = -\sum_{t \in TS} P(T = t) \cdot \log_2 \left(P(T = t)\right) \tag{4}$$

$$H(R) = -\sum_{r \in PS} P(R = r) \cdot \log_2 \left(P(R = r)\right). \tag{5}$$

The values $H(T)$ and $H(R)$ give us an idea of how much information is encoded in the sets of types or properties of the resources. A higher value corresponds to more information, which in turn means that the sets of types and sets of properties appear more equally distributed. To be more concrete: an entropy value of 0 indicates that there is no information contained. For instance, a value of $H(T) = 0$ would indicate that all resources have exactly the same set of types (likewise for $H(R) = 0$ the exactly same set of properties). A maximal value, instead, is reached when the distribution is an equal distribution, i.e., each set of types or properties is equally probable. This fact also allows for normalizing the entropy values by:

$$H_0(T) = \frac{H(T)}{H^T_{\max}} = \frac{H(T)}{\log_2(|TS|)} \tag{6}$$

$$H_0(R) = \frac{H(R)}{H^R_{\max}} = \frac{H(R)}{\log_2(|PS|)}. \tag{7}$$

The normalized entropy value ranges between 0 and 1 and indicates whether the distribution is closer to a degenerated or a uniform distribution. Please note that this normalization also renders the entropy values independent of the choice of the basis of the logarithm.

3.1.2 Conditional Entropy

The question (b), how much information is still contained in the properties, once we know the types of a resource implies a conditional probability and, thus, a conditional entropy. We have to take a look at the distribution of the property sets given that we already know the types of a resource. The entropy in this case (i.e., the conditional entropy) conveys how much information is still in the additional observation of the properties. Again, if the set of types perfectly defines the set of properties to expect, there would be no more information to be gained. Thus, the conditional entropy would be zero. If, instead, the types were virtually independent from the properties, we would expect to observe the marginal distribution of the properties and its according entropy. Formally the conditional entropy for a given type set t is defined as:

$$H(R|T=t) = -\sum_{r \in PS} P(R=r|T=t) \cdot \log_2 \left(P(R=r|T=t)\right) \tag{8}$$

$$= -\sum_{r \in PS} \frac{p(t,r)}{P(T=t)} \cdot \log_2 \left(\frac{p(t,r)}{P(T=t)}\right). \tag{9}$$

Equivalently, to answer question (c), the conditional entropy for a given property set r is:

$$H(T|R=r) = -\sum_{t \in TS} \frac{p(t,r)}{P(R=r)} \cdot \log_2 \left(\frac{p(t,r)}{P(R=r)}\right). \tag{10}$$

One value of particular interest is a conditional entropy of 0. For instance, in the case of $H(R|T=t)=0$ knowing the set of types t is already conveying all the information, i. e., the set of properties can be derived with probability 1. Equivalently in the case of $H(T|R=r)=0$, we can derive the set of types from the set of properties. Accordingly, we are interested in the probability of such a conditional entropy of 0, e. g., $P(H(R|T=t)=0)$ for the case of given type sets. Treating the conditional entropy itself as a random variable allows for easily estimating this probability by $P(H(R|T=t)=0) = \sum_{H(R|T=t)=0} P(T=t)$.

3.1.3 Expected Conditional Entropy

The conditional entropies defined above are fixed to one particular set of types t or set of properties r. As we are interested in a global insight on a large scale data set like the LOD cloud, it is not feasible to look at all the individual observations. Rather we need an aggregated value which we introduce here.

The expected conditional entropy $H(R|T)$ follows an idea similar to the one of looking at the probability to observe a conditional entropy of zero. This aggregated measure of $H(R|T)$ also considers the conditional entropy as a random variable and computes the expected values of this variable based on the probability to actually observe a certain set of types t. The definition of this aggregation is:

$$H(R|T) = \sum_{t \in TS} P(T=t) \cdot H(R|T=t) \tag{11}$$

$$= -\sum_{t \in TS} P(T{=}t) \left[\sum_{r \in PS} P(R{=}r|T{=}t) \log_2 \left(P(R{=}r|T{=}t)\right) \right] \tag{12}$$

$$= -\sum_{t \in TS} \sum_{r \in PS} p(t,r) \cdot \log_2 \left(\frac{p(t,r)}{P(T=t)} \right) \tag{13}$$

and equivalently $H(T|R)$ is defined as:

$$H(T|R) = \sum_{r \in PS} P(R=r) \cdot H(T|R=r)$$

$$= -\sum_{r \in PS} \sum_{t \in TS} p(t,r) \cdot \log_2 \left(\frac{p(t,r)}{P(R=r)} \right). \tag{14}$$

3.1.4 Joint Entropy

In addition to the conditional entropies introduced above, we will also take a look at the joint entropy of T and R. It is defined as:

$$H(T,R) = -\sum_{t \in TS} \sum_{r \in PS} p(t,r) \cdot \log_2 \left(p(t,r)\right) \tag{15}$$

3.1.5 Mutual Information

To finally answer question (d) how far one of the schema information (either properties or types) can explain the respective other, we employ mutual information (MI) [7]. MI captures the joint information conveyed by two random variables—and thereby their redundancy. The MI of explicit and implicit schema information of the LOD cloud is defined as:

$$I(T,R) = \sum_{r \in PS} \sum_{t \in TS} p(t,r) \cdot \log_2 \frac{p(t,r)}{P(T=t) \cdot P(R=r)}. \tag{16}$$

The $\log$ expression in this sum, i.e., the expression $\log_2 \frac{p(t,r)}{P(T=t) \cdot P(R=r)}$ is also known as *pointwise mutual information* (PMI). PMI can be explained as the strength of the correlation of two events, in our case how strongly a particular type set and a particular property set are associated with each other.

One characteristics of MI is the open range of its values. A normalization of MI is given in [31] and involves the entropy of the marginal distributions of T and R. It is used as a direct measure for redundancy and is defined as:

$$I_0(T,R) = \frac{I(T,R)}{\min\left(H(T), H(R)\right)}. \tag{17}$$

3.2 Results of our Analysis

Table 2 gives an overview of the count statistics and values obtained from the different measures for the smaller data sets (A), (B) and (C). The table compares the values of the lossless gold standard schema computation with the efficient stream based approach. The observed deviations in the number of type sets in the data sets (A), (B) and (C) are very low[9] and confirm the accuracy observed in previous experiments [12]. While for the data sets (B) and (C) also the number of property sets obtained by the stream-based approach does not differ much from the gold standard, we observe a slightly stronger deviation on the Rest data set (A). The sheer count of type and property sets, however, does not reflect the number of entities behind the individual elements in the schema. Thus, it is necessary to consider the distributions and the metrics derived from those. Here, we observe a generally quite good behaviour of the efficient schema approximation using the stream-based approach. The differences in the metrics are relatively small and consistent within each data set. In conclusion, we have decided that the loss of accuracy due to the efficient stream-based schema computation is counterbalanced by the capabilities to analyze data sets which are an order of magnitude larger: the observation of more data allows for a more reliable estimation of the joint distribution $p(t,r)$ and, thus, a more sound evaluation of schema information on the LOD cloud.

Table 3 gives an overview of the computed information theoretic measures on the large data sets. Already the differences in the number of observed type and property sets underline the heterogeneity of the data sets. We will now go into the details of the results.

3.2.1 Entropy in Type Sets and Property Sets

We can observe the tendency that the property sets convey more information than type sets. This can be observed in the higher values of the normalized entropies. For instance, the normalized marginal entropy of the property sets has a value of 0.324 on the *DBpedia* (E) data set, while the normalized marginal entropy of the type sets is 0.093. This observation provides a hint that on *DBpedia* the distribution into type sets is far more skewed than the distribution of property sets. Similar observations can be made for the data sets (A), (F), and (G), though to a lower extent. An exception is the *Datahub* data set (D), where the distribution of resources in type sets and property sets seems comparable.

[9] Please note, that the efficient stream-based approach can cause an increase in the number of type sets and property sets, as well. This is due to the fact that a single missed type can cause the deduction of a new type set which does not actually occur in the lossless gold standard.

Table 2 Statistics of the schema information obtained for the smaller data sets when using lossless and efficient (stream-based) schema computation.

Data set	**(A) Rest**		**(B) Datahub (extract)**		**(C) TimBL (extract)**	
Schema construction	lossless	efficient	lossless	efficient	lossless	efficient
$\|TS\|$	791	793	3,601	3,656	1,306	1,302
$\|PS\|$	8,705	7,522	4,100	4,276	3,015	3,085
$H(T)$	2.572	2.428	3.524	3.487	2.839	2.337
$H_0(T)$	0.267	0.252	0.298	0.295	0.274	0.226
$H(R)$	4.106	4.708	6.008	6.048	3.891	3.258
$H_0(R)$	0.314	0.366	0.501	0.501	0.337	0.281
$H(T\|R)$	0.295	0.289	1.158	1.131	0.670	0.512
$P(H(T\|R=r)=0)$	29.32%	38.02%	60.77%	57.79%	27.81%	21.52%
$H(R\|T)$	1.829	2.568	3.643	3.692	1.723	1.433
$P(H(R\|T=t)=0)$	6.22%	5.31%	12.01%	11.08%	6.06%	4.51%
$H(T,R)$	4.401	4.997	7.166	7.179	4.561	3.770
$I(T,R)$	2.277	2.140	2.365	2.356	2.169	1.824
$I_0(T,R)$	0.885	0.881	0.671	0.676	0.764	0.781

Table 3 Statistics of the schema information obtained for the full data sets when using efficient (stream-based) schema computation.

Data set	**(A) Rest**	**(D) Datahub (full)**	**(E) DBpedia**	**(F) Freebase**	**(G) TimBL (full)**
$\|TS\|$	793	28,924	1,026,272	69,732	4,139
$\|PS\|$	7,522	14,712	391,170	162,023	9,619
$H(T)$	2.428	3.904	1.856	2.037	2.568
$H_0(T)$	0.252	0.263	0.093	0.127	0.214
$H(R)$	4.708	3.460	6.027	2.868	3.646
$H_0(R)$	0.366	0.250	0.324	0.166	0.276
$H(T\|R)$	0.289	1.319	0.688	0.286	0.386
$P(H(T\|R=r)=0)$	38.02%	11.59%	54.85%	80.89%	15.15%
$H(R\|T)$	2.568	0.876	4.856	1.117	1.464
$P(H(R\|T=t)=0)$	5.31%	10.83%	3.73%	2.05%	1.60%
$H(T,R)$	4.997	4.779	6.723	3.154	4.032
$I(T,R)$	2.140	2.585	1.178	1.751	2.182
$I_0(T,R)$	0.881	0.747	0.635	0.860	0.850

3.2.2 Conditional Entropies

Looking at the expected conditional entropies reveals some interesting insights. Recall that the aggregation we chose for the conditional entropy provides us with the expected entropy, given a certain type set or property set. We can see in Table 3 that the expected entropy given a property set tends to be far lower than the one when given a type set. In conclusion: knowing the properties of a resource in these cases already tells us a lot about the resource, as the entropy of the conditional distribution can be expected to be quite low. On the contrary, when knowing the type of a resource the entropy of the distribution of the property sets can be expected to be

still relatively high (when compared to the entropy of the marginal distribution). We looked at the data more closely to investigate how often a given type set is already a clear indicator for the set of properties (and vice versa). This insight is provided by considering the probabilities $P(H(R|T = t) = 0)$ and $P(H(T|R = r) = 0)$ to observe a conditional entropy of 0. The most extreme case is the *Freebase* data set (F), where for 80.89% of all resources it is sufficient to know the set of properties in order to conclude the set of types associated with this resource. Knowing, instead, the types of a resource conveys less information: only in 2.05% of the cases this is sufficient to predict the set of properties of a resource. Again, and with the exception of *Datahub* (D), the other data sets exhibit a similar trend. However, at very different levels: the probability of knowing the type set for a given property set ranges between 15.15% and 54.85%. The *Datahub* data set shows a far more balanced behaviour. Both probabilities $P(H(R|T = t) = 0)$ and $P(H(T|R = r) = 0)$ are at around 11%, confirming the particular form of this data set.

3.2.3 Mutual Information

Finally, the value of the normalized MI gives us insights on how much one information (either properties or types) explains the respective other. Also here, we observe a quite wide range of values from 0.635 on *DBpedia* (E) to 0.881 on *Rest* (A). Accordingly, extracting only type or only property information from LOD can already explain a quite large share of the contained information. However, given our observations a significant part of the schema information is encoded also in the respective other part. The degree of this additional information depends on the part of the LOD cloud considered. As a rule of thumb, we hypothesise that collaborative approaches without a guideline for a schema (such as *DBpedia*) tend to be less redundant than data with a narrow domain (*TimBL*) or some weak schema structure (*Freebase*).

3.3 Discussion of the Results

The observations on the large data sets provide us with insights into the form and structure of schema information on the LOD cloud. First of all, the distribution of type sets and property sets tend to have a relatively high normalized entropy. We can conclude that the structure of the data is not dominated by a few combinations of types or properties. Accordingly for the extraction of schema information, we cannot reduce the schema to a small and fixed structure but need to consider the wide variety of type and property information. Otherwise the schema would loose too much information.

A second observation is the dependency between types and properties. The conditional entropy reveals that the properties of a resource usually tell much more about its type than the other way around. This observation is interesting for various applications. For instance, suggesting a data engineer the types of a resource based on the already modelled properties seems quite promising. We assume that this observation can also be seen as an evidence that property information on the LOD cloud actually considers implicit or explicit agreements about the domain and range of the according

property. However, this observation is not valid for the entire LOD cloud. Depending on the concrete setting and use case, a specific analysis might need to be run.

Finally, the observed MI values underline the variance of schema information in the LOD cloud. Ranges from 63.5% to 88.1% redundancy between the type sets and property sets have been observed. Thus, approaches building a schema only over one of these two types of schema information run at the risk of a significant loss of information.

4 Analysis of Schema Structures over Pay-level Domains

After having looked at information and redundancy in schema structures on the LOD cloud at a global level, we now focus on single data providers. Single data providers typically control the vocabulary they use and the schema structures they implement[10]. Thus, a relevant question is how structured and consistent is the schema of data sets provided by individual data providers. This is of particular interest when integrating data within applications. A more consistent and reliable schema might be in favour when several data providers offer the same type of information.

As boundaries for data controlled by a single authority, i. e., for our notion of a data provider, we use pay-level domains (PLDs) as described in Section 2.5. For the purpose of analyzing the schema structures on this level, we can use the same metrics as in Section 3.1. However, given the high number of 840 individual data providers in our data set, there is little insight to be gained from looking at single data sources. Rather, we are interested in insights based on aggregated analytical results. We want to answer the following questions w.r.t. the level of data publishers:

(a) How is the information in schema structures distributed over PLDs?
(b) What are typical levels of entropy and redundancy in schema information on a PLD level?
(c) For which PLD do we observe a particular outlier behaviour?

In the following, we first argue for the choice of distributions we consider for our analysis of the schema structure of LOD on PLD granularity. Subsequently, we present the results for the entropy of marginal distributions, expected conditional entropy, as well as the distribution of the normalized mutual information over the PLDs. Finally, we analyze outliers in the normalized mutual information distribution.

4.1 Choice of Distributions over Pay-level Domains

When applying entropy and mutual information measures to data provided by individual data providers, we need to be aware that the data observations underlying the probabilistic model is far thinner. If a data provider publishes very little information (RDF triples about only a few or even a single USU) the entropy of the schema structures becomes less expressive. Furthermore, we already motivated the need to

[10] This might not entirely be the case in semi-automated extraction of Linked data (e. g., DBPedia) or when using crowd sourcing for the creation of Linked Data (e. g., Freebase)

aggregate data in order to be able to obtain insights at the level of data providers. The approach we take here is to consider the distribution of the values of the measures in Section 3.1.

To consider distributions of any of the measures like those used before, we need to model them as random variables. This technical twist has already been used in Section 3.1.3 for the analysis of the expected conditional entropy on the larger data sets. Subsequently, we estimate the distribution from the observations we can make over the PLDs. Formally, we consider the values of the marginal entropy, the conditional entropy, and the mutual information as results of random experiments and model them with a random variable. The sample space of all possible outcomes corresponds to the space of all possible ways to design, model, and publish Linked Data on the cloud.

Take, for instance, the normalized marginal entropy. If we model this measure as random variable, then we are interested in its distribution and the moments of this distribution. For the purpose of estimating the distribution, we consider the data sets of each data provider as an individual and independent sample from the entire sample space.

From this sample, we can also obtain values for mean and variance. By looking at the distribution itself, we can get an impression of how the distribution looks like. The mean might be particular intresting as it gives the value of marginal entropy we can expect when picking a random data provider. For visualizing the distribution, we will use histograms. To compute the distributions, we bin values within equidistant intervals. Given the continuous range of possible values, this allows for a better representation of the density function. Furthermore, we make use of cumulative distribution functions. Specifically, we consider the following distributions:

- Entropy of the marginal distributions: We consider the entropy for both, the marginal distributions of type sets and property sets. This allows for an analysis of the distribution of the type sets and property sets on PLD level. Effectively, we look at the distribution of the values taken by $H(T)$ and $H(R)$ over all PLDs.
- Expected conditional entropy: The distribution of $H(T|R)$ and $H(R|T)$ values provides insights into what information is still left at the PLD level once we know the property set or type set, respectively.
- normalized mutual information: We introduced this metric for measuring redundancy above. Accordingly, we can obtain insights how the redundancy levels are distributed over the PLDs.

4.2 Entropy of the Marginal Distributions, Expected Conditional Entropy and Normalized Mutual Information

We start by comparing the distribution of the marginal entropies of type sets and property sets in Figure 5 and Figure 6, respectively. We see that they follow a fairly similar distribution. In the both cases of the marginal entropies of type sets and property sets, there are more PLDs having higher entropy values. To better compare the two distributions, we look at the cumulative distribution of the normalized marginal

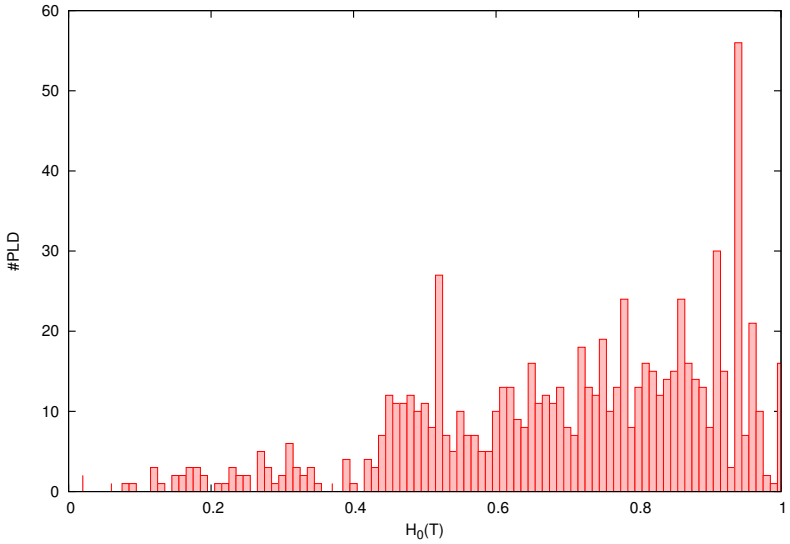


Figure 5 Distribution of marginal entropy values $H_0(T)$ for type sets over PLDs.

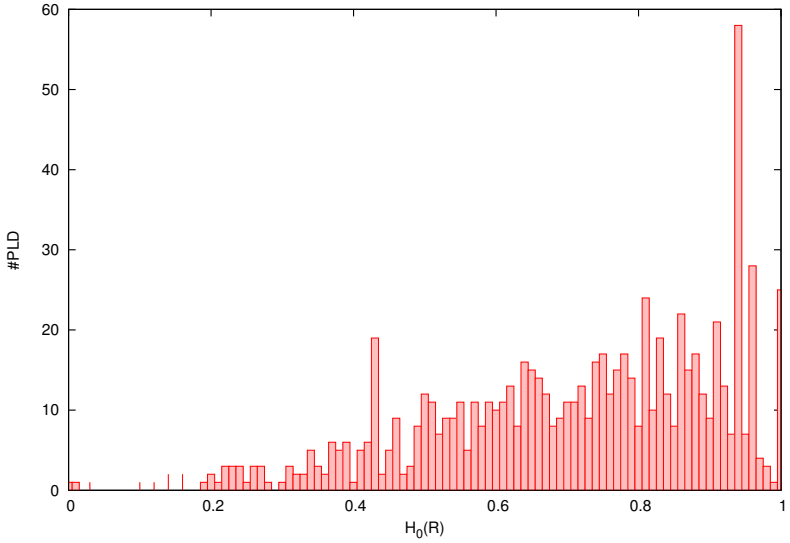


Figure 6 Distribution of marginal entropy values $H_0(R)$ for property sets over PLDs.

entropy values in Figure 7. In this visualization, we can see that the distributions are rather similar.

The next distribution we consider is the one of the expected cumulative entropy. The cumulative distribution of the expected conditional entropy for given properties is shown in Figure 8. The steep increase of the plot indicates that very many PLDs

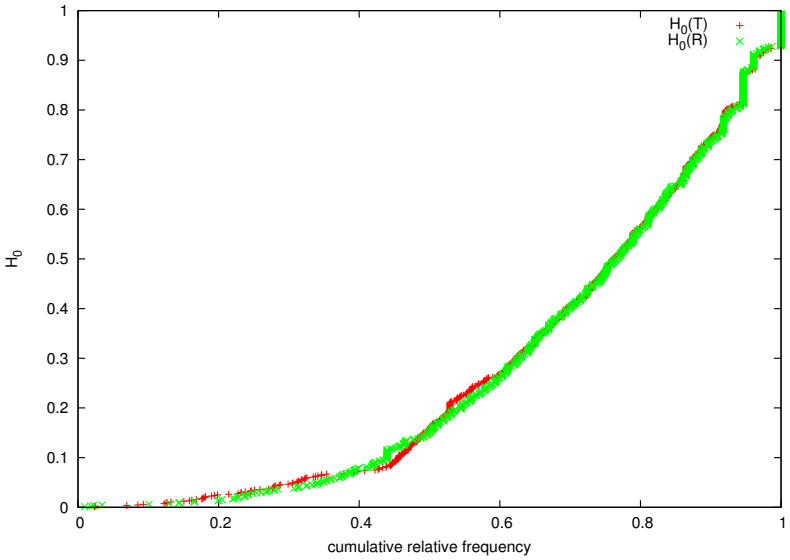


Figure 7 Cumulative distribution of the marginal entropy values for $H_0(R)$ and $H_0(T)$.

show a relatively low conditional entropy. This follows the general observation at global scale that a given set of attributes is indicative for the types of a resource. Approximately 20% of the PLDs have a conditional entropy of 0. Instead, when looking at the cumulative distribution of the expected conditional entropy for given types in Figure 9, we observe a slighter increase. Only about 6% of the PLDs have in this case a conditional entropy of 0. Also for higher conditional entropy values the cumulative distributions increases slower, which means that less PLDs have low values. Again, this observation is in line with the global trend of the types being less indicative for the properties. However, to be sure these observations are not an artefact of the many small data sets among the PLDs, we compared also the distribution of the expected conditional entropy values over the data set size. This is demonstrated in the scatter plots Figure 10 for the $H(R|T)$ values and in Figure 11 for $H(T|R)$. Both plots do not show strong clusters around singular values for small data sets. Thus, we conclude that the observations are genuine.

Finally, we have plotted the distribution of the normalized mutual information over the PLDs. Figure 12 shows how many PLDs share the same normalized MI value I_0. As one can see in Figure 12, the distribution tends to show an exponential curve with the major part of PLDs having a mutual information larger than 0.5. The top 100 largest PLDs make about 99,84% of all USUs observed in our data set (see Section 2.5). Thus, we also investigate the distribution of the normalized mutual information for the top 100 largest PLDs only. As one can see from Figure 13, the largest PLDs distribute quite well over the different MI values. Thus, one cannot say that large PLDs have a more homogeneous or less homogeneous use of common sets of RDF types and RDF properties for describing their entities.

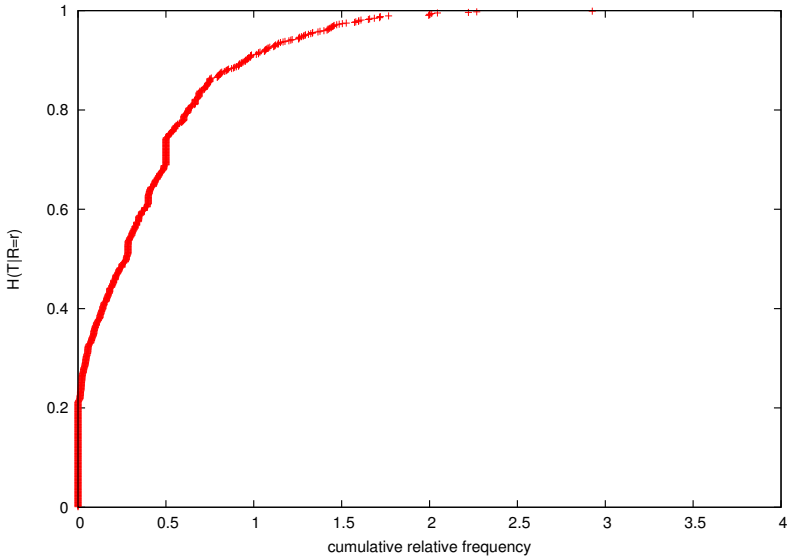


Figure 8 Cumulative distribution of the expected conditional entropy for $H(T|R)$.

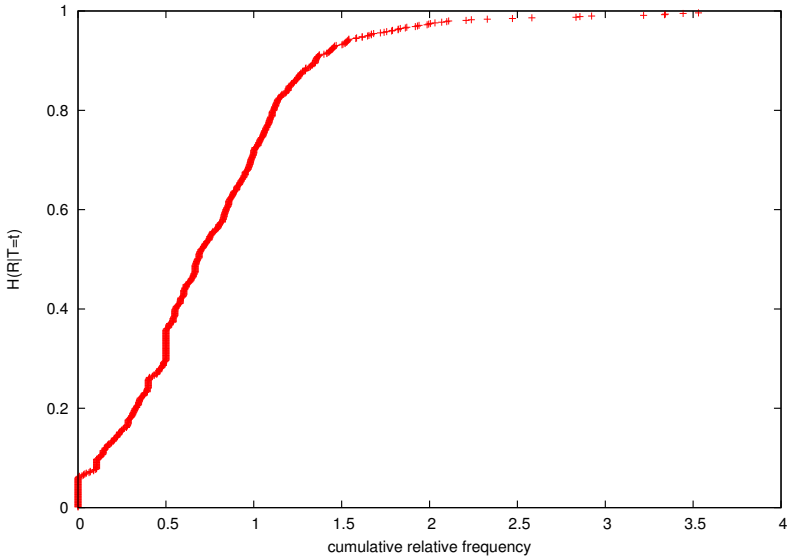


Figure 9 Cumulative distribution of the expected conditional entropy for $H(R|T)$.

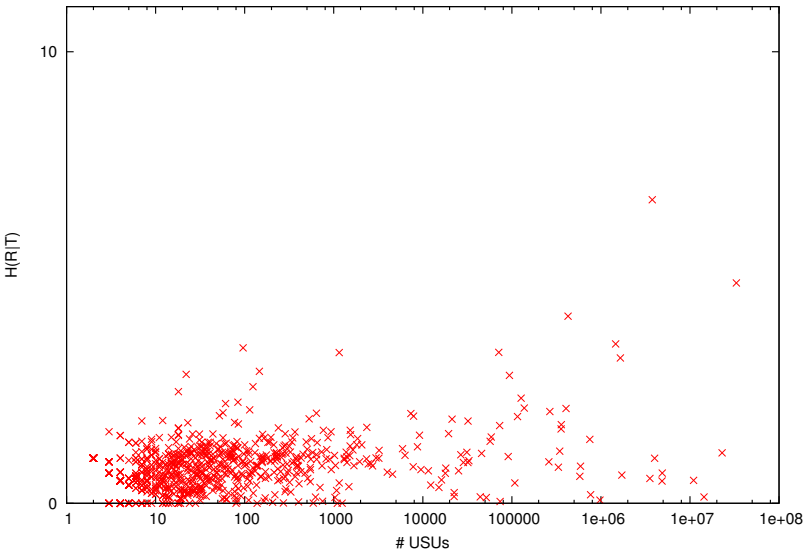


Figure 10 Distribution of the expected conditional entropies $H(R|T)$ depending on the size of the data set.

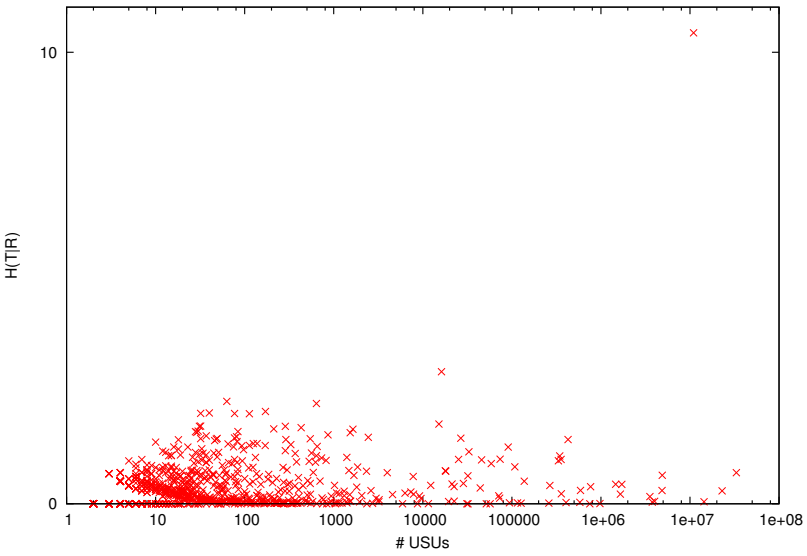


Figure 11 Distribution of the expected conditional entropies $H(T|R)$ depending on the size of the data set.

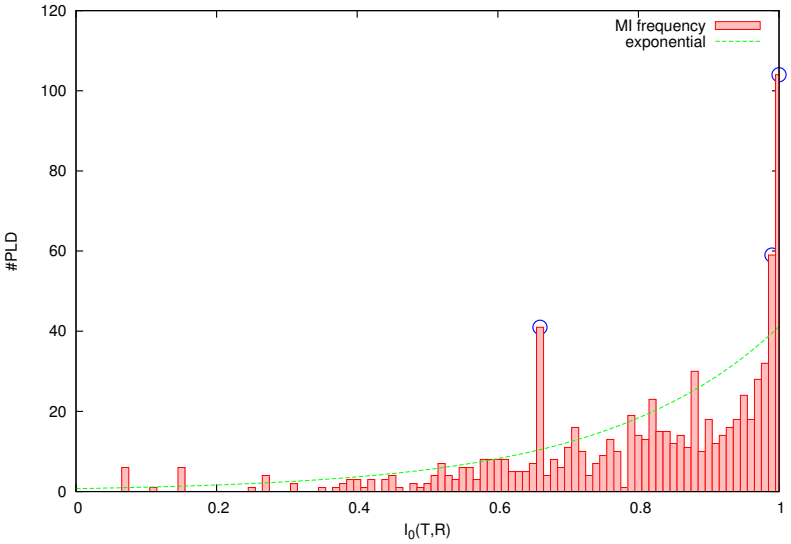


Figure 12 Distribution of the normalized MI values I_0 over the PLDs. The dotted trend line was used for outlier detection. The three outlier values are indicated with an embracing circle.

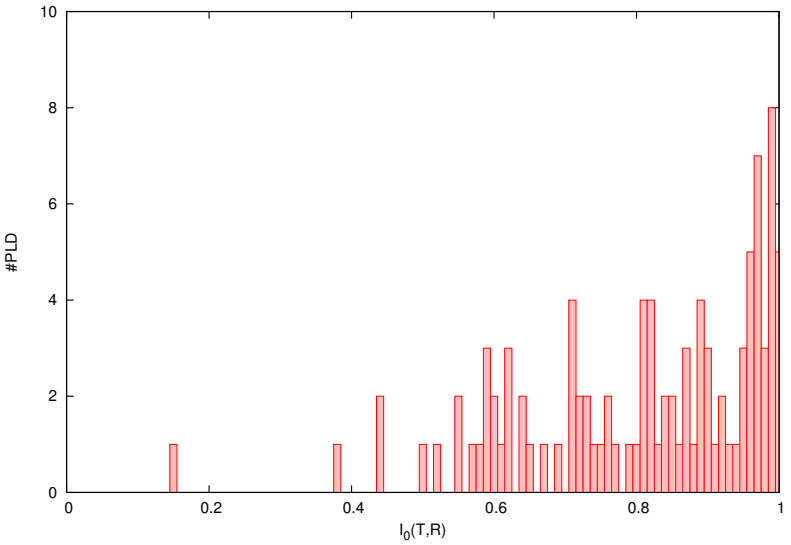


Figure 13 Distribution of the normalized MI values I_0 for the 100 largest PLDs.

4.3 Analysis of Outliers in the Distribution of Normalized Mutual Information

The distribution of normalized Mutual Information as shown in Figure 12 motivates for a further analysis. We can see a slight trend that higher values of I_0 are observed more often. Nevertheless, some values seem to be outliers. Given that the underlying analysis is based on actual data, this cannot be an error in the observations. However, there is a small possibility that these observations are caused by a sampling bias. A third explanation can be an artefact in the data itself. For PLDs with very little data (e. g., only one or two USUs) the combinatorial possibilities restrict the value that can be taken by the MI measure. In any case, it is worth to analyze the distribution of normalized mutual information for outliers and to further investigate them in order to understand what causes them.

To this end, we have performed an outlier analysis as a first step to identify the MI values for which we definitely observed a strong deviation from the general trend. To this end, we have first fitted an exponential curve through the data points. This curve is visualized in Figure 12. Second, we have measured the distance of the single observations to this curve. Using the interquartile range method [18], we have determined the outlier values. The interquartile range *IQR* is the distance between the first quartile ($Q1$) and the third quartile ($Q3$) of the distribution of deviations from the fitted curve. An observation is determined as outlier if for the distance d of the observation to the fitting curve holds: $d \geq Q3 + 1.5\mathit{IQR}$. Thus, we are only interested in the outliers above the fitting curve. Those correspond to PLDs sharing the same MI value unusually often.

As one can see from Figure 12, there are three outliers for the normalized mutual information values, namely 0.66, 0.99, and 1.00. These MI values are shared by three sets of PLDs. A mutual information of $I_0 = 0.66$ is shared by 41 PLDs, 59 PLDs have a MI value of $I_0 = 0.99$ in common, and 104 PLDs have a MI value of $I_0 = 1.00$. Table 4 shows for each outlier the top 10 PLDs with respect to the number of containing USUs. We can see that PLDs with an I_0 value of 0.66 contain only very few USUs. In particular only 4 of the PLDs provide information about more than 4 entities. Thus, this outlier is very likely an artefact of the smaller data sets. The MI level of 0.99 and 1.00, instead, are shared by much larger PLDs. These data sets model hundreds or thousands of entities. An explanation is that these data sets are based on a perfectly redundant schema, where the type definitions for the entities entail a complete description using always the same set of properties. An explanation for the I_0 value of 0.99 is simply the inaccuracies of our efficient stream-based schema computation. Such small deviations can easily occur, especially on larger data sets.

4.4 Discussion of the Results

Our analysis of the data sets on the level of PLDs provide some interesting results. First, the distributions of the conditional entropy along PLDs confirm the global analysis that the RDF types are less indicative than the properties to characterize the USUs. This is in particular interesting as the observation holds true for both the large as well as the small PLDs in terms of number of USUs they contain. Based on this

Table 4 Top 10 PLDs of outliers in the distribution of normalized mutual information values.

$I_0 = 0.66$		$I_0 = 0.99$		$I_0 = 1.00$	
PLD	# USU	PLD	# USU	PLD	# USU
harth.org	146	kasabi.com	974,307	lexvo.org	751,022
scot-project.net	44	codehaus.org	51,610	nytimes.com	57,072
lstadler.net	41	zbw.eu	32,165	opencalais.com	31,210
periapsis.org	8	advogato.org	11,539	esd-toolkit.eu	5,918
xrea.com	4	oszk.hu	8,721	bnf.fr	2,351
vub.ac.be	4	rainbowdash.net	2,730	heroku.com	835
vihart.com	4	debian.org	1,096	dewey.info	339
toxi.co.uk	4	robots.net	1,046	pbworks.com	272
sonnenburgs.de	4	planet-libre.org	642	xtrasgu.org	165
sideshowbarker.net	4	saz.im	553	foaf.me	131

observation and the fact that the top 100 PLDs make more than 99.5% of the data, one might be tempted to argue to limit future analyses on the basis of the top 100 PLDs. This might be an appropriate approach for data sets comparable to ours, i. e., data sets that are of large scale (more than one billion triples) and that have been obtained in a similar fashion like the BTC 2012 data set that is crawled from the web. Finally, the distribution of the normalized mutual information shows that for about 20% of the PLDs the RDF types and properties are entirely redundant and do not provide additional information to each other at all. Here, aggressive compression techniques might be applied to reduce the amount of information needed to store the data.

5 Analysis of Schema Structures over Vocabularies

A further level of schema information is contained in the vocabularies used for modelling data. Single vocabularies are typically designed for specific domains. Therefore, the number of vocabularies might be a hint towards the variety of domains covered in a PLD. On a more finegrained level, it is furthermore of interest for which ratio of the data the individual vocabularies are used on a PLD level.

This motivation leads to the questions we address regarding the use of vocabularies on a PLD level:

(a) What is the strength of individual vocabularies in specific PLDs?
(b) Which patterns can we observe about the use of vocabularies?

To address these questions, we will define a metric to capture the notion of vocabulary strength in a data set in Section 5.1. Based on this notion, we can answer the questions posed above. Please note: The analysis we perform in this section can in principle also be carried out on a global level, i. e., summarized over all PLDs. However, the expressiveness would be very low as the results would be blurred by the heterogeneous nature of our data set crawled from the web.

In Figure 14, we see that most data providers model their data using up to 10 vocabularies. The largest number of PLDs make use of six vocabularies. In the plot in Figure 14, we also included a Poisson distribution fitted to the observed data. Visual

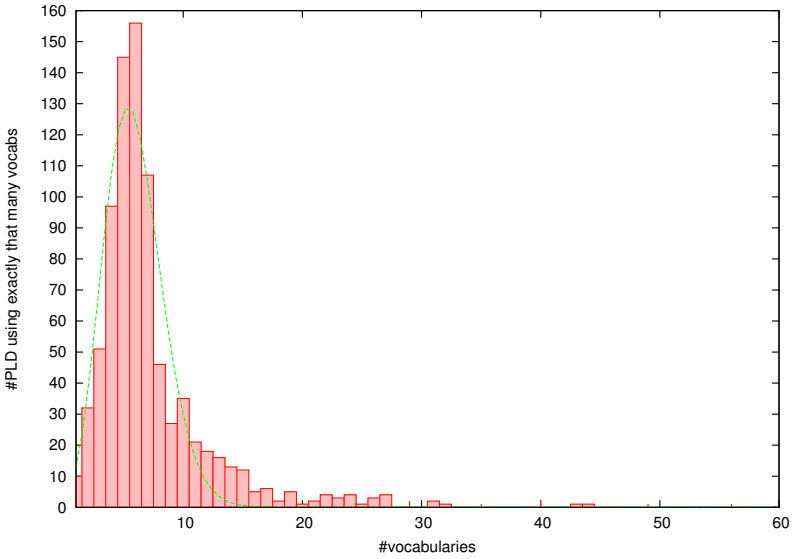


Figure 14 Distribution of the number of vocabularies used in PLDs. The fine dashed line represents a Poisson distribution fitted to the data.

inspection hints that a Poisson distribution seems to describe the data quite well. The λ parameter of the fitted distribution lies at 5.88.

5.1 Strength of Vocabularies in Distinct Pay-level Domains

To identify the impact and coverage of a vocabulary on a given PLD, we need to define a strength metric. The aim of this metric is to identify how important and central is a certain vocabulary for a given data set. We base our metric again on the concept of USUs. After all, the USUs provide a sense of how many entities are modelled in a data set. Therefore, they provide the right granularity for identifying the objects covered by a certain vocabulary.

Accordingly, we define the strength of a vocabulary in a data set to be the fraction of USUs which is described by at least one triple making use of this vocabulary. Please note, a data set DS is in our case the set of RDF triples provided by a single PLD. Formally, this leads to:

$$\mathit{strength}(V, DS) = \frac{|\{u \in DS : u \textit{ is described using } V\}|}{N_{DS}} \tag{18}$$

where N_{DS} is the number of USUs in data set DS.

Thus, if, for instance, 7 out of 10 USUs in a data set are described using the FOAF vocabulary, this would lead to a strength of FOAF in this data set of 0.7.

Note, that this definition of strength is an absolute metric in relation to other vocabularies, as a single USU can be described by various vocabularies. Therefore, there is no upper limit for the sum of strength values over all vocabularies. Effectively,

Table 5 Top 10 largest PLDs by numbers of USUs and their redundancy values.

PLD data sets		**Number of USUs**	**Redundancy I_0**
(1)	dbpedia.org	33,191,714	0.6225
(2)	freebase.com	22,967,599	0.8598
(3)	livejournal.com	14,373,588	0.9428
(4)	dbtropes.org	10,983,136	0.5903
(5)	data.gov.uk	4,888,610	0.8074
(6)	legislation.gov.uk	4,850,236	0.8970
(7)	identi.ca	4,004,911	0.9723
(8)	ontologycentral.com	3,773,117	0.8788
(9)	opera.com	3,547,299	0.8667
(10)	loc.gov	1,714,943	0.7636

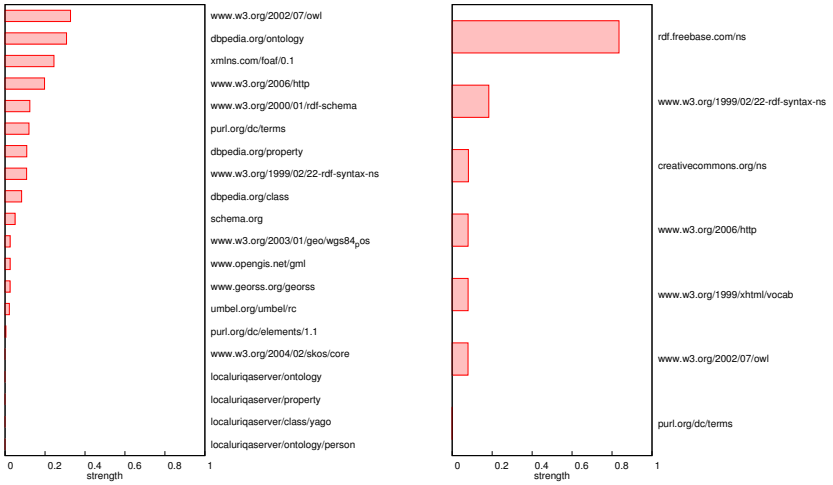
even if a vocabulary reaches a strength value of 0.9 it does not necessarily have to be the strongest vocabulary in the data set.

5.2 Dominating Vocabularies in Selected Pay-level Domains

After having defined the strength metric, we look at the strength values of the vocabularies at a PLD level. Given that this involves looking into single data sources only, we focussed our analysis on the larger PLDs. We present some of the more characteristic observations made in this context. In particular, we will show the results for the top-10 largest PLDs as listed in Table 5. The strength of individual vocabularies for these PLD is shown in Figures 15 to 19. It can be seen that vocabularies are used and mixed in quite different ways and patterns by single data providers.

For instance, in Figure 16(a) we see a strong focus on three vocabularies which are used to describe nearly all entities modelled by USUs. These three vocabularies are RDF, RDF Schema (RDFS), and FOAF. RDF and RDFS are needed to provide schema related information, FOAF is chosen as domain specific vocabulary to model people and social relations. To a far lower degree, we see use of the DC Terms vocabulary and other properties and classes used to model specific information, e.g. geographic locations via the WGS84 vocabulary. A similar, focussed use of few vocabularies can also be observed on Freebase, DBTropes, Identi.ca, and Ontology Central. These data providers consistently describe nearly all USUs using the same set of vocabularies and seem to add little optional information about entities where appropriate. This is also reflected by tendentiously higher redundancy values in the schema of these PLDs.

A different and more varied use of vocabularies can be observed on DBpedia in Figure 15(a). This is reflected in the overall higher number of vocabularies as well as overall lower strength values. Also here, well established vocabularies like OWL, RDF, RDFS, DBpedia, FOAF, and DC Terms are used to describe entities. However, none of them has a very high strength value. A similar combination of vocabulary strength can be observed on Data.gov, Legislation.gov, and loc.gov. In turn, the redundancy values are tendentiously lower in the schema of these PLDs.



(a) dbpedia.org (b) freebase.com

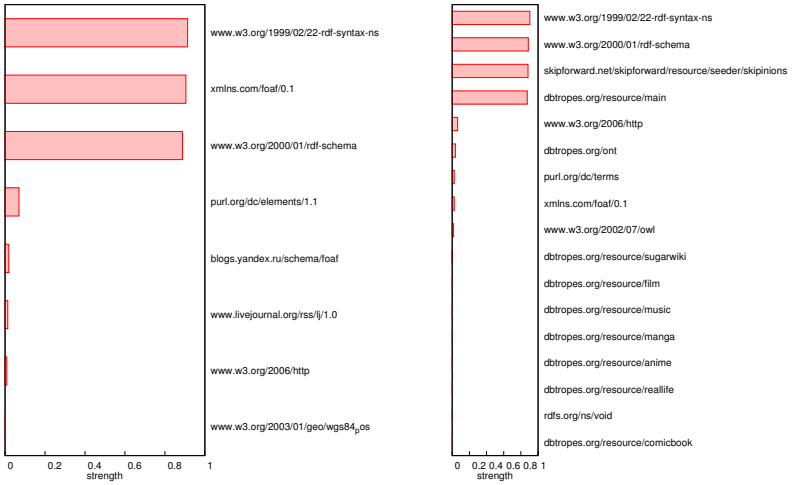
Figure 15 Strength of vocabularies on dbpedia.org and freebase.com.

A second observation regards the vocabularies themselves. Most data providers make strong use of a few selected well established vocabularies. These cover the W3C vocabularies RDF, RDFS, OWL, as well as commonly known vocabularies such as FOAF, DC Terms, or Cube[11], a semantic version of SDMX and an data provider specific vocabulary for modelling information from Eurostat. Depending on the domain of the data set, several or even all of these vocabularies appear in all of the PLDs. A second trend is to incorporate own vocabularies such as the DBPedia ontology, Freebase Schema, and DBTropes concepts. Finally, a variety of other vocabularies are used to a small degree (low strength values) to model specific information which is not required for all entities.

In Figure 20, we finally look at the strength of vocabularies used on kasabi.com and lexvo.org—the two largest PLDs we identified as outliers observed in Section 4.3. Both demonstrate a very regular use of vocabularies at very regular levels of strength. This perfectly matches the observation of high redundancy values I_0 of 0.99 and 1.0. The data seems to be modelled following a perfect schema. The slight deviation in kasabi.com is probably due to the efficient stream-based processing of the data which bears the risk of loss of accuracy in the schema information.

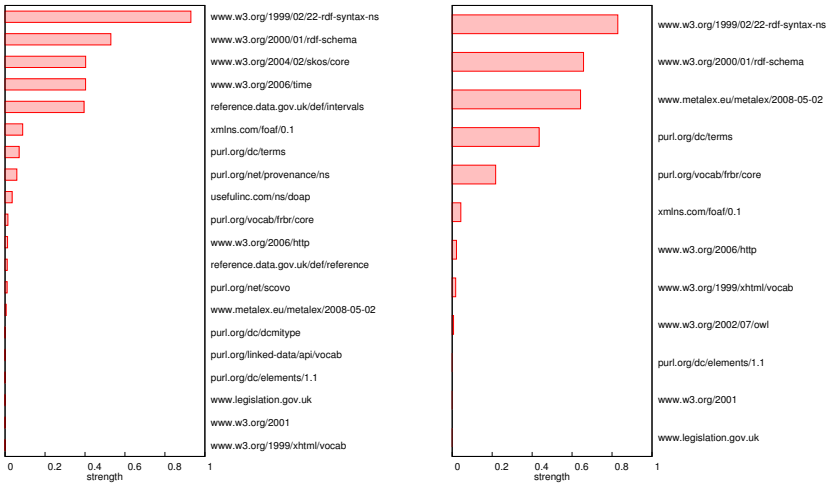
To get a deeper insight into the observations, we looked not only for the strength of the vocabularies per PLD, but also searched for typical combinations of vocabulary uses. Therefore, we employed the Apriori algorithm [1] for mining frequent itemsets over the used vocabularies. To this end, we considered each USU as transaction and modeled the vocabularies used to describe the USU as items associated with these transactions. Mining frequent itemsets allows for finding patterns of vo-

[11] `http://www.w3.org/TR/2013/CR-vocab-data-cube-20130625/` accessed: 11 September 2013



(a) livejournal.com (b) dbtropes.org

Figure 16 Strength of vocabularies on livejournal.com and dbtropes.org.

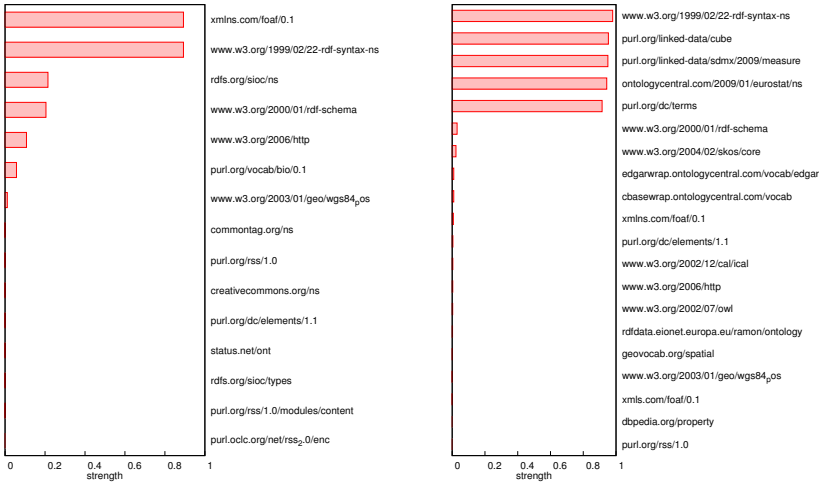


(a) data.gov.uk (b) legislation.gov.uk

Figure 17 Strength of vocabularies on data.gov.uk and legislation.gov.uk.

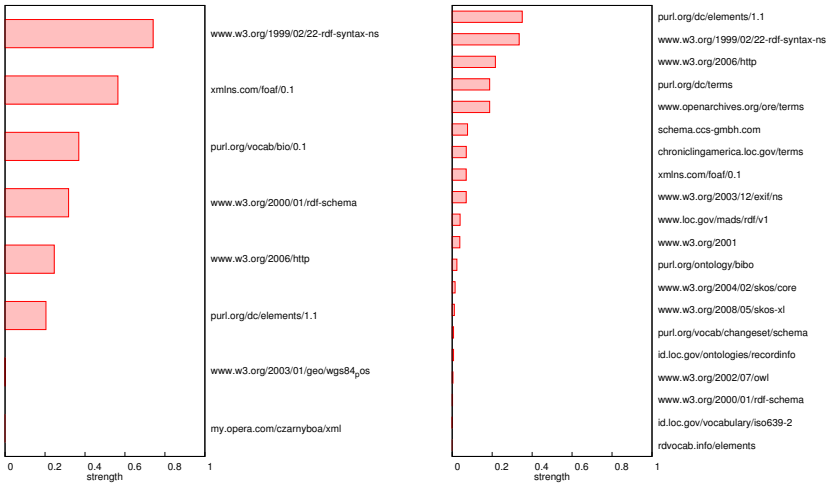
cabulary combinations which appear very often and cannot be explained as random co-occurrences.

For example, on Ontology Central we have seen a few strong vocabularies dominating the entire data set. Here, we find some interesting patterns of using vocabularies. More than 90% of the USUs are described by the vocabularies RDF, DC Terms,



(a) identi.ca (b) ontologycentral.com

Figure 18 Strength of vocabularies on identi.ca and ontologycentral.com.

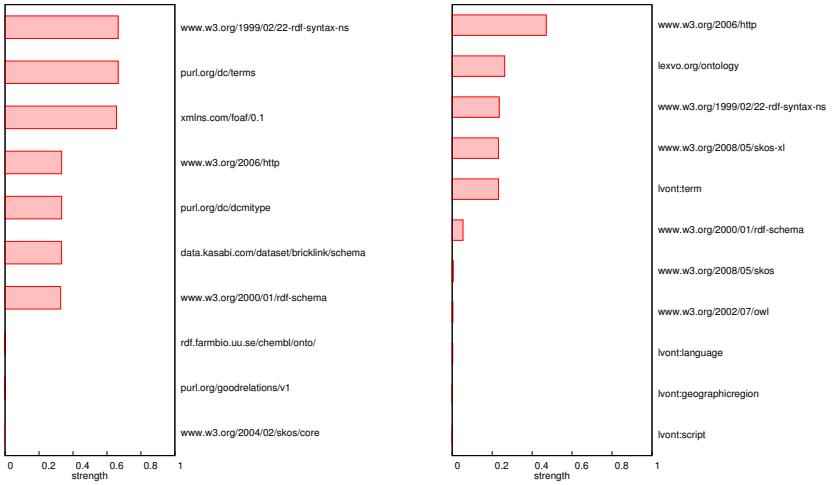


(a) opera.com (b) loc.gov

Figure 19 Strength of vocabularies on opera.com and loc.gov.

Data Cube. Given that Ontology Central is an aggregation and conversion service for statistical data, this seems plausible. Most data is automatically converted from proprietary or non-semantic data formats.

In contrast, for DBpedia we already observed a much wider mix of vocabularies. Thus, also the frequent patterns are much less strong. Hardly any pattern applies to



(a) kasabi.com (b) lexvo.org

Figure 20 Strength of vocabularies of kasabi.com and lexvo.org as the two largest PLDs in the outliers of the distribution of normalized mutual information.

more than 10% of the USUs and no pattern affects more than 12% of the USUs. The largest typical combination we identified in fact uses the stronger vocabularies of RDF, RDFS, OWL, FOAF, and DC Terms.

The patterns in kasabi.com as one of the outliers in Section 4.3 reflect exactly the two levels of vocabulary strength shown in Figure 20(a). Frequent combinations of vocabularies affect approximately 66% of the USUs like the combinations of RDF, DC Terms, and FOAF. Other, less frequent patterns, reflect approximately 32% of the USUs and exhibit vocabulary combinations such as RDF, RDFS, DC Terms, FOAF, and a vocabulary defined by kasabi.com itself. These two levels of patterns can be explained well with the different types of entities modelled in this data source, which call for different vocabularies to describe their properties.

5.3 Discussion of the Results

Based on the analysis about vocabulary use by data providers, we can obtain some interesting insights. First of all, most data providers use between 3 and 8 vocabularies. On average, approximately six vocabularies are combined in a data source. This aligns well with the recommendations and best practices on modeling Linked Data using a good mix of vocabularies and observations in previous analysis of Linked Data [19].

Regarding the strength of individual vocabularies on specific PLDs, we observed two trends. Some data providers focus on a few vocabularies which are consistently used to describe nearly all modelled entities. In practice this means, that nearly all entities are provided with types and properties from the same vocabulary. This indi-

cates a strong schema background of the data and the entities can well be modelled using a standard schema. This hypothesis is also supported by the observation that in most of these cases the schema information of type sets and property sets is highly redundant. A different pattern lies in the mix of many vocabularies and the observation that there is no predominant vocabulary which is used to describe a high fraction of all entities. In this case the mix of vocabularies indicates a rather free modelling process where each entity is described individually using the vocabularies suitable to model the available information. Accordingly, there is tendentiously a less strong correlation between the type sets and property sets. Typically, this can be seen in lower values of I_0 on these PLDs.

Finally, the last observation is that some standardized and well established vocabularies are used frequently and across several data sources. Also other, domain specific vocabularies that are introduced by the data providers themselves are typically used strongly on their own data. Vocabularies which model more specific information (e. g., geo-locations), instead are used as needed and typically to a far lower degree. However, this perfectly makes sense and implements the paradigm of combining and re-using vocabularies as needed to model the available information.

6 Related Work

The related work is structured as follows: We start with related articles that conduct structural and/or semantic analysis of LOD for the purpose of building statistics about the nature of the data and the LOD network. Subsequently, we discuss related work that builds statistics over LOD in order to check for the compliance of the data to guidelines or best practices. Finally, we investigate analyses of LOD that are carried out in the context of query optimization.

6.1 Statistical Analysis of Linked Open Data

Several tools aim at providing statistics for the LOD cloud. The tool make-void[12] computes statistics for a given RDF file in the Vocabulary of Interlinked Data sets (VoID) format [2]. These statistics usually contain information about the total number of triples, classes, properties, instances for each class, the use of each property, and the number of triples that link a subject on one domain to an object on another domain. Another framework for generating statistics on RDF data is RDFStats[13]. In contrast to make-void, the RDFStats framework can also operate on SPARQL endpoints and uses a different vocabulary for its statistics. RDFStats' statistics are based on SCOVO [16], a framework for modelling and publishing statistical data. Bizer et al. have conducted an analysis of the structural data extracted from a common web crawl[14]. The authors investigated among others the different formats of structural data found in common web pages, the top-k domains providing semantic data, as well as

[12] `https://github.com/cygri/make-void` accessed: 21 March, 2013
[13] `http://rdfstats.sourceforge.net/` accessed: 21 March, 2013
[14] `http://www.webdatacommons.org/` accessed: 21 March, 2013

the most frequently used classes and properties per domain and an entity count. In addition, different matrices are provided containing information about the correlation, i. e., number of instances where a specific class occurs together with a specific property.[15] However, the authors do not consider the general case of co-occurrence of arbitrary sets of classes and sets of properties like it is done in this work. In addition, also no statistics based on information theoretical approaches are considered in the related work. The LODStats framework [3] computes 32 different statistics on Linked Open Data such as those covered by VoID. The tool provides descriptive statistics such as the frequencies of property usage and datatype usage the average length of literals, or counting the number of namespaces appearing at the subject URI position. LODStats operates on single triple patterns, i.e., it does not provide statistics of, e.g., star patterns or other (arbitrary) graph patterns. However, it covers more complex schema-level characteristics like the RDFS subclass hierarchy depth. Overall, analyses of the correlating use of different properties, RDF types, or the common appearance of properties and types like we investigate is out of their scope.

Ding and Finin [8] have applied different metrics over a crawl of approximately 300 million triples to characterize the structure of semantic data on the web. Different analyses have been conducted on the level of so-called Semantic Web documents (SWDs), i. e., the responses like a single RDF document that is returned by providers of semantic data when the client dereferences a specific URL. For example, the authors estimate the total size of the Semantic Web in 2006 using an industry search engine, extracting the number of SWDs per top-level domain, as well as the number of documents per pay-level domain, the size of SWDs in terms of number of triples, the age of the SWDs based on the last-modified time extracted from the HTTP response header, and others. As these analyses are on the granularity of entire SWDs, this part of the investigations complement our work that is on the level of property sets and type sets found in the LOD. Closer related to our work is, however, the further investigations by Ding and Finin [8] such as the complexity of USUs. Here, the authors count the number of terms, i. e., number of properties and classes used in a single USU. The observation is similar to ours that the data follows a power law distribution. Interesting are deviations at the head and tail of the distribution. The authors interpret this as a result of which complexity of USUs is useful and which still manageable. When there are only very few terms in an USU, its definition does not contain much domain-specific information. While very large USUs (one observation had more than 1000 triples) are not manageable any more. However, the authors have not distinguished explicitly the use of properties and classes in USUs. Thus, they have also not analyzed any kind of correlation between properties and classes like we do. In addition, the authors have analyzed the use of terms specified in RDFS versus OWL. Not surprisingly, the OWL namespace occurs in only very few SWDs (8%) and consequently in very few USUs (7%). RDFS is used in 47% of all SWDs and 37% of all USUs. Thus, 45% of all SWDs and over 50% of all USUs do not specify any vocabulary terms using RDFS or OWL. As the authors do not explicitly refer to the RDF namespace, they do not investigate the use of rdf:type and rdf:Property that

[15] `http://webdatacommons.org/2012-08/stats/how_to_get_the_data.html#toc2` accessed: 21 March, 2013

would have revealed valuable information about how the USUs are actually defined. Our investigations explicitly refer to the use of rdf:type and rdf:Property in the RDF namespace.

Cheng and Qu [6] have analyzed the different distributions and statistics on a data set obtained from a Semantic Web crawl of the Falcons search engine [5]. The authors have computed among others the cumulative distribution of pay-level domains versus the number of RDF documents per domain and the cumulative distribution of RDF documents versus RDF triples per document providing insight about the size of pay-level data sources and RDF documents. In addition, the cumulative distribution of number of terms defined in vocabularies is contrasted as well as the number of properties versus classes in a vocabulary. Further analyses conducted in the work of Cheng and Qu are, e. g., the in-degree and out-degree of terms as well as the reachability of terms. Those analyses are complement to our work.

Ding et al. [10] have conducted an extensive analysis of the sameAs networks in LOD on the Billion Triple Challenge dataset 2010. Computations on the dataset include determining the number of weakly connected components, the average path lengths and maximum path length in the components, as well as the topological nature of the components. Most interesting investigation though is the analysis of the connectedness of the data publishers through OWL's sameAs relations. By splitting up the data on pay-level domains, the domains have been compared pairwise by determining the k-most frequently used RDF types. In addition, the authors considered the overlap between two classes by comparing their occurrence as explicit RDF type statement of a common subject URI versus their occurrence for two different subject URIs but which are connected via a sameAs relation. This analysis is insofar different from our analysis, as the analysis are based merely on the sameAs network of the Billion Triple Challenge dataset. Our investigations are considering any kind of properties and RDF types for a common subject URI.

The statistical analyses described above are focusing on LOD and thus on RDF data. Only few investigations have been conducted on semantically richer semantic data like ontologies specified in the Web Ontology Language (OWL). OWL is explicitly considered, e. g., by Ding and Finin [8] but this analysis only ran alongside. An example of a statistical analysis on the use of OWL on the Semantic Web has been conducted by Wang et al. [29]. They have obtained almost 1,300 OWL ontologies and RDFS vocabularies from the Semantic Web search engine Swoogle [9]. The authors have counted the ontologies and vocabularies, e. g., along their species such as belonging to the RDFS, OWL Lite, OWL DL, or OWL Full. As a more fine grained analysis of expressiveness, the authors have also binned the ontologies and vocabularies into different description logics classes. At the current state, our analysis also focuses on RDF and RDFS vocabularies. It remains future work to extend it to more expressive ontology languages like OWL.

6.2 Analysis for Compliance to Linked Open Data Principles

Hogan et al. have conducted an empirical study to investigate the conformance of Linked Data sources with 14 different Linked Data principles [19]. As metric, the au-

thors apply the number of unique namespaces used by the respective data providers and provide a ranked list in terms of top-5 and bottom-5 data providers. Among others, the authors analyzed how different classes and properties of vocabularies defined at one data source are re-used and mixed by other Linked Data providers. In contrast, the analysis of the correlation of class terms and property terms of different (or the same) vocabularies done here is agnostic to the actual source the Linked Data originates from. Bizer et al. have recently analyzed the joined occurrence of a single class with a single property on the structured data extracted from a large web crawl[16]. Lorey et al. [22] developed a frequent item set approach over properties for the purpose of detecting appropriate and diverging use of ontologies. None of these works addresses information theory metrics as it is done in the paper at hand. The application of information theoretic measures on RDF data is addressed in [23]. However, the analysis there is focussing on a different level of schema re-use of concepts and does not consider any property information. In addition, although both class terms and property terms are taken into account by the authors' metric, they do not differentiate between them. Finally, in a similar fashion to Hogan et al., also Bizer et al. have analyzed the LOD cloud for its compliance with the Linked Data principles using nine different criteria. They provide statistics such as the LOD cloud diagram or the inter-linkage of the datasets in the LOD cloud.[17] The authors have computed how many pay-level domains use a specific vocabulary in total and averaged over the total number of domains. Our vocabulary analysis reveals this information in comparison to the previous results. In addition, our work extends the previous analysis by providing information about the strength of the vocabulary in the pay-level domains, i. e., the percentage how dominating a vocabulary is for describing the data in a specific domain.

6.3 Analysis of Linked Open Data for Query Optimization

One application where schema information can be of value is query optimization. Neumann and Moerkotte [25] employ so-called *characteristic sets*, which basically classify RDF resources by the correlation of their (outgoing) predicate links. Knowledge about these sets allows for quite precise estimates of the result cardinality of join operations. Further insights into the correlation between properties in an RDF graph were not necessary. Neither were explicit schema information provided in form of RDF types considered. A similar approach is presented by Maduko et al. [24]. Here the focus was on efficient approaches to estimate subgraph frequencies in a graph database. This subgraph frequency information is then used for conducting efficient queries on the graph database. In their work, Maduko et al. use both implicit schema information and explicit schema information. However, they do not determine the cardinality of intermediate join results of the two schema information sources for executing these queries. They used modifications of common pattern mining algorithms such as gSpan [30] to discover and count frequency of all subgraphs of a

[16] `http://webdatacommons.org/` accessed: 21 March, 2013
[17] `http://lod-cloud.net/state/` accessed: 21 March, 2013

specific length and applied two different pruning techniques, namely Maximal Dependency Tree and Pattern Tree. Although the authors investigate properties as well as RDF types, no insight in their correlation is given like it is done in this work. Harth et al. [15] propose an approximative approach to optimize queries over multiple distributed LOD sources. They build a QTree index structure over the sources, which is used to determine the contribution of the single sources to the query results.

7 Summary and Future Work

In this paper, we have proposed a method and metrics for conducting in-depth analyses of schema information on Linked Open Data at three levels of granularities: unique subject URIs (USUs), pay-level domains (PLDs), and vocabularies. In the first step, we have addressed the question of dependencies between the types of resources and their properties on the level of USUs. Based on the five segments of the BTC 2012 data set, we have computed various entropy measures as well as mutual information. In conclusion, we observe a trend of a reasonably high redundancy between the types and properties attached to resources. As more detailed conclusion, we can derive that the properties of a resource are rather indicative for the type of the resource. In the other direction, the indication is less strong. However, this observation is not valid for all sources on the LOD cloud. In conclusion, if the application and data domain is not known, it is necessary to capture both: explicit and implicit schema information.

In a second step, we have split up the large-scale BTC 2012 data set along individual PLDs. For the resulting 840 PLD data sets, we have investigated among others the distribution of the normalized mutual information. This distribution shows that about 20% of the PLDs share a common value of normalized mutual information that is 0.99 or higher. This means in conclusion that for about 20% of the PLDs, the types of the USUs may be omitted but it would be possible to still fully explain the graph. Please note that the characteristics of the USUs may also be fully explained by keeping the types and removing the properties. However, when removing the properties one would also remove the instance-level relationships between resources and USUs, respectively.

Finally, we have investigated the strength distribution of vocabularies on the level of PLDs. Thus, we can state how many of the USUs contained in a PLD are defined by at least one triple using a particular vocabulary. The results are interesting and show two general trends of how Linked Data is modelled and published: Either the data providers apply a strong schematic design of their data sets, i. e., they use the vocabularies very consistently all over the USUs. Or the data providers apply a mix of a wider range of vocabularies to model and publish their data, i. e., the USUs in their domain. Further, we have investigated if there are specific patterns that occur when combining the vocabularies to describe the USUs.

As future extensions of our analyses so far, we plan to investigate the strength distribution of the vocabularies on the level of USUs. Thus, we like to understand which vocabularies dominate in the definition of the individual entities. Another analysis we like to conduct is to investigate the number of different vocabularies used in the PLDs

in relation to the size of PLDs measured by number of USUs they contain. Finally, it will be interesting to further characterize the vocabularies used in the PLDs that share a very high value of normalized mutual information.

Acknowledgements The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013), REVEAL (Grant agree number 610928). Parts of the computations and experiments were conducted on resources provided by the HPI Future SOC Lab.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases, VLDB '94, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1994). URL http://dl.acm.org/citation.cfm?id=645920.672836
2. Alexander, K., Cyganiak, R., Hausenblas, M., Zhao, J.: Describing linked datasets with the void vocabulary. http://www.w3.org/TR/void/. (accessed 9 March 2013)
3. Auer, S., Demter, J., Martin, M., Lehmann, J.: Lodstats – an extensible framework for high-performance dataset analytics. In: A. Teije, J. Völker, S. Handschuh, H. Stuckenschmidt, M. d'Acquin, A. Nikolov, N. Aussenac-Gilles, N. Hernandez (eds.) Knowledge Engineering and Knowledge Management, *Lecture Notes in Computer Science*, vol. 7603, pp. 353–362. Springer Berlin Heidelberg (2012). DOI 10.1007/978-3-642-33876-2_31
4. Bizer, C.: The emerging web of linked data. Intelligent Systems, IEEE **24**(5), 87–92 (2009)
5. Cheng, G., Ge, W., Qu, Y.: Falcons: searching and browsing entities on the semantic web. In: Proceedings of the 17th international conference on World Wide Web, WWW '08, pp. 1101–1102. ACM, New York, NY, USA (2008). DOI 10.1145/1367497.1367676. URL http://doi.acm.org/10.1145/1367497.1367676
6. Cheng, G., Qu, Y.: Term dependence on the semantic web. In: Proceedings of the 7th International Conference on The Semantic Web, ISWC '08, pp. 665–680. Springer-Verlag, Berlin, Heidelberg (2008). DOI 10.1007/978-3-540-88564-1_42. URL http://dx.doi.org/10.1007/978-3-540-88564-1_42
7. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley-Interscience (1991)
8. Ding, L., Finin, T.: Characterizing the semantic web on the web. In: The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4273, pp. 242–257. Springer (2006)
9. Ding, L., Finin, T.W., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V., Sachs, J.: Swoogle: a search and metadata engine for the semantic web. In: CIKM. ACM (2004)
10. Ding, L., Shinavier, J., Shangguan, Z., McGuinness, D.L.: Sameas networks and beyond: Analyzing deployment status and implications of owl: sameas in linked data. In: The Semantic Web - ISWC 2010 - 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part I, *Lecture Notes in Computer Science*, vol. 6496, pp. 145–160. Springer (2010)
11. Gottron, T., Knauf, M., Scheglmann, S., Scherp, A.: A Systematic Investigation of Explicit and Implicit Schema Information on the Linked Open Data Cloud. In: ESWC'13: Proceedings of the 10th Extended Semantic Web Conference (2013). To appear
12. Gottron, T., Pickhardt, R.: A detailed analysis of the quality of stream-based schema construction on linked open data. In: CSWS'12: Proceedings of the Chinese Semantic Web Symposium (2012). To appear
13. Gottron, T., Scherp, A., Krayer, B., Peters, A.: Get the google feeling: Supporting users in finding – relevant sources of linked open data at web-scale. In: Semantic Web Challenge, Submission to the Billion Triple Track (2012)
14. Gottron, T., Scherp, A., Krayer, B., Peters, A.: LODatio: Using a Schema-Based Index to Support Users in Finding Relevant Sources of Linked Data. In: K-CAP'13: Proceedings of the Conference on Knowledge Capture (2013)

15. Harth, A., Hose, K., Karnstedt, M., Polleres, A., Sattler, K.U., Umbrich, J.: Data summaries for on-demand queries over linked data. In: WWW, pp. 411–420. ACM (2010)
16. Hausenblas, M., Halb, W., Raimond, Y., Feigenbaum, L., Ayers, D.: Scovo: Using statistics on the web of data. In: The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009, Heraklion, Crete, Greece, May 31-June 4, 2009, Proceedings, *Lecture Notes in Computer Science*, vol. 5554, pp. 708–722. Springer (2009)
17. Heath, T., Bizer, C.: Linked Data: Evolving the Web Into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool (2011)
18. Hinkle, D., Wiersma, W., Jurs, S.: Applied statistics for the behavioral sciences. Applied Statistics for the Behavioral Sciences. Houghton Mifflin (2003)
19. Hogan, A., Umbrich, J., Harth, A., Cyganiak, R., Polleres, A., Decker, S.: An empirical survey of linked data conformance. Web Semantics: Science, Services and Agents on the World Wide Web **14**(0), 14 – 44 (2012). DOI 10.1016/j.websem.2012.02.001
20. Konrath, M., Gottron, T., Scherp, A.: Schemex – web-scale indexed schema extraction of linked open data. In: Semantic Web Challenge, Submission to the Billion Triple Track (2011)
21. Konrath, M., Gottron, T., Staab, S., Scherp, A.: Schemex—efficient construction of a data catalogue by stream-based indexing of linked data. Web Semantics: Science, Services and Agents on the World Wide Web **16**(5), 52 – 58 (2012). DOI 10.1016/j.websem.2012.06.002. URL `http://www.sciencedirect.com/science/article/pii/S1570826812000716`. The Semantic Web Challenge 2011
22. Lorey, J., Abedjan, Z., Naumann, F., Böhm, C.: Rdf ontology (re-) engineering through large-scale data mining. In: Semantic Web Challenge (2011)
23. Luo, X., Shinavier, J.: Entropy-based metrics for evaluating schema reuse. In: A. Gómez-Pérez, Y. Yu, Y. Ding (eds.) The Semantic Web, *Lecture Notes in Computer Science*, vol. 5926, pp. 321–331. Springer Berlin Heidelberg (2009). DOI 10.1007/978-3-642-10871-6_22
24. Maduko, A., Anyanwu, K., Sheth, A., Schliekelman, P.: Graph summaries for subgraph frequency estimation. In: S. Bechhofer, M. Hauswirth, J. Hoffmann, M. Koubarakis (eds.) The Semantic Web: Research and Applications, *Lecture Notes in Computer Science*, vol. 5021, pp. 508–523. Springer Berlin Heidelberg (2008). DOI 10.1007/978-3-540-68234-9_38
25. Neumann, T., Moerkotte, G.: Characteristic sets: Accurate cardinality estimation for rdf queries with multiple joins. In: Proceedings of the 27th International Conference on Data Engineering, ICDE 2011, April 11-16, 2011, Hannover, Germany, pp. 984–994 (2011)
26. Schaible, J., Gottron, T., Scheglmann, S., Scherp, A.: LOVER: Support for Modeling Data Using Linked Open Vocabularies. In: LWDM'13: 3rd International Workshop on Linked Web Data Management (2013). To appear
27. Scheglmann, S., Gröner, G., Staab, S., Lämmel, R.: Incompleteness-aware programming with rdf data. In: E. Viegas, K. Breitman, J. Bishop (eds.) Proceedings of the 2013 Workshop on Data Driven Functional Programming, DDFP 2013, Rome, Italy, January 22, 2013, pp. 11–14. ACM (2013)
28. Shannon, C.: A mathematical theory of communication. Bell System Technical Journal **27**, 379–423 and 623–656 (1948)
29. Wang, T.D., Parsia, B., Hendler, J.A.: A survey of the web ontology landscape. In: The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4273, pp. 682–694. Springer (2006)
30. Yan, X., Han, J.: gspan: Graph-based substructure pattern mining. In: ICDM, pp. 721–724. IEEE Computer Society (2002)
31. Yao, Y.: Information-theoretic measures for knowledge discovery and data mining. In: Karmeshu (ed.) Entropy Measures, Maximum Entropy Principle and Emerging Applications, *Studies in Fuzziness and Soft Computing*, vol. 119, pp. 115–136. Springer Berlin Heidelberg (2003). DOI 10.1007/978-3-540-36212-8_6