Labels in the Web of Data

Basil Ell, Denny Vrandečić, and Elena Simperl

KIT
Karlsruhe, Germany
{basil.ell,denny.vrandecic,elena.simperl}@kit.edu

Abstract. Entities on the Web of Data need to have labels in order to be exposable to humans in a meaningful way. These labels can then be used for exploring the data, i.e., for displaying the entities in a linked data browser or other front-end applications, but also to support keyword-based or natural-language based search over the Web of Data. Far too many applications fall back to exposing the URIs of the entities to the user in the absence of more easily understandable representations such as human-readable labels. In this work we introduce a number of label-related metrics: completeness of the labeling, the efficient accessibility of the labels, unambiguity of labeling, and the multilinguality of the labeling. We report our findings from measuring the Web of Data using these metrics. We also investigate which properties are used for labeling purposes, since many vocabularies define further labeling properties beyond the standard property from RDFS.

Keywords: Web of Data, labels, human interfaces, data quality

1 Introduction

A growing number of applications is expected to use the Web of Data. They will discover descriptions of interesting entities on the Web, load these descriptions, and improve the user experience by being smarter, or enable completely new scenarios, by building on the knowledge found in the Semantic Web [8]. These applications often need to expose the entities and the data they have gathered about these entities from the Web to the end user. In order to do so, labels are often used as human-readable names for the entities. Labels can be utilized for a number of different purposes:

- displaying the data to end-users, instead of displaying the URIs,
- for searches over the Web of Data, be they keyword-based or question-based,
- for indexing purposes, or
- for training and using annotation tools with a given knowledge base, etc.

In order to be able to utilize labels, they need to be made accessible to the application. In the general case it is assumed that labels will be made available, among other information, by dereferencing the URI of an entity using the HTTP protocol, following Linked Open Data principles [21].

In reality, the situation is slightly more complicated. Issues such as internationalization, multiple labels for an entity, the computational costs associated with dereferencing, or the use of alternative labeling properties make the task of finding a label for a given entity much harder than expected. In this paper we investigate how labeling on the Web of Data is actually used. The findings of our analysis allow us to derive a number of recommendations for data publishers. We define a number of metrics that provide a baseline for a quantitative analysis of the state of labeling on the Web. We finally come up with some suggestions on how to improve the current situation. The suggestions are aimed at simplifying the usage of data from the Semantic Web in any application.

The paper is structured as follows. Section 2 describes related work, especially how current applications (mostly browsers for linked data) deal with the issue of labeling. Section 3 draws the distinction between information resources and non-information resources, and how they are currently dealt with by data publishers with regards to labels. In Section 4 we investigate which properties are actually used to provide labels. Even though there is a property defined in the RDFS standard, a number of vocabularies define alternative properties to provide labels. Based on those properties, we define metrics in Section 5 in order to assess the current state of labeling in the Web of Data, followed by the results of applying those metrics on a sample of the Web in Section 6. We close with a number of recommendations and conclusions in Section 7.

2 Related work

Applications enabling human users to exploit the Web of Data can be classified into three categories: Linked Data browsers, Linked Data search engines, and domain specific Linked Data applications [20].

Linked data browsers, such as Disco [9], Tabulator [5] or Marbles [4] to name just a few, enable human users the exploration of linked data similar to how HTML browsers enable exploration of the traditional Web of documents. Instead of navigating between HTML pages, they allow navigation between RDF documents following links in the data by following RDF links. Since applications consuming linked data such as linked data browsers are intended to be used by a broad audience if the Web of Data becomes widely used, hiding technical details such as URIs when displaying facts to human users becomes crucial. For annotating entities with human-readable descriptions, the property `rdfs:label` from the RDF vocabulary is commonly used to provide a human-readable version of a resource's name besides its URI [10].

For example when displaying data available in the linked data cloud for the artist Sidney Bechet using the linked data browser *Sig.ma*, the list of information items for his affiliation contains, amongst other items, the following three items:

- `http://rdf.freebase.com/ns/m.049jnng`
- `http://rdf.freebase.com/ns/m.043j22x`
- Sidney Bechet and His Orchestra

For the first two items no human-readable labels are available to Sig.ma, therefore the URI is displayed which does not represent anything meaningful to the user besides the information that Freebase contains information about Sidney Bechet.

If for a resource no label is known or an unexpected property is used for labeling or the label is not retrieved by resolving the URI, developers of linked data browser came up with a set of options when dealing with the problem of missing human-readable labels:

1. The URI itself is displayed to the user. The URI can be meaningful for some users that do not regard it as noise and that are capable of deriving the meaning from some readable strings in the URI. However, this requires URIs that have been created by following a convention to use meaningful names for URIs.[1] Displaying the URI also often leads to an overly technical feel of the interface.
2. The last part of the URI is used, i.e. the local name or the fragment identifier. For example for the URI `http://www.example.com/about#bob` the fragment identifier `bob` is used, and for the URI `http://www.example.com/people/alice` the last part of the path is used, i.e. `alice`.
3. A more complex mechanism, as e.g. used in Protégé [18] which allows the user to specify which property values to display.

Human-oriented search engines such as Falcons, Sindice, MicroSearch, Watson, SWSE, and Swoogle provide keyword-based search services. Keyword search on graphs relies on the existence of nodes that are labeled thus allowing to match keywords to nodes via their labels [19, 30], or on meaningful URIs .

While measurements of the Web of Data have been performed before [14, 32, 13], an analysis of labels in the Web of Data has not been performed. However, Azlinayati et al. [24] analyzed identifiers and labels in 219 OWL ontologies. Given that the Web of Data mainly consists of instance data, their analysis regarding schema data can be seen as complementing our approach which analyses instance data.

3 Information resources and non-information resources

URIs are used to identify resources, where a resource might be anything from a person over an abstract idea to a simple document on the Web [23]. *Information resources* (IR) are resources that consist of information and therefore all of their essential characteristics can be conveyed in a message and be transported over

[1] However, `http://www.w3.org/Provider/Style/URI` recommends not to use topic names in a URI since thereby an URI's creator binds herself to some classification that can be subject to change, and would therefore require a renaming of the URI, which is considered undesired.

protocols such as HTTP. IRs can be copied from and downloaded via the Internet given their URLs. Disjoint from this set of resources is the set of *non-information resources* (NIR) – resources that cannot be accessed and downloaded via the Internet – such as a person or a country. Nevertheless, a non-information resource can be identified with a URI. Resolving the URI should result in metadata that describes the non-information resource. This idea is part of the Linked Open Data principles [21].

The distinction between information and non-information resources is relevant for the further investigation of labeling behaviour on the Web of Data: whereas NIRs are not directly accessible to the machine (i.e. the machine can talk *about* a resource, but not access or transform it), IRs can be downloaded, displayed, and further processed. IRs do not necessarily require labels in order to be useful to the end-user, whereas for NIRs there is not much else that can be used to represent them in the user interface. IRs can be represented by themselves (in case of a picture), or by a hyperlink to the document, or by the document title (in case of an HTML page or Office document). Applications such as Linked Data browsers should thus be aware of the difference, and treat NIRs and IRs differently. Indeed, some browsers do so. Tabulator [6], Explorator [2], and Graphite[2] display, for instance, images inline with the other data in the browser.

Whether a URI refers to an information resource or to a non-information resource should be determined as follows: Non-information resources should have a hash URI or, if they have a slash URI, resolving the URI should lead to an HTTP `303 See also` response. Hash URIs include a fragment, with a special part that is separated from the rest of the URI by a hash symbol `#` [27].[3] URIs of information resources on the other hand should ultimately resolve with the given information resource, which means with an HTTP response code `200 OK` (after following redirects). When we receive an error when resolving a URI (i.e. a response in the `4xx` or `5xx` range), we cannot infer whether this URI refers or has referred to an information resource or a non-information resource.

Even though URIs are supposed to be opaque [7], an analysis performed on URIs with extensions from the BTC 2010 corpus revealed that URIs with file name extensions such as `.html` or `.jpg` often refer to information resources. In order to test this hypothesis, we collected all URIs ending in extensions from the BTC 2010 corpus. The Billion Triple Challenge (BTC) 2010 corpus[4] is a dataset consisting of linked data crawled from the web which is stored as *ntriples*. Here, each of the 3,167,799,445 ntriples is a quad constituted by a subject, a predicate, an object, and a context, where the context is the URI of the resource the triple has been crawled from. When ignoring the context, thus reducing the quads to triples, the dataset contains 1,441,499,718[5] distinct triples. Looking

[2] `http://graphite.ecs.soton.ac.uk/`

[3] e.g. `http://www.example.com/about#alice`

[4] Available at `http://km.aifb.kit.edu/projects/btc-2010/`, (accessed May 2011)

[5] `http://gromgull.net/blog/2010/09/redundancy-in-the-btc2010-data-its-only-1-1b-triples/` (accessed 2011-06-29)

through the corpus, we found 75,6 Million distinct URIs that were either in the subject or the object position.[6] Of these, 10,3 Million URIs ended in an extension (13,6%). For each extension, we selected a random sample of 50 URIs, and issued HTTP `HEAD` requests. The aim of the request was not to retrieve the whole resource, but only the HTTP header information. If the response to the request was a `303 See other`, the URI would have been a non-information resource even though the URI ended in a file extension. Extensions that appear more than 100,000 times in the BTC 2010 corpus are `.jpg`, `.html`, `.rdf`, `.bml`, `.do`, `.json`, `.ttl`, `.jsp`, `.xml`, `.php`, `.htm`, `.png`, and `.gif`. The percentage of NIRs among those resources is 0% – indeed not a single URI returned a 303 among these extensions. A complete list of all extensions and results can be found online[7]). The results show that almost all URIs ending with an extension are indeed information resources, as expected. The only surprising number we encountered was among `.svg` files, which were encountered 3,287 times. Of these SVG URIs, 31% gave a `303 See other` response. We further investigated the matter, and found that all those URIs came from DBpedia [3], and can be traced back to DBpedia's extraction mechanism, which transforms infobox links to local SVG files on Wikipedia articles as properties of a given entity.

The BTC 2010 corpus also provides a file that contains all URIs that had a `303 See other` response when they have been resolved, and the URIs they have been redirected to.[8] This list contains about 6 Million URIs. Some of them point to HTML documents, and not only to RDF files, but in general we assume that this list contains a subset of the NIRs that are within the BTC 2010 corpus.

4 Labeling properties

The RDFS standard defines the property `label`, which can be used to connect an entity to a name aimed at human consumption [11]. But `rdfs:label` is only one of the many means that are actually used on the Web to assign a human-readable name to an entity. There are several different reasons for using alternative labeling properties. Some vocabularies prefer to use more specific properties to assign names. For example, the FOAF vocabulary [12] defines `foaf:name` to assign a name to a person, as it sounds much more acceptable to give a person a name than a label. The SWRC ontology [29] provides `swrc:name` as well. SKOS even provides a set of properties for preferred and alternative labels [25], as the simple label property from RDFS is not sufficient for the needs of SKOS. Other vocabularies might provide an alternative labeling property due to legacy reasons. FOAF introduces a `foaf:LabelProperty` class for labeling such properties, but this is not used even within FOAF itself.

[6] We also looked at the URIs in the property positions, but within a sample of ca. 40 Million triples we only found a single URI with an extension, and subsequently ignored this case.

[7] `http://km.aifb.kit.edu/sites/label/btc/`

[8] The file `redirects.nx` in the BTC 2010 corpus.

Table 1. Most often used properties for labeling purposes.

Number of quads	Property URI
184,848,373	http://www.w3.org/2000/01/rdf-schema#label
71,742,600	http://xmlns.com/foaf/0.1/nick
17,005,858	http://purl.org/dc/elements/1.1/title
7,107,149	http://purl.org/rss/1.0/title
6,083,581	http://xmlns.com/foaf/0.1/name
2,914,013	http://purl.org/dc/terms/title
2,808,455	http://www.geonames.org/ontology#name
2,413,957	http://xmlns.com/foaf/0.1/nickname
1,649,940	http://swrc.ontoware.org/ontology#name
1,506,497	http://sw.cyc.com/CycAnnotations_v1#label
1,133,192	http://rdf.opiumfield.com/lastfm/spec#title
1,021,985	http://www.proteinontology.info/po.owl#ResidueName
713,219	http://www.proteinontology.info/po.owl#Atom
713,219	http://www.proteinontology.info/po.owl#Element
713,219	http://www.proteinontology.info/po.owl#AtomName
663,485	http://www.proteinontology.info/po.owl#ChainName
541,038	http://purl.uniprot.org/core/fullName
488,528	http://purl.uniprot.org/core/title
452,537	http://www.aktors.org/ontology/portal#has-title
434,237	http://www.w3.org/2004/02/skos/core#prefLabel
404,950	http://www.aktors.org/ontology/portal#name
391,730	http://xmlns.com/foaf/0.1/givenName
358,077	http://www.w3.org/2000/10/swap/pim/contact#fullName
337,650	http://xmlns.com/foaf/0.1/surName
336,063	http://swrc.ontoware.org/ontology#title
317,076	http://swrc.ontoware.org/ontology#booktitle
290,178	http://www.aktors.org/ontology/portal#has-pretty-name
283,754	http://purl.uniprot.org/core/orfName
253,034	http://purl.uniprot.org/core/name
211,193	http://www.daml.org/2003/02/fips55/fips-55-ont#name
186,984	http://www.geonames.org/ontology#alternateName
157,019	http://purl.uniprot.org/core/locusName
132,317	http://www.w3.org/2004/02/skos/core#altLabel
126,250	http://creativecommons.org/ns#attributionName
126,126	http://www.aktors.org/ontology/portal#family-name
126,086	http://www.aktors.org/ontology/portal#full-name

In order to find out which properties are used for labeling, we examined the BTC 2010 corpus. From the corpus we extracted the property from all quads with a literal with the datatype `xsd:string` or without a given datatype. We counted the number of occurrences for each such property. From the set of 178 properties that occurred at least 100,000 times[9] we manually assessed whether the property is used for the purpose of labeling. To do so we performed a URI lookup on the property itself, checking the label and the description of the property, and then looked at instance data. This resulted in a list of 36 properties shown in Table 1 that are used for the purpose of labeling. Note that the numbers in Table 1 should not be read as the number of labeled entities, since an entity can have multiple labels or an entity can be labeled several times in multiple contexts.

Most of these properties are not connected to `rdfs:label` in a way that would allow for machines to automatically discover the alternative labeling property. From the given list, only FOAF [12], SKOS [25], and Geonames[10] explicitly connect their labeling properties to `rdfs:label` via the `rdfs:subPropertyOf` property. Under both RDFS [11] and OWL 2 semantics [17], this would allow to automatically infer that any label connected with the alternative labeling property is also a valid value for `rdfs:label`.[11] Also, the pattern occurs so frequently that it might be worthwhile to hard-code it into an application, to avoid the overhead implied by the usage of a reasoner. Note that proteinontology contains multiple properties used for labeling. This is due to the fact, that these properties are annotated as functional properties with a given domain. For example the domain of the property `po:Atom` is the class `po:Atoms`. That means that when using such a property, besides labeling an entity this, this entity can be uniquely referred to via that label and it can be inferred that this entity belongs to class `po:Atoms`.

5 Metrics

In this section we define a number of metrics that help study the properties of labeling within a dataset. In the following section we will discuss the results of measuring the Web of Data along these metrics.

5.1 Completeness

All non-information resources should have labels. The labeling completeness metric *LC* tells us if this is indeed the case. It is the ratio of all URIs with at least one value for a labeling property to all URIs in a given knowledge base. The metric is extended with three parameters: the actual properties used to assign the label, the entities to be regarded by the metric, and the dataset.

[9] The whole set is available at `http://km.aifb.kit.edu/sites/label/btc/`

[10] `http://www.geonames.org`

[11] Note that this was not true for the OWL 1 Lite and DL semantics since `rdfs:label` is an `owl:AnnotationProperty` [28], but OWL 2 was extended to enable this pattern.

Labeling properties are indicated by the subscript of the metric. They may be defined strictly as only `rdfs:label` (LC_{rdfs}), or including any formally defined subproperty of `rdfs:label` (LC_{rdfs+}), or as any other set of labeling properties lp (LC_{lp}) (such as the set presented in Section 4, which we call BTC).

The regarded entities are defined by the superscript. Most often, we will only want to consider the non-information resources (LC^{NIR}). For an automatic assessment of this metric we also must devise a method to decide whether a given URI is an information resource, or a non-information resource, as discussed in Section 3. One might also argue that some non-information resources actually do not require labels, as some resources are basically artifacts of the knowledge representation (LC^{-}). In RDFS and OWL this would most prominently include nodes that model n-ary relations [26].

The third parameter is given as the argument of the metric. Thus $LC(D)$ is the labeling completeness of the dataset D. We expect $LC(D)$ to always be 1 for a good knowledge base D.

Note that a dataset may include data from several RDF files, and indeed most of the time LC is defined over the merged data from a whole site. In this paper, for example, we regard the BTC as a whole, the merged data from several million look-ups.

5.2 Efficient accessibility

A wide-spread method to work with data from the Semantic Web is called *follow your nose*, and it works due to the Linked Open Data principles [21]: whenever an application encounters an unknown URI, it can simply dereference the URI in order to access information about the entity identified by the URI. This will usually include a label for the entity of interest, and also links to other entities to which the given entity is connected, so that the application can further dereference these as well.

Assume that for the URI `ex:Berlin` the result of this exercise looks as follows:

```
ex:Berlin ex:location ex:Germany .
ex:Berlin rdfs:label "Berlin" .
```

A linked data browser can display the string *Berlin* to represent the resource of interest, but it has to look up both `ex:location` and `ex:Germany` before it can represent the single fact that is included in the response. If an RDF graph contains 50 triples, with about 60-80 different URIs, the application actually needs to make several dozens of HTTP requests in order to display the facts within that single resource. This turns out to be the main reason for the slow performance of linked data browsers [31]: a single browsing step can fire dozens, if not hundreds, of requests.

Imagine that the response would instead be:

```
ex:Berlin ex:location ex:Germany .
```

```
ex:Berlin rdfs:label "Berlin" .
ex:location rdfs:label "Location" .
ex:Germany rdfs:label "Germany" .
```

Now the application can display the fact without any additional lookup. This approach has nevertheless several disadvantages: it implies redundancy, and leads to larger data files. In general it is expected to nevertheless *reduce* the load and bandwidth of serving linked open data as the amount of requests would be significantly reduced.

We define the metric LE as the ratio of all mentioned URIs with at least one value for a labeling property to all mentioned URIs in a given RDF graph. The subscript and superscript are defined as for LC, the superscript can further define a background set of known labels (e.g. for a widely deployed vocabulary like FOAF or GoodRelations [22]). For example, the following graph would have a LE_{rdfs}^{foaf-} of 1, but a LE_{rdfs}^{-} of 0.5 (since the `foaf:img` property has no label). Note that for brevity RDF and RDFS are always assumed to be known.

```
ex:Basil foaf:img ex:basil.jpg .
ex:Basil rdfs:label "Basil" .
```

Whereas for the LC metric we can always look up a given URI, this is not allowed for LE. Nevertheless, LE with sensible parameters should always be 1 in order to increase the utility of any given response for inquiring applications.

5.3 Unambiguity

Each entity can have a whole set of different labels attached to it. This will likely yield meaningful results if the application can distinguish between these labels: SKOS includes different properties for denominating preferred and alternative labels [25], and given a multi-lingual knowledge base we expect to have several labels for a given entity, one in each language (see the following section).

But an entity can also have several labels that are not at all differentiated. In this case an application has to select one of the labels. And unless it does not have a deterministic selection procedure, the application might end up being inconsistent, displaying different labels every time the entity is displayed – which might easily lead to confusion for the user of the application. Even if the application provides a deterministic selection procedure, as long as this procedure is not common among all applications the user uses to interact with a given knowledge base, the user will be exposed to confusing inconsistencies in the interface.

We introduce the metric LU_f which is the ratio of all URIs that have exactly one preferred label according to a selection procedure f to all URIs with any label in a given knowledge base. The superscript is the same as for LC, but the subscript is replaced by the selection procedure f, which might be, in the simplest case, just selecting any value of `rdfs:label` (LC_{rdfs}), but could also include a more sophisticated preference function (e.g. if there is a `skos:prefLabel` take that, otherwise any `rdfs:label`).

As with all the other metrics in this paper, a good knowledge base should have a LU of 1.

5.4 Multilinguality

Language tags can be used on plain literals to state the natural language used by the literal. This enables applications to select the most appropriate literals based on their user's language preferences. An example for a literal with a language tag is `"university"@en` or `"Universität"@de`.

In order to measure multilinguality we define two metrics: LLN, the number of languages used with a labeling property, and LLC^{lang}, the completeness for a given language, i.e. the ratio of URIs with at least one label tagged with the given language or a less specific one to all URIs in a given knowledge base. The same sub- and superscripts apply as for LC (note that there are two different superscripts). If no superscript defines the language, then the average over all used languages is supposed.

6 Results

We used the metrics defined in the previous section on the BTC 2010 corpus. For measuring, we did not consider entailments as defined by the formal semantics of RDFS, OWL, or RIF. In particular we did not mush entities together through `owl:sameAs` statements or inverse functional properties, but regarded them URI by URI.

The BTC2011 corpus consists of 219 chunks. From each chunk we extracted the URIs from the first 100 nquads which resulted in 7195 URIs. For each URI we performed a lookup and identified 1376 NIRs by `303 See other` redirect. By following the redirect and analyzing the RDF data we found that for 526 NIRs at least one label exists given the properties in Table 1. This means that only 38.2% of the analyzed NIRs have a label. Table 2 shows which properties are used to assign labels.

Table 2. Completeness of NIR labels.

Number of NIRs	Labeling property
451	`http://www.w3.org/2000/01/rdf-schema#label`
73	`http://xmlns.com/foaf/0.1/name`
53	`http://purl.org/dc/elements/1.1/title`
20	`http://xmlns.com/foaf/0.1/givenName`
13	`http://purl.org/dc/terms/title`
5	`http://xmlns.com/foaf/0.1/nick`
4	`http://www.w3.org/2004/02/skos/core#prefLabel`

In order to measure the efficient accessibility, we looked through a sample of five random graphs from each second level domain in the BTC 2010 corpus. The results are given in Figure 1. In order to define a set of known vocabularies, we took the ten most widely used vocabularies in the BTC 2010 corpus (see Table 3).

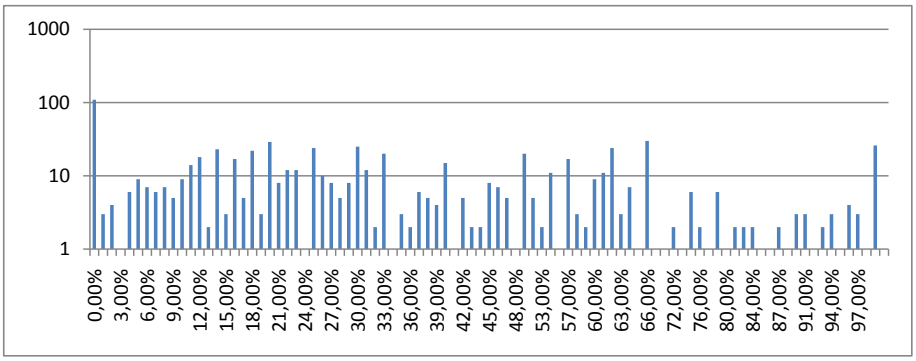

Fig. 1. Histogramm of the LE^{top}_{BTC} of up to five random graphs from each of the domains in the BTC 2010 corpus, for a total of 741 graphs.

Table 3. Top ten most occurring vocabulary namespaces in the BTC 2010 corpus (according to http://gromgull.net/2010/10/btc/explore.html).

Vocabulary namespace	**Number of occurences**
`http://www.w3.org/2000/01/rdf-schema#`	845,952,387
`http://data-gov.tw.rpi.edu/vocab/p/90/`	651,432,324
`http://www.w3.org/1999/02/22-rdf-syntax-ns#`	567,247,265
`http://purl.org/goodrelations/v1#`	527,323,224
`http://xmlns.com/foaf/0.1/`	209,249,423
`http://purl.uniprot.org/core/`	41,961,030
`http://purl.org/dc/elements/1.1/`	29,596,285
`http://www.proteinontology.info/po.owl#`	13,661,605
`http://purl.org/dc/terms/`	12,579,646
`http://www.w3.org/2002/07/owl#`	12,362,503

We measured the unambiguity of the corpus. From the set of 57,532 NIRs that have at least one label in the corpus, 903 NIRs have multiple labels – either multiple labels for at least one of the labeling properties shown in Table 1, or multiple labels for at least one property and language. This results in an unambiguity ratio of 0.98.

Finally, we measured the multilinguality of the Web of Data. In general, most data sources contained at most one language (2.2%), if any was specified. A merry few (0.7%) contained several language tags, but even they did not have a high completeness. The most commonly used language tags are `en` (44.72%), `de` (5.22%), and `fr` (5.11%).

Labels are used in order to provide a human-readable names for entities. Every entity should have labels in all relevant languages. Almost none of the datasets on the Web have a full set of labels in more than one language, i.e. most ontologies are not multi-lingual. Thus they miss a potential benefit of the Semantic Web, i.e. the language-independence of the Web of Data.

7 Conclusion

Our work has investigated the current state of labeling the Web of Data, and some problems that need to be considered in the future in order to optimize the ways application developers and potential end-users interact with the data. We have defined metrics to assess the completeness, efficient accessibility, unambiguity, and multilinguality. These metrics address issues that were problematic during the development of applications. The list is not complete, but sound given that they are all based in previous experience. While defining the metrics, we noticed that we had to include a number of parameters that depend on the application that will use the knowledge. This is not surprising: data on the Web of Data is hardly ever evaluable by itself – it greatly benefits from knowing the context of an application that will use the data. The parameters in the evaluation metrics allow to customize the metrics based on the given application, on the labeling properties the application understands, and on the set of entities that are expected to play a role when using the application.

Based on our findings and the argumentation leading to the definition our metrics, we can nevertheless make a number of suggestions on how to improve the quality and usefulness of labels in the data:

- Provide labels for all URIs *mentioned* in a given RDF graph, not only for the *main entities*, as this will considerably speed displaying the data with human-readable names and reduce the number of requests significantly.
- Provide a complete set of labels in all supported languages. One of the biggest advantages of the Web of Data is its inherent multilinguality, but currently this is a tremendously underused feature of the architecture.
- If you are using a labeling property of your own, connect your labeling property to `rdfs:label` explicitly with the `rdfs:subPropertyOf` property. Use `rdfs:label` redundantly as well, since many tools will not provide the inferencing needed to understand your labeling property. If possible, simply avoid using your own labeling property.
- Do not provide more than one obvious preferred label for each entity, in order to decrease the possible confusion for the end-user when using an application over your data.

The suggestions given above lead to an obvious problem: even a moderately small RDF graph with about 100 triples will include about 150 entities. Labeling all these entities in, e.g. ten languages will lead to an extra 1500 triples – a huge overhead (and not even considering the costs creating those labels, a task that would highly benefit from automation). While one could devise new protocols to deal with these problems, there is also an under-utilized existing solution: HTTP allows to set the `Accept-Language` header, that defines a set of natural languages the response should cover [16]. By using the HTTP headers a data provider could both provide all labels necessary for an efficient exposure of the data, as well as not unnecessarily inflate the size of the response by only providing the requested languages.

Labels should follow a style guide and be used consistently. A style guide should define if classes are labeled with plural or singular noun, if properties are labeled with nouns or verbs, etc. Labels should never use camel case or similar escape mechanisms for multi word terms, but instead simply use space characters (or whatever is most suitable for the given language). I.e. an URI `http://example.org/LargeCity` should have a label `"large city"@en`. External dictionaries such as WordNet [15] can be used to check consistency with regards to a style guide.

In an environment where datasets are assembled on the fly from multiple datasets [1], the assembled parts may follow different style guides. The assembled dataset will then not adhere to a single style guide and thus offer an inconsistent user interface. It is not expected that a single style guide will become ubiquitous on the whole Web. Instead, a dataset may specify explicitly what style guide it follows, and even provide labels following different style guides. This would allow to introduce a subproperty of label that is style guide specific, which would in return allow for the consistent display of assembled datasets.

Even when subproperties of `rdfs:label` are defined, there should always be one label (per supported language) given explicitly by using `rdfs:label` itself. Even though this is semantically redundant, many tools (especially visualization tools) do not apply reasoning for fetching the labels of an entity but simply look for the explicit triple stating the entity's label.

Many of the problems described in this paper are a consequence of publishing data using the Linked Open Data principles. It is not clear if this is indeed the best way to publish data on the Web of Data. Serving data through a SPARQL endpoint provides a viable alternative, with the big advantage that the application can, in a very fine-grained way, describe exactly what kind of information, labels, and language it needs. The SPARQL endpoint can then try to understand the query and do its best effort to provide a viable response.

The Linked Open Data principles have spread widely due to their obvious advantages derived from being part of the Web architecture. But the principles are meeting their limitations, as this investigation on labels shows. The Semantic Web has long struggled with the chicken and egg problem of data vs. applications. Now that the data is there, we see that applications don't yet follow with the same force that the datasets had. One of the reasons is the lack of quality in some of the published datasets.

Labeling may be just a small, but at the same time it is an absolutely essential piece of the puzzle that is needed for the Web of Data to finally become widely used.

Acknowledgements

We thank Andreas Harth for his support on how to crawl the data. The work presented in this paper is supported by the European Union's 7th Framework Programme (FP7/2007-2013) under Grant Agreement 257790.

References

1. H. Alani. Position paper: ontology construction from online ontologies. In L. Carr, D. D. Roure, A. Iyengar, C. A. Goble, and M. Dahlin, editors, *Proceedings of the 15th international conference on World Wide Web (WWW2006)*, pages 491–495, Edinburgh, Scotland, May 2006. ACM.
2. S. F. C. Arajo, D. Schwabe, and S. D. J. Barbosa. Experimenting with explorator: a direct manipulation generic rdf browser and querying tool.
3. S. Auer and J. Lehmann. What have Innsbruck and Leipzig in common? Extracting semantics from wiki content. In E. Franconi, M. Kifer, and W. May, editors, *Proc. 4th European Semantic Web Conference (ESWC)*, 2007.
4. C. Becker and C. Bizer. Marbles, 2009.
5. T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. Tabulator: Exploring and Analyzing linked data on the Semantic Web. In *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, volume 2006, 2006.
6. T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. Tabulator: Exploring and analyzing linked data on the Semantic Web. In L. Rutledge, m.c. schraefel, A. Bernstein, and D. Degler, editors, *Proceedings of the Third International Semantic Web User Interaction Workshop SWUI2006 at the International Semantic Web Conference ISWC2006*, 2006.
7. T. Berners-Lee, R. Fielding, and L. Masinter. Uniform Resource Identifier (URI): Generic Syntax. Technical Report 3986, Internet Engineering Task Force, June 2005. RFC 3986 (available at `http://www.ietf.org/rfc/rfc3986.txt`).
8. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 2001(5), 2001. available at http://www.sciam.com/2001/0501issue/0501berners-lee.html.
9. C. Bizer and T. Gau. Disco - hyperdata browser, January 2007.
10. D. Brickley and R. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, 2004.
11. D. Brickley and R. V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, February 2004.
12. D. Brickley and L. Miller. The Friend Of A Friend (FOAF) vocabulary specification, July 2005.
13. M. d'Aquin, C. Baldassarre, L. Gridinoc, S. Angeletou, M. Sabou, and E. Motta. Characterizing knowledge on the semantic web with watson. In R. Garcia-Castro, D. Vrandecic, A. Gmez-Prez, Y. Sure, and Z. Huang, editors, *EON*, volume 329 of *CEUR Workshop Proceedings*, pages 1–10. CEUR-WS.org, 2007.
14. L. Ding and T. Finin. Characterizing the semantic web on the web. In *Proceedings of the 5th International Semantic Web Conference*, 2006.
15. C. Fellbaum. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. MIT Press, May 1998.
16. R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, June 1999.
17. B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309–322, 2008.
18. W. E. Grosso, H. Eriksson, R. W. Fergerson, S. W. Tu, and M. A. Musen. Knowledge modeling at the millennium: the design and evolution of PROTEGE-2000. In *Proceedings of the 12th International Workshop on Knowledge Acquisition, Modeling and Mangement (KAW-99)*, Banff, Canada, October 1999.

19. H. He, H. Wang, J. Y. 0001, and P. S. Yu. Blinks: ranked keyword searches on graphs. In C. Y. Chan, B. C. Ooi, and A. Zhou, editors, *SIGMOD Conference*, pages 305–316. ACM, 2007.
20. T. Heath. How Will We Interact with the Web of Data? *IEEE Internet Computing*, 12:88–91, September 2008.
21. T. Heath and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool, 2011.
22. M. Hepp. Goodrelations: An ontology for describing products and services offers on the web. In A. Gangemi and J. Euzenat, editors, *EKAW*, volume 5268 of *Lecture Notes in Computer Science*, pages 329–346. Springer, 2008.
23. I. Jacobs and N. Walsh. Architecture of the World Wide Web Vol. 1, 2004. W3C Recommendation 15 December 2004, avail. at `http://www.w3.org/TR/webarch/`.
24. N. A. A. Manaf, S. Bechhofer, and R. Stevens. A Survey of Identifiers and Labels in OWL Ontologies. In *Proceedings of the 6th International Workshop on OWL Experiences and Directions (OWLED 2010)*, 2010.
25. A. Miles and S. Bechhofer. SKOS Simple Knowledge Organization System Reference, 2009. W3C Recommendation 18 August 2009, available at `http://www.w3.org/TR/skos-reference/`.
26. N. Noy and A. Rector. Defining n-ary relations on the semantic web. W3C Working Group Note, April 2006. available at `http://www.w3.org/TR/swbp-n-aryRelations/`.
27. L. Sauermann and R. Cyganiak. Cool URIs for the Semantic Web. W3C Interest Group Note, December 2008.
28. M. K. Smith, C. Welty, and D. McGuinness. OWL Web Ontology Language Guide, February 2004. W3C Recommendation 10 February 2004, available at `http://www.w3.org/TR/owl-guide/`.
29. Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann, and D. Oberle. The swrc ontology - semantic web for research communities. In G. D. Carlos Bento, Amilcar Cardoso, editor, *Proceedings of the 12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence (EPIA 2005)*, volume 3803 of *LNCS*, pages 218 – 231, Covilha, Portugal, Dezember 2005. Springer.
30. D. T. Tran, H. Wang, S. Rudolph, and P. Cimiano. Top-k exploration of query candidates for efficient keyword search on graph-shaped (rdf) data. In *Proceedings of the 25th International Conference on Data Engineering (ICDE'09)*, Shanghai, China, März 2009.
31. D. Vrandečić, V. Ratnakar, M. Krötzsch, and Y. Gil. Shortipedia: Aggregating and curating semantic web data. In *Proceedings of the ISWC 2010*, Shanghai, China, 11 2010.
32. T. D. Wang, B. Parsia, and J. Hendler. A survey of the web ontology landscape. In *Proc. of the ISWC 2006*, 2006.