Capturing the Currency of DBpedia Descriptions and Get Insight into their Validity

Anisa Rula[1], Luca Panziera[2], Matteo Palmonari[1], and Andrea Maurino[1]

[1] University of Milano-Bicocca
{rula|palmonari|maurino}@disco.unimib.it
[2] KMI, The Open University
{luca.panziera}@open.ac.uk

Abstract. An increasing amount of data is published and consumed on the Web according to the Linked Open Data (LOD) paradigm. In such scenario, capturing the age of data can provide insight about their validity under the hypothesis that more up-to-date data is, more likely is to be true. In this paper we present a model and a framework for assessing the currency of the data represented in one of the most important LOD datasets, DBpedia. Existing currency metrics are based on the notion of date of last modification, but often such information is not explicitly provided by data producers. The proposed framework extrapolates such temporal metadata from time-labeled revisions of Wikipedia pages (from which data has been extracted). Experimental results demonstrate the usefulness of the framework and the effectiveness of the currency evaluation model to provide a reliable indicator of the validity of facts represented in DBpedia.

Keywords: Data Quality, DBpedia, Information Extraction, Currency, Validity, Linked Data

1 Introduction

Linked Open Data (LOD) can be seen as a Web-scale knowledge base consisting of named resources described and interlinked by means of RDF statements [3]. The extraction of structured information from semi-structured content available in Wikipedia enabled several knowledge bases to be published within the DBpedia project[3]. Information in LOD datasets changes over time to reflect changes in the real world; new statements are added and old statements are deleted [16, 5], with the consequence that entity documents consumed by users can soon become outdated. Out-of-date documents can reflect inaccurate information. Moreover, more up-to-date information should be preferred over less up-to-date information in data integration and fusion applications [10, 12].

The problems of dealing with change, providing up-to-date information, and making users aware of how much up-to-date information is have been

[3] http://dbpedia.org/

acknowledged in the Semantic Web. A resource versioning mechanism for linked data has been proposed [6], which allows data providers to publish time-series of descriptions changing over time; however this mechanism has been adopted only by a limited number of datasets and not from DBpedia. DBpedia Live [9] provides a model to continuously update RDF descriptions and to publish data that are as up-to-date as they are in Wikipedia, the original source. This approach provides a step forward in the delivery of up-to-date and reliable information. However, even this approach is based on batch update strategies and, although data is nearly synchronized with Wikipedia, there is still some delay in the propagation of changes that occur in Wikipedia. As an example, the publication of the new album titled "Get Up!" by the singer Ben Harper on January 29th 2013, is represented in Wikipedia, but not in DBpedia Live as of February, 15th, 2013. Moreover, DBpedia Live is not as complete as English DBpedia, the DBpedia Live approach has not been applied to localized datasets, and several applications still need to use local DBpedia dumps to implement scalable systems.

Data *currency* (currency, for short) is a quality dimension proposed in the data quality domain to describe and measure the age of (relational) data, and the speed at which a system is capable to record changes occurring in the real-world [2]. The computation of these currency measures needs *versioning metadata* [16, 13], which represent the date when RDF statements or documents are created or modified. Unfortunately, the use of versioning metadata in LOD datsets, and in particular in DBpedia, is not a frequent practice as shown in [13]. To overcome the mentioned problems, in this paper, we address the issue of supporting the consumer to evaluate the freshness of explored data by defining:

- a model to measure currency in LOD that encompasses two currency measures, namely *basic currency*, based on the age of data, and *system currency*, based on the delay with which data is extracted from a web source;
- a method for estimating the last modification date of entity documents in DBpedia starting from the corresponding Wikipedia pages;
- an evaluation method for estimating the quality of proposed currency measures based on the validity of facts.

The paper is organized as follows: Section 2 discusses related work on the assessment of time-related data quality dimensions; Section 3 introduce the definitions adopted in the paper, and the metrics for measuring and evaluating currency. Our framework for assessing currency of DBpedia documents is presented in Section 4. Section 5 shows results of our experimentation on DBpedia and, in Section 6, we draw conclusions and future work.

2 Related Work

Data currency is a quality dimension well known in the literature of information quality [2]. Assessing currency in traditional Database Management Systems (DBMSs) is straightforward due to the fact that DBMSs track data updates into log files. In the LOD domain, the definition of currency needs to be adapted

to a new data model, where the resource identifiers are independent from the statements where they occur. Further, the statements can be added or removed and these actions are not represented in the data model.

SIEVE is a framework for evaluating quality of LOD, proposed in the context of, and applied to, a data fusion scenario [10]. The framework results show that data currency is a crucial driver when data coming from heterogeneous sources have to be integrated and fused. The authors do not investigate the specific problem of measuring data currency on arbitrary datasets. Our contribution can be seen as a point of support to the approaches such as SIEVE since the currency evaluation needs temporal meta-information to be available.

A recent technique able to assess the currency of facts and documents is presented in [14]. Given a SPARQL query q over an LOD dataset, it estimates the currency of the returned result set. Furthermore, to overcome the problem of the semantic heterogeneities related to the use of different vocabularies in the LOD datasets, the authors propose an ontology, which integrates in one place all the temporal meta-information properties. According to this approach, currency is measured as a change between the time of the last modification and the current time, whereas we provide a further measure that assess the time between a change in the real world and a change in the knowledge base. Both approaches, SIEVE and the one in [14] rely on timestamps such as *lastUpdate* or *lastModificationDate* which are often unreliable or may not be provided by many documents. Our proposed method extends the approaches seen before. It deals with the incomplete and scattered temporal metadata available in LOD today. In particular, we estimate the currency of documents for which such metadata are not available.

Other time-related metadata define the temporal validity of RDF statements according to the Temporal RDF model [7]; the model has been extended to improve the efficiency of temporal query answering [15] and to link temporal instants and intervals so that they can be associated to several entities [5]. However, only few datasets like Timely YAGO [17] associate RDF statements with their temporal validity defined by a temporal interval. For this reason they leverage explicit temporal metadata available from the Wikipedia's infoboxes, which had been not considered in DBpedia. Only a subset of DBpedia statements can be associated with temporal validity following this approach, though. The goal of our approach is different because we aim at extracting the last modification date of a full document. In order to achieve this goal, and differently from Timely YAGO, we look into version history of Wikipedia pages. Similar to our approach, the work in [1] also detects and extracts from the revision history of Wikipedia, the last modification date of each attribute in the infobox. In contrast to our approach, it gathers the revision history of Wikipedia based on data dumps. To ensure that it gathers updated information, the system in [1] aggregates evidences from a large number of crawled Wikipedia documents. This approach is complementary to our current approach and can easily be combined with it. The work in [**?**] is a recently proposed method to parse the Wikipedia revision history dumps and for each DBpedia fact it returns the last modification date.

The approach is based on combining a standard Java SAX parser with the DBpedia Extraction Framework. Yet, it does not use the most recent version since it accesses the dumps of Wikipedia documents rather than using the newer revisions on the Web.

3 Currency Model

3.1 Preliminaries

Linked Open Data describes resources identified by HTTP URIs by representing their properties and links to other resources using the RDF language. Given an infinite set $\mathcal{U}$ of URIs, an infinite set $\mathcal{B}$ of blank nodes, and an infinite set $\mathcal{L}$ of literals, a statement $< s, p, o > \in (\mathcal{U} \cup \mathcal{B}) \times \mathcal{U} \times (\mathcal{U} \cup \mathcal{B} \cup \mathcal{L})$ is called an *RDF triple*. As the use of blank nodes is discouraged for LOD [3], we will assume that the subject and the property are URIs, while the object can be either a URI or a literal.

Most resources described in the LOD Cloud represent real-world objects (e.g., soccer players, places or teams); we use the term *entities* as a short form for *named individuals* as defined in the OWL 2 specification. According to the LOD principles [3], we assume that each entity e can be dereferenced. The result of the dereferencing is an RDF document denoted d^e which represents a *description* of the entity e. We say that d^e *describes* e and d^e is an *entity document* [8]. As an example, an entity document d^e returned by DBpedia in NTriples format contains all the RDF triples where e occurs as a subject. In this work, RDF triples occurring in entity documents are called *facts*. A *fact* represents a relation between the subject and the object of the triple and, intuitively, it is considered true when the relation is acknowledged to hold. In this paper, as we are interested in analyzing different versions of entity documents that are published at different time points, we regard time as a discrete, linearly ordered domain, as proposed in [7].

Let $E = \{e_1, e_2, \ldots, e_n\}$ be a set of entities in a dataset G and $D = \{d^{e_1}, d^{e_2}, \ldots, d^{e_n}\}$ be a set of entity documents. Let S be a non-structured or semi-structured web source in the web that consists of a set of web pages $S = \{p_1, p_2, \ldots, p_n\}$. We assume that each entity e_i represented by an entity document d^{e_i} has a corresponding web page $p_i \in S$; in the following we use the notation p^e to refer to the web page that describes an entity e. Entities descriptions and source web pages change over time; different *versions* of a web page corresponding to an entity exist. Formally, a version of a web page p can be represented as a triple $v =< id, p, t >$, where id is the version identifier, p is the target page, and t is a time point that represents the time when the version has been published.

3.2 Currency Measures

Starting from the definition of currency proposed for relational databases [2], we define two quality dimensions namely age-based currency and System currency.

Definition 1 (Age-Based Currency). *The data currency of a value is defined as the age of the value, where the age of a value is computed as the difference between the current time (the observation time) and the time when the value was last modified.*

The above definition can be easily extended to RDF documents. Let $cTime$ and $lmTime(d^e)$ be respectively the *current time* (the assessment time) and the last modification time of an entity document d^e. We first define the *age measure of an entity document*, based on the above informal definition of data currency. The $age(d^e) : D \rightarrow [0 \cdots + \infty]$ of an entity document d^e, with $e \in E$, can be measured by considering the time of the last modification of the entity document according to the following formula:

$$age(d^e) = cTime - lmTime(d^e) \tag{1}$$

We introduce a measure called *age-based currency*, which depends on the document *age* and returns normalized values in the range $[0, 1]$ that are higher when the document are more up-to-date and lower when documents are old. The age-based currency $\beta(d^e) : D \rightarrow [0, 1]$ of an entity document d^e is defined as follows:

$$\beta(d^e) = 1 - \frac{age(d^e)}{cTime - startTime} \tag{2}$$

where $startTime$ represents a time point from which currency is measured (as an example, $startTime$ can identify the time when first data have been published in the LOD). Observe that this age-based currency measure does not depend on the particular nature of RDF document. In fact, the same measure can be adopted to evaluate currency of other web documents such as Wikipedia pages (and will be used to this aim in Section 5).

Age-based currency is based on age, as defined in Equation 1, and strongly depends on the assessment time (current time). We introduce another measure, which is less sensitive to current time (current time is used only for normalizing the values returned by the measure). We take inspiration from, and adapt a definition of currency proposed for relational databases.

Definition 2 (System-Currency). *Currency refers to the speed with which the information system state is updated after the real-world system changes.*

According to this definition currency measures the temporal delay between changes in the real world and the consequent updates in the data. An *ideal* system of currency measure in our domain should evaluate the time elapsed between a change affecting a real-world entity and the update of the RDF document that describes that entity. However, changes in the real-world are difficult to track because real-world is opaque to a formal analysis. Since the data under the scope of our investigation are extracted from a web source, we can define the system currency of an RDF document by looking at the time elapsed between the update of the web source describing an entity and the update of the correspondent RDF document.

We first define the notion of system delay with respect to an entity, defined by a function $systemDelay(d^e) : D \rightarrow [0 \cdots + \infty]$ as the difference between the time of last modification of a web page p^e and the time of last modification of its respective entity document d^e as follows:

$$systemDelay(d^e) = lmTime(p^e) - lmTime(d^e) \tag{3}$$

Based on the measure of system delay, we introduce a new currency measure called *system currency* that returns normalized values in the interval $[0, 1]$, which are higher when the data are more up-to-date and lower values when data are less-up-to-date with respect to the web source. The system currency $\sigma(d^e) : E \rightarrow [0, 1]$ of an entity document d^e is defined as:

$$\sigma(d^e) = 1 - \frac{systemDelay(d^e)}{cTime - startTime} \tag{4}$$

4 DBpedia Currency Assessment

We provide an assessment framework that leverages the temporal entities extracted from Wikipedia to compute the data currency of DBpedia entity documents; the assessment framework follows three basic steps: (1) extract a document representing an entity from the DBpedia dataset, (2) estimate the last modification date of the document looking at the version history of the page that describes the entity in Wikipedia, (3) use the estimated date to compute data currency values for the entity document.

The main problems we have to face, in order to apply the currency model described in Section 3, concerns the (un)availability of the temporal meta-information required to compute the currency, namely the last modification date of an RDF document. According to a recent empirical analysis, only 10% of RDF documents and less than 1% of the RDF statements are estimated to be associated with temporal meta-information [13]; which means that it is not possible to extrapolate information about the last modification of a document from temporal meta-information associated with statements. To assess the data currency of RDF documents, we propose an assessment strategy based on the measures defined in Section 3. Our assessment framework takes a DBpedia entity as input and returns a data currency value. We propose a method, similar to the data extraction framework proposed in [11], which uses versioning metadata available in Wikipedia pages, which are associated with time-stamped global version identifiers, to extract the time-stamps to associate to RDF documents as last modification date.

A pseudocode of the algorithm that implements the strategy is described in Algorithm 1. The input of the algorithm is an entity e for which the currency has to be evaluated using the currency measures proposed in Section 3. To obtain the estimated last modification date of an entity document we need to extract its document description and its correspondent page from the web.

The description of the entity d^e is obtained by the function $rdfFetcher(e)$, line 2, which reads the statements recorded in an N-Triples entity document and

Algorithm 1: DBpediaCurrency

Input: An entity e
Output: Data currency values $\beta(d^e)$ and $\sigma(d^e)$

```
1  result = φ
2  d^e = rdfFetcher(e)
3  v = getLastVersionId(p^e)
4  d̄^e_v = buildRdf(v)
5  while d̄^e_v is not equal to d^e do
6      v = getPreviousVersionId(p, v)
7      d̄^e_v = buildRdf(p, v)
8  τ = getTimestamp(v)
9  β(d^e) = computeBasicCurrency(τ)
10 σ(d^e) = computeSystemCurrency(τ)
11 return β(d^e), σ(d^e)
```

records them. Among all the statements in the document we keep only those that use DBpedia properties; considered documents represent relational facts and do not include typing information, links to other datasets (e.g., same as links and other links to categories); in other words we consider in entity documents statements that are more sensitive to changes.

From line 3 to line 4, the algorithm extracts the ID of the last revision of the web page p^e (i.e the infobox information) corresponding to the entity e and builds a structured representation of p^e. In order to build a structured RDF content we need to identify properties and their values in the semi-structured part of the document (p^e). The algorithm creates RDF statements from p^e by using the same approach and mappings used to extract DBpedia statements from Wikipedia infoboxes [4].

Given p and d^e, the algorithm finds whether the structured representation of p provided by $\overline{d}^e_v$ matches the entity description d^e (see line 4-7); we use an exact match function. In case the last revision of the structured representation of p does not match to the entity document, the algorithm checks for older versions and stops only when a matching version is found. At this point, we associate the timestamp of the v version to the entity document d^e (line 8). In this way we compute the currency of d^e (see line 9-11).

5 Experimental Evaluation

5.1 Experimental Setup

Methodology and Metrics. When multiple metrics are defined the problem of evaluating the quality of such dimensions arises. In this section we propose a method to evaluate the effectiveness of data currency measures addressing entity documents extracted from semi-structured information available in web sources. Out-of-date documents may contain facts that are not valid anymore when data are consumed; intuitively a description can be considered *valid* when it accurately

represents a real-world entity. A data currency measure that is highly correlated to the validity of fact is useful since it may provide insights about the reliability of data. We define two metrics, namely *accuracy* and *completeness*, to capture the intuitive notion of validity by comparison with the web source which data are extracted from. These metrics use the semi-structured content available in the web source - Wikipedia infoboxes in our case - as a Gold Standard against which entity documents are compared.

Let us assume to have a mapping function μ that map every Wikipedia fact to an RDF fact. This mapping function can be the one used to extract DBpedia facts from Wikipedia facts[4]; however other techniques to extract RDF facts from semi-structured content have been proposed [12]. A DBpedia fact s is valid at a time point t iff there exist a Wikipedia fact w in a page version $v = \langle id, p, t' \rangle$ such that $\mu(w) = s$, with $t' \leq t$ and v being the last page version at t.

We define the accuracy $A(d^e)$ of an RDF document d^e as the number of semantically accurate facts $VF(d^e)$ in d^e divided by the total number of facts $TF(d^e)$ in d^e:

$$A(d^e) = \frac{VF(d^e)}{TF(d^e)} \tag{5}$$

We define the completeness of an RDF document d^e as the number of semantically accurate facts $VF(d^e)$ in d^e divided by the total number of existing facts $WF(p^e)$ in the original document p^e:

$$C(d^e) = \frac{VF(d^e)}{WF(p^e)} \tag{6}$$

Dataset. In order to produce an exhaustive and significant experimentation, we define a specific subset of DBpedia entities that is a representative sample for the entire dataset. These entities belong to the *Soccer Player* category. The main facts describing a soccer player in the Wikipedia's infobox template are the facts of a soccer player such as his appearance, his goals or moving from one club to another, which denote evidences about the changes performed. The changes can usually vary from three to seven days, which imply a high frequency of modifications. Furthermore, after observing for several months the modification set of the Wikipedia's infoboxes of the soccer players, we noticed that there is a high interest of the end-users to maintain the infoboxes up-to-date. Often due to the high frequency of changes in Wikipedia, the new modifications are not replicated also to DBpedia.

5.2 Currency and validity correlation

To realize the experiments we define ten samples composed each one by thirty soccer players chosen randomly where each sample contains a set of entities that have a age-based currency evaluated on DBpedia defined in a specific interval (e.g., $[0, 0.1]$, $[0.1, 0.2]$, ... $[0.9, 1]$).

[4] http://mappings.dbpedia.org

Table 1. Pearson's correlation between currency metrics, accuracy and completeness for DBpedia

	$\beta_{Wikipedia}$	$\beta_{DBpedia}$	σ
$A(d^e)$	-0.3208	0.1596	0.4405
$C(d^e)$	-0.3066	0.2815	0.5402
$H(A(d^e), C(d^e))$	-0.3286	0.2355	0.5150

As provided in table 1, the results of the computations show that the Pearson's coefficients are lower than 0.75 which mean that there is no a linear relationship between currency and accuracy or currency and completeness. Even though the correlation between system currency and the other estimators is lower than 0.75, we notice that the correlation between system currency and completeness is higher than the other correlation.

In addition, we calculated the Spearman's rank correlation coefficient, which is a metric that can be used to verify if there exists a non-linear correlation between the quality metrics. Table 2 shows the Spearman's coefficient computation between the two classes of quality measures performed on the collected soccer player entities.

Table 2. Spearman's correlation between currency metrics, accuracy and completeness for DBpedia

	$\beta_{Wikipedia}$	$\beta_{DBpedia}$	σ
$A(d^{e_i})$	-0.4874	-0.0713	0.5620
$C(d^{e_i})$	-0.4642	0.2349	0.8634
$H(A(d^{e_i}), C(d^{e_i}))$	-0.5351	0.0553	0.7568

The evaluation results allow us to assert that there exists a non-linear correlation between system currency and completeness because Spearman's coefficient is higher than 0.75. The correlation between the harmonic mean and the system currency is low (even if it shows a value closed to the threshold), we can deduce that there exists a correlation between the two components, which is driven by completeness. Finally, the complete absence of the correlation among validity indicators and the age-based currency of the Wikipedia and DBpedia ones, confirms that the accuracy and completeness do not depend on the age of the document.

In order to figure out the behaviour of the system currency and completeness, we compute the LOWESS (locally weighted scatterplot smoothing) function. This function is able to compute the local regression for each subset of entities and to identify, for each interval of system currency, the trend of the two distributions.

Figure 1 shows the LOWESS line between system currency and completeness. As provided by the graph, the local regression cannot be represented by a well-known mathematical function (e.g., polynomial, exponential or logarithmic) because it has different behaviours over three system currency intervals. Notice that for system currency values greater than 0.6, the two distributions tend to increase and for system currency values up to 0.4, the metrics tend to decrease. A particular case is shown in the interval with system currency values going from 0.4 up to 0.6 where the LOWESS is constant.

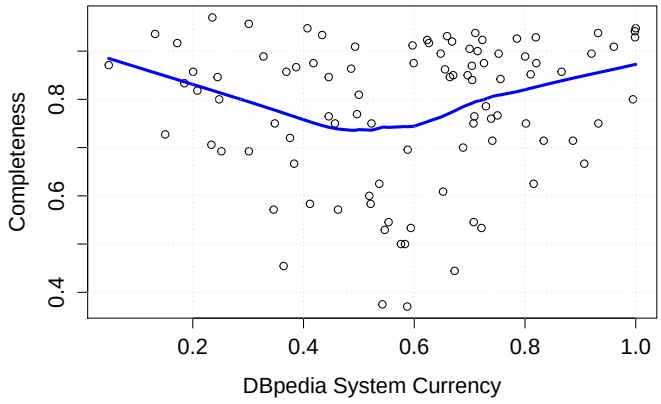

Fig. 1. Local regression between system currency and completeness for entities on DBpedia

Table 3. Pearson's and Spearman's correlation between system currency intervals and completeness

σ	*Pearson's corr.*	*Spearman's corr.*
> 0.6	0.5234	0.8103
$[0.4, 0.6]$	-0.0593	-0.1330
< 0.4	-0.4103	-0.7827

In order to verify the linearity on the three entities subsets, we compute the correlation coefficients also on the identified system currency intervals. The results provided in table 3 show that even for the subsets there does not exist a linear correlation. In particular, there is no correlation in the central interval where the Pearson's coefficient is close to 0 and the distribution is disperse as shown in figure 1. Furthermore, the entities in the central interval show that there does not exist a non-linear correlation.

Instead, the distributions increases according to a non-linear function because Spearman's coefficient is greater than 0.75 for the entities having high system currency and the lower interval distribution decrease according to a non-linear function represented by the Spearman's correlation which is lower than -0.75.

6 Discussion and Conclusions

Based on the experimentation we can provide several general observations about the relation between system currency and completeness. The first one is that entities with high system currency tend to be more complete. Commonly, there is an update to the soccer player infobox each week. In particular, the two facts changing frequently in the infobox are appearances and goals associated with the current club. While these two facts change often, the fact that the soccer

players moves to a new club can change at most two times a year. Thus, we can deduce that in a soccer player domain only two facts can change in a short time. High system currency implies low elapsed time between the last modification time of a document extracted from DBpedia and the correspondent document extracted from Wikipedia. In case the elapsed time is low, the probability that the player changes club is low too which means that only a few infobox modifications occurred. According to the formula 6, the completeness increases if the number of facts changing in the Wikipedia infobox is small.

The second observation concerns to the behaviour of entities with low system currency. A deep analysis of these particular entities shows that the associated Wikipedia pages have two common characteristics: (i) low frequency of modifications (approximately between six and twelve months), and (ii) refer to ex-soccer players or athletes with low popularity. The low frequency of changes in a document, typically known as *volatility* [2], implies a low system currency. Hence, the time elapsed between the last modification of the document extracted by DBpedia and the last modification of the associated document on Wikipedia is high. Furthermore, ex-soccer players infoboxes do not change often because they are considered to have finished their career. As a consequence of rare modifications (few changes) in the infobox attributes, such entity documents expose high completeness.

In case of entities for which the players have low popularity, the scenario is quite different. The infobox modification for these entities implies high completeness. Information about the unpopular players can change weekly, as well as for famous players. Therefore, if a page is not frequently updated could provide incorrect information. Consequently, we can infer that DBpedia entities of unpopular soccer players with low system currency and high completeness refers to Wikipedia pages infrequently updated, therefore, could not represent current information of real world that is also reflected in DBpedia.

At the end of such deep analysis of our experiments, we can assert that: (i) our approach for measuring currency based on the estimated timestamp is effective; (ii) there exist a non-linear correlation between system currency and completeness of entities; (iii) more the system currency of an entity is higher, more the associated DBpedia document has high completeness; and (iv) entities with low system currency, that are instances of DBpedia classes of which their information can change frequently (e.g., unpopular soccer player), are associated with Wikipedia pages that could not provide real world information.

We plan to investigate several research directions in the future; on one hand; we will extend the experiments to analyze the currency of every DBpedia entity and the correlation to the validity of their respective documents; on the other hand, we will study the correlation between volatility and currency to improve the estimation of the validity of facts.

Acknowledgements

The work presented in the paper is supported in part by the EU FP7 project COMSODE - Components Supporting the Open Data Exploitation (under contract number FP7-ICT-611358).

References

[1] E. Alfonseca, G. Garrido, J.-Y. Delort, and A. Peas. Whad: Wikipedia historical attributes data. *Language Resources and Evaluation*, pages 1163–1190, 2013.
[2] C. Batini, C. Cappiello, C. Francalanci, and A. Maurino. Methodologies for data quality assessment and improvement. *ACM Computing Surveys*, 2009.
[3] C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *IJSWIS*, 2009.
[4] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. DBpedia - a crystallization point for the web of data. *Web Semantic*, 2009.
[5] G. Correndo, M. Salvadores, I. Millard, and N. Shadbolt. Linked timelines: Temporal representation and management in linked data. In *COLD*. 2010.
[6] H. V. de Sompel, R. Sanderson, M. L. Nelson, L. Balakireva, H. Shankar, and S. Ainsworth. An HTTP-Based Versioning Mechanism for Linked Data. In *LDOW*, 2010.
[7] C. Gutiérrez, C. A. Hurtado, and A. A. Vaisman. Temporal rdf. In *ESWC*, 2005.
[8] T. Heath and C. Bizer. Linked data: Evolving the web into a global data space. *Synthesis Lectures on the Semantic Web: Theory and Technology*, 2011.
[9] S. Hellmann, C. Stadler, J. Lehmann, and S. Auer. Dbpedia live extraction. In *OTM*, 2009.
[10] P. N. Mendes, H. Mühleisen, and C. Bizer. Sieve: Linked Data Quality Assessment and Fusion. In *LWDM*, 2012.
[11] F. Orlandi and A. Passant. Modelling provenance of DBpedia resources using wikipedia contributions. *Web Semantics*, 2011.
[12] L. Panziera, M. Comerio, M. Palmonari, F. D. Paoli, and C. Batini. Quality-driven extraction, fusion and matchmaking of semantic web api descriptions. *JWE*, 2012.
[13] A. Rula, M. Palmonari, A. Harth, S. Stadtmüller, and A. Maurino. On the diversity and availability of temporal information in linked open data. In *ISWC*, 2012.
[14] A. Rula, M. Palmonari, and A. Maurino. Capturing the age of linked open data: Towards a dataset-independent framework. In *DQMST*, 2012.
[15] J. Tappolet and A. Bernstein. Applied temporal rdf: Efficient temporal querying of rdf data with sparql. In *ESWC*, 2009.
[16] J. Umbrich, M. Hausenblas, A. Hogan, A. Polleres, and S. Decker. Towards dataset dynamics: Change frequency of linked open data sources. In *LDOW*, 2010.
[17] Y. Wang, M. Zhu, L. Qu, M. Spaniol, and G. Weikum. Timely yago: harvesting, querying, and visualizing temporal knowledge from wikipedia. In *EDBT*, 2010.