Data Conflict Resolution among Same Entities in Web of Data

Mojgan Askarizade
University of Isfahan, Isfahan, Iran
Mojgan.askarizade@gmail.com

Mohammad Ali Nematbakhsh
University of Isfahan, Isfahan, Iran
nematbakhsh@eng.ui.ac.ir

Enseih Davoodi Jam
University Of Isfahan, Iran
nc_davodi.softcom@eng.ui.ac.ir

Abstract

With the growing amount of published RDF datasets on similar domains, data conflict between similar entities (same-as) is becoming a common problem for Web of Data applications. In this paper we propose an algorithm to detect conflict of same properties values of similar entities and select the most accurate value. The proposed algorithm contains two major steps. The first step filters out low ranked datasets using a link analysis technique. The second step calculates and evaluates the focus level of a dataset in a specific domain. Finally, the value of the top ranked dataset is considered. The proposed algorithm is implemented by Java Platform and is evaluated by geographical datasets containing "country" entities.

Keywords: Semantic Web, Linked Data, data conflict, ranking, same entities

1. Introduction

Because of growing amount of data on the Web of data, number of the structured datasets has increased significantly in recent years. Since semantic Web allows machine to process data and retrieve data automatically, availability of high accurate structured data is required. Different datasets are provided by different organization may offer the same contents. For example, WordFactbook and Eurostat datasets overlap deeply, as they both describe country.

According to principle of Linked Data, entities are identified by URI [1]. The same real-word entity can be represented by different URIs correspond to each other with owl:sameAs links. Multiple datasets may provide different values for the same property of an entity [3]. For example datasets such as: DBpedia, Geonames, and WordFactbook publish different information about population of United Kingdom that are mentioned below:

Table 1. United Kingdom different data sets

Geonames	WorldFactbook	DBpedia
60776238	62348447	58789194

However, these data inconsistencies are not acceptable in the cases where quality data is needed, for example in the case of a commercial applications. Generally three different strategies are suggested to deal with such data inconsistency [6]:

- Conflict ignoring: Because data correction is vital in most circumstances, ignorance of data inconsistency is not acceptable in these cases.
- Conflict avoiding: The second strategy handle conflicting data by conflict avoidance. With respect to large dimensions of web of data, this strategy is impossible.

- Conflict resolving: This strategy detects same entities and resolves data conflict between their values. This is the most practical strategy to deal with data inconsistency.

We deal with the inconsistency by resolving data conflict, for this reason, two main approaches are describing here:

- Selection from available data: In this approach a value is selected from inconsistent values. For example, if the population of London is one of the different values (X, Y, Z), one of them must be chosen as the population of London.
- A new value generation: Using this approach a new value is generated using available values. For example, if the population of London is one of the different values (X, Y, Z), the result can be the average value of (X, Y, Z).

Our algorithm resolve data conflict by assigning new value to inconsistent data.

In this paper an algorithm is proposed to detect inconsistency among the same properties' values of sameAs entities in order to choose the most accurate data as output of algorithm. The proposed algorithm is divided into two steps. In the first step datasets are ranked by performing link analysis, then some of them are removed from list of datasets according to their ranks. In the second step the specialty of datasets are computed based on tow metrics: ontology and size of dataset. This paper has been shown that data of the dataset is more special are more accurate. At the end of the ranking the properties' values of top ranked dataset are considered as output.

The proposed algorithm is implemented using Java Platform and evaluated on geographical datasets that contain "country" entity. This paper is organized as follows: In section 2, the related works are presented. The proposed algorithm is described in section 3. The evaluation of proposed algorithm is presented in section 4.Finally conclusion and the outlook on future works are presented in section 5.

2. Related work

Due to novelty of web of data, here are a few researches which have been done to resolve data conflicts in web of data. One of the noticeable researches suggested by Bizer et al Proposes a framework aimed for resolving the inconsistency of data [9]. The main idea of their proposed framework is based on extracting data from different Wikipedia language version and comparing these data with each other to identify the most appropriate language version. For example, despite of better quality of English version as a whole, the population of Germany (as an example of required data) could be indicated more up-to-date in German version than Italian, French or even English version. They provide several strategies to recognize the correct Wikipedia version to choose from. To do this first, data of Wikipedia are converted to RDF format (by DBpedia project) and then they are published in web of data. Because all datasets use Dbpedia ontology, no adaptation is required so Dbpedia ontology is used. Therefore, it's enough to indicate the classes referring to a single entity, using a URI. In other steps, different heuristics are utilized to resolve the inconsistency of data retrieved from different resources. The main purpose of this research is to support the hypothesis that retrieving data from different language versions instead of a single one and combining them to obtain higher quality and more complete data. In this way, the French, Italian, German and English versions of Wikipedia were utilized.

3. The proposed algorithm

This paper provides an algorithm to resolve data conflicts among properties' values of same entities in different datasets. The algorithm works on domain-specific datasets published on a single domain as input and tries to select appropriate values. Figure 1 shows the overflow of algorithm in simple.

3.1. The first step of algorithm

In the first step datasets are published on a single domain are selected then they are ranked by performing link analysis. After finishing this step the low ranked datasets are removed from list of datasets. The Page Rank algorithm is used to rank datasets [14].

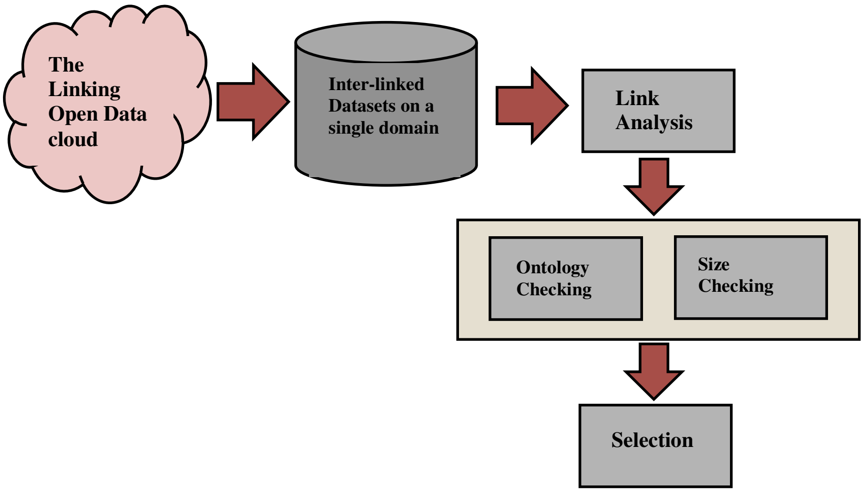
The Linking Open Data cloud
Inter-linked Datasets on a single domain
Link Analysis
Ontology Checking
Size Checking
Selection

Figure 1. General Workflow of Algorithm

The Page Rank algorithm which is widely used in most search engines such as Google could be easily used to rank linked data. By starting from a point and random surfing, this algorithm evaluates the probability of finding any given page. The algorithm assumes a link between a page i to a page j demonstrates the importance of page j. In addition, the importance of page j is associated to the importance of page i itself and inversely proportional to the number of pages i point to. To adapt this algorithm to web of data, any page considered as a dataset and links between pages considered as links between datasets. According to Page Rank Algorithm, the rank of dataset j in k level is equals to:

$$R^{k}(j) = \sum_{i \in B(j)} \frac{R^{k-1}(i)}{|L(i)|} \quad (1)$$

Let $B(j) = \{source(l) \mid \forall l \in L, t \arg et(l) = j\}$ be the set of datasets point to j and L(i) = {target(l) | $\forall l \in L$ } be the set of datasets linked by a dataset i. the operation is repeated until the algorithm reach to a specific threshold.

3.2. The second step of algorithm

There are several datasets have been published on a specific domain. Some datasets are general so they cover various domains. On the other hand, some datasets cover only specified domain. For example DBpedia is a general dataset covering different entities such as: persons, places, music albums, films etc. while LMDB is a specialized domain-specific dataset on film domain. The idea behind the algorithm is that the domain specific datasets have more accurate data than general datasets. For evaluation of specialty degree of a dataset tow criteria is considered: ontology and size of dataset.

3.2.1. Ontology

Ontology of a dataset should be evaluated to assess the semantic of dataset. As before mentioned, all selected datasets are published on a single domain such as people, place, film, book etc. at first one of the entities that defined in every selected datasets is chosen. Then the words that are semantically similar to chosen entity are extracted. This process is done by WordNet. The result is a list of synonym words. These words should be compared with synonyms list to calculate the counts of words are included in synonyms list. The specialty of ontology can be computed by using following fraction.

$$\frac{|SE|}{|TE|} \quad (2)$$

In this fraction |SE| is the number of entities that is included in synonym words and |TE| is the total number of entities.

3.2.2. Size

Another criterion is size of dataset. It's possible that a dataset be specialized but be small. It is assumed that the dataset is larger is more important and more reliable. So size of the dataset must be considered that is shown in fraction 2.

$$\frac{|ST|}{|TT|} \quad (3)$$

In above fraction |ST| is the number of entities' instances that match with words of synonyms list and |TT| is the number of instances of total entities. Finally combining two fractions 1 and 2 final score can be computed.

$$Score = 0.8 * \frac{|SE|}{|TE|} + 0.2 * \frac{|ST|}{|TT|} \quad (4)$$

4. Evaluation and implementation

The proposed algorithm has been implemented using Java Platform and evaluated via geographical domain datasets. Five major datasets of geographical domain (DBpedia, Geonames, WordFactbook, Eurostat, GeoLinked Data) are used as the input datasets of the proposed algorithm and the output is values of the most accurate data selected from these five datasets. In order to evaluate the proposed algorithm, 35 countries are selected and one property of country entity named population is examined.

In first step, datasets ranks are computed by Page Rank algorithm according link analysis. Result is shown in figure 2.

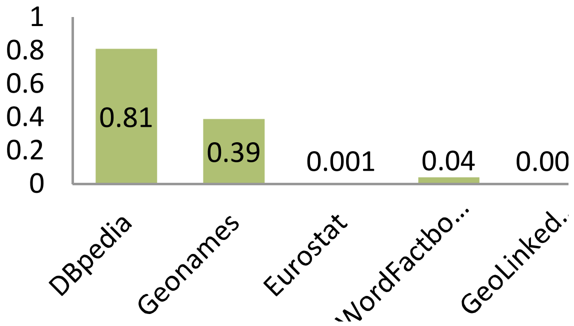

Figure 2. Ranking datasets by Page Rank

As depicted in Figure 2 DBpedia is the top ranked data set while GeoLinked Data and Eurostat are the low ranked data sets. In this stage the data sets whose rank scores are less than half of the rank scores belonging to the top ranked data set are removed from the assessment.

For gathering data we track the links between the remaining two data sets; DBpedia and Geonames. The population of the countries is considered as an attribute for evaluation. Different vocabularies are used in the Web of data. In DBpedia dbpprop:populationCensus and in Geonames gn:population is representative of the population attribute.

Data sets are evaluated based on their ontology and their size. Since the domain that we focused on is geographical data, we chose "country" as the input of WordNet dictionary. Synonyms of "country" are extracted as the output of WordNet. Some of them are land, state, city, fatherland, nation etc. which were named the synonyms list.

In the next step, we extract all entities from the data sets. For example the following SPARQL extracts all entities from DBpedia:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
SELECT distinct ?s WHERE {?s rdf:type owl:Class}
```

Finally we count the entities that contain any words in the synonyms list. The result of first the fraction in equation 4 is shown in figure 3.

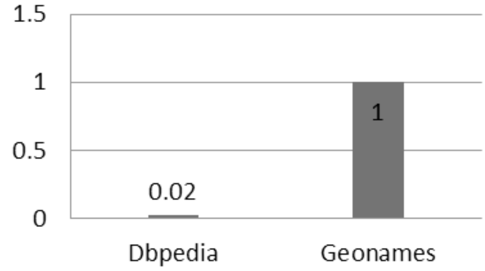

Figure 3. Result of first fraction

In the last stage the size of the data set is examined. The number of instances of specialized entities and total entities are calculated. For example "London" is an instance of a specialized entity. In table 2 one of the properties of "London" is displayed in form of a triple. As is shown, the object of the triple is dbpedia-owl:Place that contains "place" as a member of the specialized entity. And 'A Trip to the Moon' is an instance of non-specialized entity because neither the subject nor the object is in the specialized entities list. After calculating the number of specialized instances and total instances the result of the second fraction of equation 4 is shown in figure 4.

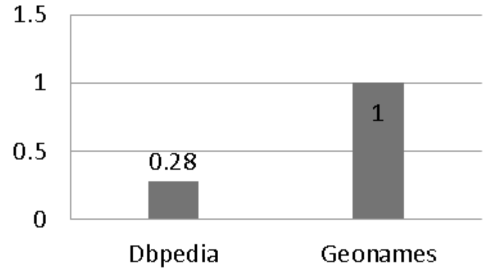

Figure 4. Result of Second Fraction

Finally, the last score with regard to the equation 4 has been calculated and is shown in figure 5. As it shows Geonames gains the highest score. Considering the proposed idea, Geonames' data is the most accurate. So when we are faced with data conflicts, Geonames' data must be chosen.

The WordFactbook is also illustrated in the above table. We can see if WordFactbook is not removed from the synonyms list it would have more accurate data compared with DBpedia data.

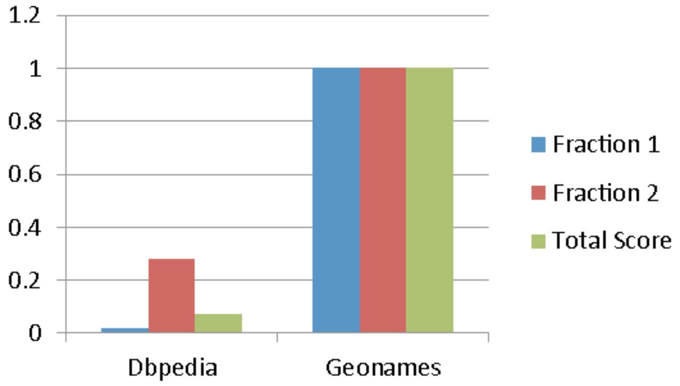

Figure 5. Final Ranking

In order to evaluate our algorithm we found real data on The Population Reference Bureau . The Population Reference Bureau informs people around the world about population, health, and the environment, and empowers them to use that information to advance the well-being of current and future generations. 33 countries were selected for evaluation. The results of our algorithm that are the most accurate values of populations are compared with the data found on The Population Reference Bureau. The numbers of the most accurate data are found on three data sets: WordFactbook, Geonames, DBpedia are shown in following table.

Table 2. The number of most accurate data

WordFactbook	Geonames	DBpedia
7	*22*	*4*

As we know WordFactbook and Geonames are domain-specific data sets on the geographic domain while, DBpedia is a general data set. The results indicate specialized data sets have more accurate data compared with general data sets. The accuracy of the final data set compared with the real data is 67%.

5. Conclusion and future works

This article proposes an algorithm to resolve data conflicts among property values of similar entities in the web of data. The proposed algorithm contains three steps. In the first step, the input data sets published on a single domain are ranked by performing link analysis. In the second step, the recognition of same entities is performed by tracking the sameAs links and vocabulary matching is done manually. In the last stage, the specialty of data sets is computed based on two criteria: ontology and the size of the data set. The proposed algorithm has been evaluated using geographic data sets that contain the "country" entity. Results were compared with real data obtained from The Population Reference Bureau. The data obtained by the proposed algorithm had a 67% matching degree with real data. Future work in relation to this article will focus on:

- Using different domains to evaluate the accuracy of the algorithm.
- Using PageRank algorithm applied on internal entities and intra-links of a data set in order to rank entities.

References

[1] T. Heath and C. Bizer, "Consuming Linked Data," in Linked Data: Evolving the Web into a Global Data Space, 1st edition. Synthesis Lectures on the Semantic Web: Theory and Technology, 2011.

[2] C. Bizer, "Quality-Driven Information Filtering in the Context of Web-Based Information Systems," PhD thesis, Freie Universität Berlin, 2007.

[3] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data - The Story So Far," Journal on Semantic Web and Information Systems, Special Issue on Linked Data, 2009.

[4] F. Naumann, A. Bilke, J. Bleiholder, and M. Weis, "Data Fusion in Three Steps: Resolving Schema, Tuple, and Value Inconsistencies," presented at IEEE Data Eng. Bull, 2006, pp.21-31.

[5] X.L. Dong, and F. Naumann, "Data fusion - Resolving Data Conflicts for Integration," Journal Proceedings of the VLDB Endowment, 2009, pp.1654-1655.

[6] J. Bleiholder and F. Naumann, "Data fusion", presented at ACM Comput. Surv., 2008.

[7] A. Nikolov, V.S. Uren, E. Motta, and A.N.D. Roeck, "Integration of Semantically Annotated Data by the KnoFuss Architecture," in 16th International Conference on Knowledge Engineering and Knowledge Management, 2008, pp.265-274.

[8] O. Hartig and J. Zhao, "Using Web Data Provenance for Quality Assessment," in Proc. SWPM, 2009.

[9] E. Tacchini, A. Schultz, and C. Bizer, "Experiments with Wikipedia Cross-Language Data Fusion," Proceedings of the 5th workshop on scripting and development for the semantic web, 2009.

[10] A. Bilke, J. Bleiholder, C. Böhm, K. Draba, F. Naumann, and M. Weis, "Automatic Data Fusion with HumMer", in Proceedings of the International Conference on Very Large Databases (VLDB), 2005, pp.1251-1254.

[11] P. Buitelaar, and T. Eigner, "Evaluating Ontology Search", In Proceedings of the 5th International Workshop on Evaluation of Ontologies and Ontology-based Tools (EON2007), 2007.

[12] J. Bleiholder and F. Naumann, "Declarative Data Fusion - Syntax, Semantics, and Implementation," in Proc. Databases and Information Systems (ADBIS), 2005, pp.58-73.

[13] W. Xing and A.A. Ghorbani, "Weighted PageRank Algorithm", in Proc. CNSR, 2004, pp.305-314.

[14] L. Page, S. Brin, R. Motwani, and T. Winograd," The PageRank Citation Ranking: Bringing Order to the Web ", Technical Report 1999-66, Stanford InfoLab (1999) .

[15] R. Delbru, N. Toupikov, M. Catasta, G. Tummarello, and S. Decker, "Hierarchical Link Analysis for Ranking Web Data", in Proc. ESWC (2), 2010, pp.225-239.

[16] N. Toupikov, J. Umbrich, R. Delbru, M. Hausenblas, and G. Tummarello, " DING! Dataset Ranking using Formal Descriptions", In: WWW 2009 Workshop: Linked Data on the Web LDOW2009 Madrid, Spain: (2009) .

[17] T. Berners-Lee. Linked data. [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html. (2011,October)

[18] C. Bizer, R. Cyganiak, T. Heath. How to publish linked data on the web. [Online]. Available: http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial.(2011 October)