A Preliminary Investigation Towards Improving Linked Data Quality Using Distance-Based Outlier Detection

Jeremy Debattista(✉), Christoph Lange, and Sören Auer

University of Bonn and Fraunhofer IAIS, Bonn, Germany
{debattis,langec,auer}@cs.uni-bonn.de

Abstract. With more and more data being published on the Web as Linked Data, Web Data quality is becoming increasingly important. While quite some work has been done with regard to quality assessment of Linked Data, only few works have addressed quality improvement. In this article, we present a preliminary an approach for identifying potentially incorrect RDF statements using distance-based outlier detection. Our method follows a three stage approach, which automates the whole process of finding potentially incorrect statements for a certain property. Our preliminary evaluation shows that a high precision is maintained with different settings.

Keywords: Outlier detection · Data quality · Linked data

1 Introduction

A rationale of the Semantic Web is to provide real-world things, also called *resources*, with descriptions in common data formats that are meaningful to machines. Furthermore, Linked Data emphasises on the reuse and linking of these resources, thus assisting in the growth of the *Web of* (meaningful) *Data*. Schemas, some being lightweight and others being more complex, have been defined for various use cases and application scenarios in order provide structure to the descriptions of semantic resource based on a common understanding. Nevertheless, since linked datasets are usually originating from various structured (e.g. relational databases), semi-structured (e.g. Wikipedia) or unstructured sources (e.g. plain text), a complete and accurate *semantic lifting* process is difficult to attain. Such processes can often contribute to incomplete, misrepresented and noisy data, especially for semi-structured and unstructured sources. Issues caused by these processes can be attributed to the fact that either the knowledge worker is not aware of the various implications of a schema (e.g. incorrectly using inverse functional properties), or because the schema is not well defined (e.g. having an open domain and range for a property). In this article, we are concerned with the latter, aiming to identify potentially incorrect statements in order to improve the quality of a knowledge base.

Y.-F. Li et al. (Eds.): JIST 2016, LNCS 10055, pp. 116–124, 2016.
DOI: 10.1007/978-3-319-50112-3_9

When analysing the schema of the DBpedia dataset we found out that from around 61,000 properties, approximately 59,000 had an undefined domain and range. This means that the type of resources attached to such properties as the subject or the object of an RDF triple can be very generic, i.e. `owl:Thing`. Whilst this is not forbidden, it makes a property ambiguous to use. For example, the property `dbp:author`, whose domain and range are undefined, has instances where the subject is of type `dbo:Book` and the object of type `dbo:Writer`, and other instances where the subject is of type `dbo:Software` and the object of type `dbo:ArtificialSatellite`.

The key research question in this paper is *can distance-based outlier techniques help in identifying quality problems in linked datasets?* In this article we investigate how triples can be clustered together based on their distance. This distance is identified by a semantic similarity measure that takes into consideration the subject type, object type, and the underlying schema. Furthermore, we evaluate complementary aspects of the proposed approach. More specifically, we were interested to see how different settings in our approach affect the precision and recall values.

This article is structured as follows. The state-of-the-art is described in Sect. 2. Our proposed approach is explained in Sect. 3. Experiments of our approach are documented in Sect. 4. Conclusions and an outlook to future work are discussed in Sect. 5.

2 Related Work

Various research efforts have tackled the problem of detecting incorrect RDF statements using different techniques. These include *statistical distribution* [8], *schema enrichment* [9,12] and *crowdsourcing* [1,10]. Outlier detection techniques such as [11] are used to validate the correctness of data literals in RDF statements, which is out of the scope of this research as our approach considers only statements where the subject and object are resources.

Statistical Distribution. Paulheim et al. [8] describe an algorithm based on the statistical distribution of types over properties in order to identify possibly faulty statements. Statistical distribution was used in order to predict the probability of the types used on a particular property, thus with some confidence verify the correctness of a triple statement. Their three step approach first computes the frequency of the predicate and object combination in order to identify those statements that have a low value. Cosine similarity is then used to calculate a confidence score based on the statement's subject type probability and the object type probability. Finally, a threshold value is applied to mark those statements that are potentially incorrect. Our approach uses semantic similarity to identify whether a statement could be a possibly incorrect statement or not, instead of statistical distribution probabilities. Therefore, our similarity approach takes into consideration the semantic topology of types and not their statistical usage.

Schema Enrichment. Schema enrichment is also a popular technique to detect incorrect statements. Töpper et al. [9] enrich a knowledge base schema with additional axioms before detecting incorrect RDF statements in the knowledge base itself. Such an approach requires external knowledge in order to enrich the ontology. Similarly, Zaveri et al. [12] apply a semi-automated schema enrichment technique before detecting incorrect triples.

Crowdsourcing WhoKnows? [10] is a crowdsourcing game where users contribute towards identifying inconsistent, incorrect and doubtful facts in DBpedia. Such crowdsourcing efforts ensure that the quality of a dataset can be improved with more accuracy, as a human assessor can identify such problems even from a subjective point of view. During the evaluation, the users identified 342 triples that were potentially inconsistent from a set of overall 4,051 triples, reporting a precision value of 46%. A similar crowdsourcing effort was undertaken by Acosta et al. in [1]. They used pay-per-hit micro tasks as a means of improving the outcome of crowdsourcing efforts. Their evaluation focuses on checking the correctness of the object values and their data types, and the correctness of interlinking with related external sources, thus making it incomparable to our approach. In contrast to crowdsourcing, our preliminary approach gives a good precision in identifying outliers without the need of any human intervention, in an acceptable time (± 3 min to compute outliers of a 10 K dump). Nonetheless, at some point, human expert intervention would still be required (in our approach) to validate the correctness of the detected outliers, but with any (semi-)automatic learning approaches, human intervention is reduced.

3 Improving Dataset Quality by Detecting Incorrect Statements

The detection and subsequent cleaning of potentially incorrect RDF statements aids in improving the quality of a linked dataset. There were a number of attempts to solve this problem in the best possible manner (cf. Sect. 2). We apply the distance-based outlier technique by Knorr et al. [6] in a Linked Data scenario. Exploiting reservoir sampling and semantic similarity measures, clusters of RDF statements based on the statement' subject and object types are created, thus identifying the potentially incorrect statements. We implemented[1] this approach as a metric for *Luzzu* [2].

3.1 Approach

Following [6], our proposed Linked Data adapted method has three stages: *initial*, *mapping*, and *colouring*. These three stages automate the whole process of finding potentially incorrect statements for a certain property. In the *initial* stage, k (the size of the reservoir) RDF statements are added to a reservoir sampler. Following the initialisations of the constants, the *mapping* stage groups data objects in various cells based on the mapping properties described in [6]. Finally, the *colouring* stage identifies the cells that contain outlier data objects.

[1] The Java code can be found in our GIT repository: https://goo.gl/bGRKxi.

Initial Stage. The initial steps are crucial for achieving a more accurate result, i.e. a better identification of potentially incorrect statements. We start by determining the approximate distance D that is used in the second stage to condition the mapping, and thus the final clustering of RDF statements. The approximate value D is valid for a particular property, i.e. the property whose triples are being assessed. Therefore, two properties (e.g. *dbp:author* and *dbp:saint*, i.e. the patron saint of, e.g., a town) will have different values of D according to the triples, their types, and ultimately the similarity measure chosen. Currently, in our approach we assume that a resource is typed with **only** one class, choosing the most specific type if a resource is multi-typed (e.g. *dbo:Writer* and not *dbo:Person*). Additionally, a threshold fraction p (between 0 and 1) is defined by the user during the initial phase, affecting the number of data objects in a cluster M. Therefore, p can be considered to be a sensitivity function that increases or decreases the amount of data objects in a cluster.

Determining the Approximate Distance. Our approach makes use of reservoir sampling as described in [3]. The rationale is that D is approximated by a sample of the data objects being assessed, to identify the acceptable maximum distance between objects mapped together in a cell, in a quick and automated way. To determine the approximate distance we applied two different implementations (cf. Sect. 4 for their evaluation), one based on a simple sampling of triples and another one based on a modified reservoir sampler, which we call the *type-selective*. From the sample set (for both implementations), a random data object is chosen to be the *host*, and is removed from the sampler. All remaining statements in the sampler are semantically compared with the host individually and their distance values are stored in a list. The median distance is than chosen from the list of distances. We chose the median value over the mean value as a central tendency since the latter can be influenced by outliers.

In the first implementation (simple sampling), the reservoir selects a sample of triples, irrelevantly of their subject and object types. The main limitation is that, irrelevantly of the size of the reservoir, the approximate distance D value can bias towards the more frequent pairs of the subject and object types. Therefore, the sampler might not represent the broad types attached to the particular property being assessed.

In order to attempt to solve the *sampler representation problem*, we propose the *type-selective* reservoir sampler. The proposed reservoir sampler modifies the simple sampler by adding a condition that only one statement with a certain subject type and object type can be added to the reservoir. In other words, when there are two distinct statements with matching subject types and object types, only one of these statements will be added to the reservoir.

Mapping Stage. The mapping stage attends to the clustering of data objects (i.e. RDF statements in our case) in cells. An RDF statement is chosen at random from the whole set of data objects and is placed in a random cell. This is called the *host* cell. Thereafter, every other RDF statement in the dataset is mapped to an appropriate cell by first comparing it to the data object in this host cell.

Semantic Similarity Measure. In order to check if an RDF statement fits in a cell with other similar RDF statements, a semantic similarity measure is used. More specifically, since we are mostly concerned about the distance between two statements, we use a normalised semantic similarity measure. The similarity between two statements S_1 and S_2 is defined as the average of the similarity between the statements' subjects, and the similarity between the statements' objects.

Colouring Stage. After mapping all data objects to the two-dimensional space, the *colouring* process colours cells to identify outlier data objects, based on the process identified in [6]. In [6], the minimum number of objects (M) required in a cell such that data objects are not considered as outliers is calculated as $M = N \cdot (1 - p)$ where N is the total number of data objects, and p is the threshold fraction value determined in the *initial* stage.

4 Experiments and Evaluations

The primary aim of this experiment is to compare if the automatic approach of setting approximate D value gives an advantage over the manual setting. All experiments in this part of the evaluation used the same similarity measure configuration, i.e. Zhou IC [13] with the Mazandu measure [7], as implemented in the Semantic Measures Library & Toolkit [4].

This experiment is split into two sub-experiments. In the first part, we evaluated triple statements in DBpedia with the predicate http://dbpedia.org/property/author using the proposed approach with the p and D parameters manually set to determine the precision and recall values. In the second part of this evaluation we repeat this experiment but the value of D is determined by the two automated approaches described in Sect. 3. For both experiments, p was set to: 0.99, 0.992, 0.994, 0.996, and 0.998.

Sub-experiment #1 – Setting Approximate D Manually. In this manual experiment, the D value for the evaluated property was obtained as an estimate from a manual calculation of the similarity values of the different types. From Fig. 1, we observe that on average our approach achieved around 76% precision. On the other hand, the recall values were low, with an average of 31%. We also observed that increasing the approximate value D does not result in an increasing precision. For example, in Fig. 1 we spot that the precision value for the D value of 0.3335 is greater than that of 0.3555 when p was set to 0.996. When D was set to 0.3555, 39 more outliers were detected, (true positives –7, false positives +42 data objects). This slight change in *true positives* and *false positives* was expected as the data objects cluster with similar data objects whose distance is the smallest. Therefore, the change in D might have moved some objects from one cell to another with the consequence that a previously non-outlier cell is now marked as an outlier, since a number of data objects might have moved to other

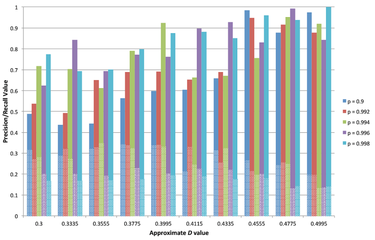

Fig. 1. The precision and recall values for the authors property dump with different values for D and p. The solid bars denote precision values, whilst the striped overlapped bars denote recall.

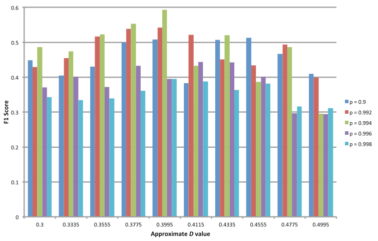

Fig. 2. The F1 score authors property dump with different values for D and p.

cells. Figure 2 represents the F1 score for the authors property dump manual experiment, showing an average of almost 43% for this harmonic mean score.

Sub-experiment #2 – Setting Approximate *D* Automatically. The same evaluated property was used in this experiment, where first an approximate D value was calculated first using the *simple* reservoir sampler and then using the *type-selective* reservoir sampler. A single host was chosen randomly from these reservoir samplers, together with a starting host location. The choice of a random data object will not affect the precision of the algorithm, as all data objects will be compared and mapped in suitable cells. From Fig. 3 we observe that the *type-selective* sampler outperforms its simpler counterpart for all p values with regard to the precision. One possible reason is due to the low approximate D values identified by the simple reservoir sampler. Low approximate D values mean that less data objects get mapped together in cells, since the approximate distance becomes smaller and data objects will be dispersed throughout the whole 2D space. This means that since less data objects are mapped in the same cell or surrounding cells, it would be more difficult to reach the $M + 1$ quota, and thus more cells will be marked as outliers. Therefore, whilst a low approximate D could lead for a decrease of *false positives* in non-outlier cells, it can also increase of *false negatives* (thus decreasing *true positives*), as objects that should not be marked as outliers could end up in outlier-marked cells. The main factors that affect the approximate D value are (1) the choice of the semantic similarity measure, and (2) the underlying schema (cf. limitations in Sect. 4.1). Furthermore, this approximate D value and the user-defined sensitivity threshold value (p) affect the precision and recall.

Following these experiments, in Fig. 4 we compared the *type-selective* precision and recall results for every p against the manual approach. For this comparison we used the manual scores that got the highest F1 measure for each p value, thus having a balance between the precision and recall. Figure 4 shows that the manual approach performed overall better than the automatic one in terms of

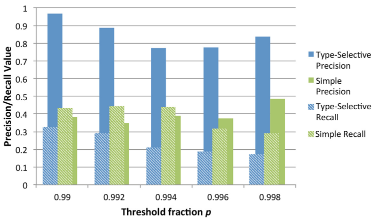

Fig. 3. The precision and recall values for the authors property dump with different values for p and a generated D value.

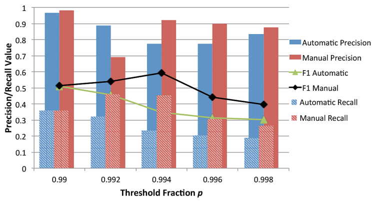

Fig. 4. Precision and recall values for the authors property dump comparing the manual results against the automatic results for multiple values of the fraction p.

the F1 measure. Nevertheless, in most cases, there are no large discrepancies between the two. The automatic approach resulted into a higher approximate D value than the manual approach. The approximation D value for the automatic approach was 0.482147, 0.0826 more than the given manual approximation D value with the highest F1 value (i.e. 0.3995 for threshold fraction p).

4.1 Discussion

The led evaluation is as yet not conclusive, since we only evaluated our approach with one property. This evaluation also showed that our approach produces a low recall value and thus a low F1 measure. A higher recall, without comprising the precision, would have been ideal, as with low recall we are missing a relevant data objects that should have been marked as outliers. One must also note that the choice of a semantic similarity measure will also affect the precision and recall values of such an approach, in a way that its results are the deciding factor where a data objects is mapped.

Nevertheless, our approach has a number of known limitations:

1. the approach is limited to knowledge bases without blank nodes, which can effect the degree of similarity, thus making this approach less robust and generic;
2. the approach does not fully exploit the semantics of typed annotations in linked datasets, since our approach assumes that an instance is a member of only one type, in particular the most specific type assigned to the resource;
3. the evaluated semantic similarity measures are limited to hierarchical '*is-a*' relations that might be more fitting to biomedical ontologies having deep hierarchies;
4. the sampled population might not reflect the actual diverse population of the data objects that have to be clustered in both sampler implementations. Thus, with both implementations we will not achieve the best representative sample, such as that obtained by stratified sampling [5];

5. whilst with the *simple* sampler outliers might occur in the sample population, with the *type-selective* sampler there is a 100% certainty that outlier data objects are present in the sample that determines the approximate D. Knorr et al. [6] had foreseen this problem and whilst suggesting that sampling provides a reasonable starting value for D, it cannot provide a high degree of confidence for D because of the unpredictable occurrence of outliers in the sample.

5 Conclusions

In this article we investigated the possibility of detecting potentially incorrect RDF statements in a dataset using a time and space efficient approach. More specifically, we applied a distance-based clustering technique [6] to identify outliers in a Linked Data scenario. While providing satisfactory results, our approach has a number of limitations that we are currently addressing. However, the preliminary results give us an indication on the research question set in the introduction. In the future, we aim to extend our experiments by using semantic relatedness measures instead of the semantic similarity measures, thus our distance based measure will also consider the semantic relationships between two terms, such as `owl:equivalentClass`.

References

1. Acosta, M., Zaveri, A., Simperl, E., Kontokostas, D., Auer, S., Lehmann, J.: Crowdsourcing linked data quality assessment. In: Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J.X., Aroyo, L., Noy, N., Welty, C., Janowicz, K. (eds.) ISWC 2013. LNCS, vol. 8219, pp. 260–276. Springer, Heidelberg (2013). doi:10.1007/978-3-642-41338-4_17
2. Debattista, J., Auer, S., Lange, C.: Luzzu - a framework for linked data quality analysis. In: 2016 IEEE International Conference on Semantic Computing, Laguna Hills (2016)
3. Debattista, J., Londoño, S., Lange, C., Auer, S.: Quality assessment of linked datasets using the approximation. In: 12th European Semantic Web Conference Proceedings (2015)
4. Harispe, S., Ranwez, S., Janaqi, S., Montmain, J.: Semantic measures for the comparison of units of language, concepts or entities from text and knowledge base analysis, October 2013. arXiv abs/1310.1285
5. Hausman, J.A., Wise, D.A.: Stratification on endogenous variables and estimation: the gary income maintenance experiment. In: Manski, C.F., McFadden, D.L. (eds.) Structural Analysis of Discrete Data with Econometric Applications. MIT Press, Cambridge (1981)
6. Knorr, E.M., Ng, R.T., Tucakov, V.: Distance-based outliers: algorithms and applications. VLDB J. **8**(3–4), 237–253 (2000)
7. Mazandu, G.K., Mulder, N.J.: A topology-based metric for measuring term similarity in the gene ontology. Adv. Bioinf. **2012**, 1–17 (2012)
8. Paulheim, H., Bizer, C.: Improving the quality of linked data using statistical distributions. Int. J. Semant. Web Inf. Syst. **10**(2), 63–86 (2014)

9. Töpper, G., Knuth, M., Sack, H.: DBpedia ontology enrichment for inconsistency detection. In: Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS 2012, pp. 33–40. ACM, New York (2012)
10. Waitelonis, J., Ludwig, N., Knuth, M., Sack, H.: WhoKnows? - evaluating linked data heuristics with a quiz that cleans up DBpedia. Int. J. Interact. Technol. Smart Educ. (ITSE) **8**(3), 236–248 (2011)
11. Wienand, D., Paulheim, H.: Detecting incorrect numerical data in DBpedia. In: Presutti, V., d'Amato, C., Gandon, F., d'Aquin, M., Staab, S., Tordai, A. (eds.) ESWC 2014. LNCS, vol. 8465, pp. 504–518. Springer, Heidelberg (2014). doi:10.1007/978-3-319-07443-6_34
12. Zaveri, A., Kontokostas, D., Sherif, M.A., Bühmann, L., Morsey, M., Auer, S., Lehmann, J.: User-driven quality evaluation of DBpedia. In: Proceedings of the 9th International Conference on Semantic Systems, I-SEMANTICS 2013, pp. 97–104. ACM, New York (2013)
13. Zhou, Z., Wang, Y., Gu, J.: A new model of information content for semantic similarity in wordnet. In: FGCNS 2008 Proceedings of the 2008 Second International Conference on Future Generation Communication and Networking Symposia, vol. 3, pp. 85–89. IEEE Computer Society, December 2008