Undefined 1 (2009) 1–5
IOS Press

DRX: A LOD dataset interlinking recommendation tool

Editor(s): Name Surname, University, Country
Solicited review(s): Name Surname, University, Country
Open review(s): Name Surname, University, Country

Alexander Arturo Mera Caraballo, Bernardo Pereira Nunes and Marco A. Casanova
Department of Informatics, PUC-Rio
Rua Marquês de São Vicente, 225 Gávea, Rio de Janeiro, RJ, Brazil, Zip code: 22451-900
E-mail: {acaraballo,bnunes,casanova}@inf.puc-rio.br

Abstract. With the growth of the Linked Open Data (LOD) cloud, data publishers face a new challenge: finding related datasets to interlink with. To face this challenge, this paper describes a tool, called DRX, to assist data publishers in the process of dataset interlinking and browsing the LOD cloud. DRX is organized in five main modules responsible for: (i) collecting data from datasets on the LOD cloud; (ii) processing the data collected to create dataset profiles; (iii) grouping datasets using clustering algorithms; (iv) providing dataset recommendations; and (v) supporting browsing the LOD cloud. Experimental results show that DRX has a potential to be used as a dataset interlinking facilitator.

Keywords: Dataset Recommendation, Dataset profiling, Dataset Clustering, Linked Data, Semantic Web

1. Introduction

Despite the efforts to foster publishing data as Linked Open Data (LOD) [7], data publishers still face difficulties to integrate their data with other datasets available on the Web [3]. However, defining RDF links between datasets helps improve data quality, allowing the exploration and consumption of the existing data.

We may divide the question of defining RDF links between two datasets into two problems. We refer to the problem of creating RDF links between a *source dataset* and a *target dataset* as the *dataset interlinking problem* and to the problem of recommending target datasets to be interlinked with a given source dataset as the *dataset interlinking recommendation problem*.

In this paper, we present an overview of the DRX tool, designed to address the dataset interlinking recommendation problem. Briefly, DRX works as follows. The tool first collects data from datasets on the LOD cloud, creates dataset profiles and groups the datasets using a clustering algorithm. The tool stores these profiles internally for later use. When a data publisher wants to interlink a source dataset d_j with other datasets, the tool applies the same profiling technique to d_j and outputs an ordered list of datasets whose profiles best match with the profile of d_j. The paper also contains experiments whose results suggest that DRX indeed has a potential to be used for dataset interlinking recommendation.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 introduces the architecture of the tool and describes in detail its modules. Section 4 presents the DRX features through a case study. Section 5 describes the experiments conducted and discusses their outcome. Finally, Section 6 concludes the work and presents suggestions for future work.

2. Related Work

A review of the literature reveals that several tools, such as LIMES [15] and SILK [19], have been specif-

ically constructed to help address the dataset interlinking problem, but relatively few studies have been dedicated to the dataset interlinking recommendation problem. For instance, Leme et al. [11] created a method based on the naïve Bayes classifier to generate a ranked list of related datasets. The relatedness between datasets was measured using *linksets*, a set of existing links between datasets, retrieved from the Datahub[1] catalog. Similarly, Lopes et al. [12] took advantage of *linksets* to provide dataset interlinking recommendations. They used link prediction measures, from the social network analysis field, to estimate the probability of datasets being interconnected.

Nikolov et al. [14] investigated the use of a Semantic Web index (Sig.ma [20]) to identify candidate datasets for interlinking. Sig.ma is queried with text literals extracted from *rdfs:label*, *foaf:name* and *dc:title* properties from a given dataset to find the most overlapping datasets w.r.t to instances. Instead of using instances, Emaldi et al. [4] relied on the structural characteristics of datasets using a frequent subgraph mining (FSM) technique to identify and possibly establish links between disparate datasets. FSM is an interesting alternative to provide a more efficient approach as it only uses the most frequent subgraphs from a dataset to perform the analysis.

Dataset profiling/summarization techniques are also related to dataset interlinking recommendation. These techniques aim at elaborating a concise but comprehensive version of datasets. Thus, techniques such as those proposed by Lalithsena et al. [10] may ease the dataset interlinking recommendation process. They use reference datasets such as DBpedia to enrich a set of instances from a given dataset and to create a general description. A similar approach is proposed by Fetahu et al. [5], which created structured dataset profiles. Their approach combines several techniques, such as sampling, named entity recognition, topic extraction methods [13] and ranking [18], to represent a dataset. A more generic approach to create profiles is presented by Kawase et al. [9], which generates histograms for text-based resources on the Web based on the 23 top-level categories of the Wikipedia ontology.

There is a limited number of studies devoted to the dataset interlinking recommendation problem, and this number decreases when it comes to tools. For example, the *Tripleset Recommendation Tool - TRT* [1] provides a recommendation method, based on Social Network link prediction theory [12], to estimate the likelihood of the existence of a link between datasets. Unlike DRX, TRT must know in advance a set of related datasets to provide a new dataset recommendation. In this case, DRX could be used as a prior and complementary step for the TRT tool. The *Tripleset Recommendation Tool Based on Supervised Learning Algorithms Tool - TRTML* [2] combines supervised learning algorithms and link prediction measures to provide dataset recommendations. It relies on the similarity between vocabularies, classes and properties used in disparate datasets. However, using such features to build a predictive model is not the best choice. As mentioned by [7], there is a trend towards the adoption of well-known vocabularies that makes such features not very discriminative. Moreover, many datasets are overlooked by TRTML as it considers only triplified datasets. DRX overcomes these problems by offering recommendations based on the dataset content.

3. The DRX architecture

DRX is based on five modules, depicted in Figure 1, which are distributed in three different layers: data acquisition, data processing and application. These modules perform five main tasks:

1. Collect data from datasets in the LOD cloud.
2. Process the data collected to create dataset profiles, called *fingerprints*.
3. Group *fingerprints*, using clustering algorithms.
4. Provide dataset recommendations.
5. Support browsing the dataset profiles.

The data acquisition layer includes the *crawling* module, which discovers metadata about the LOD datasets from LOD catalogs (such as the Mannheim[2] catalog) as well as from manually submitted data. LOD catalogs typically stores metadata such as maintainer, SPARQL endpoint, relationships, VoID vocabulary, tags, license and resources. The crawling module uses the CKAN API [22] to query metadata available in such catalogs.

Since data can be in many different formats, the data acquisition layer also provides a set of specialized text extractors, that the crawling module uses to extract text literals from the datasets. Once a dataset is located, the crawling module creates a document containing the re-

[1] http://datahub.io

[2] http://linkeddatacatalog.dws.informatik.uni-mannheim.de

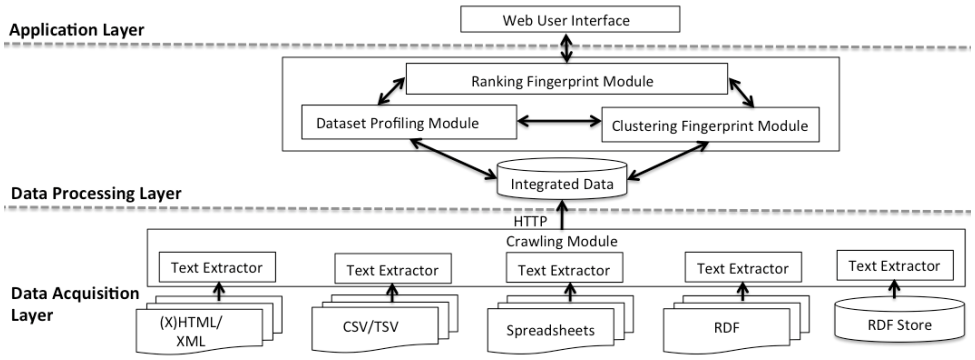

Fig. 1. Architecture DRX tool.

trieved text literals; the module ignores datasets with no text literals, since the technique implemented in the profiling module requires text literals.

The data processing layer includes three main modules: *profiling*, *clustering* and *ranking*.

The profiling module processes the documents retrieved from a dataset and computes a description that characterizes the content stored in the dataset. DRX implements the technique described in [9], that generates dataset profiles or *fingerprints* from text literals. The technique has five steps:

1. Extract entities from a given text literal.
2. Link the entities to English Wikipedia articles.
3. Extract the categories of the articles.
4. Follow the path from each extracted category to its top-level category and compute a vector with scores for the top-level categories thus obtained (such as agriculture, applied science, arts, belief, business, chronology, culture and so on).
5. Perform a linear aggregation in all dimensions of the vectors to generate the final profile, represented as a histogram for the 23 top-level categories[3] of the English Wikipedia.

The clustering module groups together fingerprints that are similar. The top-level categories of the English Wikipedia act as a set of features. The DRX tool implements the X-Means clustering algorithm, which is part of the WEKA [6] suite, and includes an efficient estimation of the number of clusters [6,16].

The last module implements two strategies to provide recommendations for a given dataset d_t: *cluster-based* and *profiling-based* strategies. The first strategy recommends datasets in the same cluster as d_t, whereas the profiling based strategy considers all datasets identified by fingerprints. Independently of the strategy chosen, for a given dataset d_t, the dataset recommendation module outputs a list of datasets ordered by the probability of being interlinked.

Assume that d_t is in cluster C_{d_t}. The cluster-based strategy creates a ranked list by taking into account only the distance between the fingerprint of d_t and the fingerprints of the other datasets in C_{d_t}. By contrast, the profiling-based strategy creates a recommendation list based on the distance between the fingerprint of d_t and the fingerprints of all other profiled datasets. We note that the distance function is a parameter of the recommendation algorithm; in the experiments, we adopted the cosine distance.

4. DRX GUI and Case Study

DRX is available at [21]. The DRX GUI allows the user to browse the LOD cloud by using dendrograms, tables and coordinate graphs (see Figure 2) and does not require any expertise in Semantic Web technologies or languages. The user may register a new source dataset s or select the source dataset s from those already in the LOD cloud. In either case, the user may then request recommendations for target datasets.

To illustrate how DRX works, we selected an independent dataset, *rkb-explorer-newcastle*[4], created jointly with other datasets under the ReSIST[5] project. These datasets are available through the RKBExplorer[6] Semantic Web browser that supports the Com-

[3] https://en.wikipedia.org/wiki/Category:Main_topic_classifications

[4] http://linkeddatacatalog.dws.informatik.uni-mannheim.de/dataset/rkb-explorer-newcastle

[5] http://www.resist-noe.org/

[6] http://www.rkbexplorer.com/

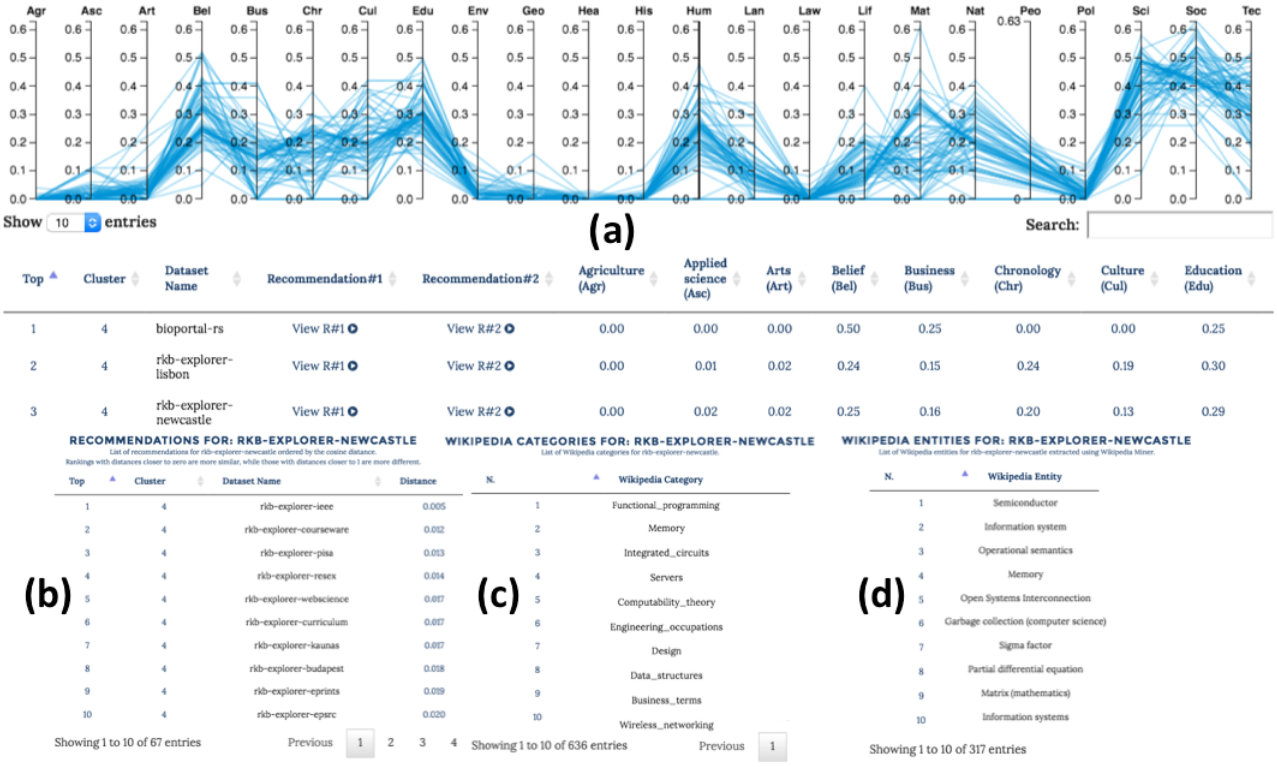

Fig. 2. Example of the use of the DRX.

puter Science research domain. It combines information from multiple heterogeneous sources, such as published RDF sources, personal Web pages and data bases in order to provide an integrated view of this multidimensional space.

In what follows, we refer to the five steps of our technique, introduced in Section 3. The goal of the first step is to collect text literals from data sources. If the user wants to consider a new dataset, he will first register it in the Mannheim catalog and submit its Mannheim URL entry to the tool. If the user selects an existing dataset, then the crawling module has already collected text literals, with the help of text extractors. In the case study, the *rkb-explorer-newcastle* dataset was crawled, using its SPARQL endpoint, to extract the text literal values of the *rdfs:label*, *skos:altLabel* and *skos:prefLabel* properties.

The second step is carried out transparently to the user. Here, the text literals collected in the first step are used as input to the profiling module to create a description of the dataset content through a fingerprint. Table 1 presents the fingerprint generated for the case study dataset, where the 23-dimension vector shows peaks for “Society”, “Technology” and “Science”, categories that are strongly related to the data content that the *rkb-explorer-newcastle* dataset provides.

To facilitate the exploration and selection of datasets, it is important to reduce the search space of datasets in the LOD cloud. Therefore, the third step generates clusters of datasets that share a certain similarity. The clustering module implements a simple interface that allows users to enter input parameters (such as the minimum and maximum number of clusters and the number of seeds and to execute the clustering process for all collected LOD datasets.

In the case study, we used a minimum of 8 clusters, since this is the number of categories of the LOD diagram[7]. The maximum number of clusters and the number of seeds were set to 10.

The results of the clustering process is represented as a dendrogram that allows users to navigate over the clusters and their respective members. For the case study, the clustering process generated 9 clusters; the *rkb-explorer-newcastle* dataset belongs to cluster #4.

The user interface also offers a zoom-in/out feature which allows users to explore the members of

[7] http://lod-cloud.net/

each cluster in more detail; the user has to click inside a cluster area to zoom-in the cluster. For example, after zooming in cluster #4, we may observe some of its members, such as *rkb-explorer-roma*, *bioportal-cheminf* and *rkb-explorer-newcastle*.

The user interface also provides a table with relevant information about the members of a selected cluster (see Figure 2(a)). This table offers column sorting and full text search. Each row shows the following information: the *"top"* column provides a dataset ranking based on the centrality of the datasets in the cluster; the *"cluster"* column represents the cluster membership; the *"dataset name"* column is a link to the dataset page in the Mannheim catalog; the *"Recommendation#1"* column provides recommendations, for the dataset d_j on the selected row, from datasets of the same cluster as d_j; the *"Recommendation#2"* column provides recommendations, for the dataset d_j on the selected row, based on all datasets available and, finally, the other columns show the vector with the 23 top-level categories.

For the case study, regarding cluster #4, Figure 2(a) shows detailed information of 10 members of cluster #4 out of a total of 67. The *rkb-explorer-newcastle* dataset was assigned the third position in the list, based on its centrality degree in the cluster.

Additionally, the user interface offers a feature to obtain interlinking recommendations. The user simply selects a dataset from the table in Figure 2(a), then select one of the recommendation strategies and finally clicks on the corresponding cell of column. For the case study the first recommendation strategy was selected. Then, a table is displayed, containing a list sorted by ascending order of the cosine distance values (see Figure 2(b)). For the case study, 10 recommendations, of a total of 67 are displayed. Note that the top ten dataset recommendations are from the project that *rkb-explorer-newcastle* belongs to (see Figure 2(b)).

Finally, the user interface provides two tables containing the set of categories and entities extracted from the dataset in analysis. For the case study, Figure 2(c) and Figure 2(d) show the categories and entities obtained, respectively.

5. Evaluation

5.1. Data and Evaluation Metrics

The approach that DRX implements was assessed using data retrieved from the Mannheim catalog, a metadata repository for open datasets. Through the CKAN API, the catalog enables querying dataset metadata, including two multivalued properties (*relationships* and *extras*), which in turn allow data publishers to assert that a dataset links to another. Both properties were used to retrieve the linksets between datasets in the Mannheim catalog. During the crawling step, we retrieve all datasets that have at least one resource or data associated. In early 2016, the data collected amounts to 387 datasets that were profiled. However, during the evaluation, we considered a total of 165 datasets that were profiled and belong to the LOD diagram.

Table 1
Generated fingerprint for the *rkb-explorer-newcastle* dataset.

Category	**value**	Category	**value**
Agriculture	0	Humanities	0.16
Applied Science	0.02	Language	0.08
Arts	0.02	Law	0.01
Belief	0.25	life	0.09
Business	0.16	Mathematics	0.26
Chronology	0.20	Nature	0.25
Culture	0.13	People	0
Education	0.29	Politics	0.02
Environment	0.02	Science	0.49
Geography	0.01	Society	0.42
Health	0	Technology	0.42
History	0	-	-

As in [1,2,4,11,12], linksets were used to define the gold standard for the dataset interlinking recommendation approach of the DRX tool. That is, the evaluation consisted in removing the existing linksets between datasets and verifying to what extent DRX was able to include known interlinked datasets in the recommendation lists it produces. The performance of DRX was measured using the overall *Mean Average Precision* (MAP), defined in what follows.

Note that the gold standard comprises only the datasets listed in the Mannheim catalog for which the fingerprints could be computed. We deemed as unsuitable datasets with no associated data or with inaccessible endpoints, even if their metadata would indicate the existence of linksets. Clearly, there is no reason to recommend a dataset that is not accessible to participate in an interlinking process.

More precisely, let d_t be a *source dataset* for which one wants to recommend datasets to be interlinked with and L_t be a ranked list of datasets recommended for d_t. Let G_{d_t} be the gold standard for d_t, i.e., the set of datasets that have linksets with d_t in the gold stan-

dard. A dataset d_j is *relevant* for d_t, in the context of G_{d_t}, iff there are linksets connecting d_j and d_t in G_{d_t}. We then define:

- $Prec@k\,(L_t)$, the *precision at position* k of L_t, is the number of relevant datasets in L_t until position k.
- $AveP\,(L_t)$ is the average precision at position k of L_t, defined as: $AveP(L_t) = \sum_k Prec@k\,(L_t)\,/\,|G_{d_t}|$.

Recall from Section 3, that the ranked list L_t of datasets recommended for d_t can be generated using two strategies: (i) *cluster-based*, that is, based on the datasets available within a cluster; and (ii) *profiling-based*, that is, based on all datasets available. The *overall mean average precision (MAP)* for these strategies is then defined in slightly different ways.

For the profiling-based strategy, we define:

- The *overall MAP* is the average of $AveP\,(L_t)$ taken over the set of all datasets d_t

and, for the cluster-based strategy, we define:

- $MAP(C_i)$, the *Mean Average Precision* for C_i is the average of $AveP\,(L_t)$ taken over the set of all datasets d_t in C_i
- The *overall MAP* is the average of $MAP(C_i)$ taken over the set of all clusters C_i.

5.2. Results

We ran experiments considering the two recommendation strategies. For the cluster-based strategy, Figure 3 shows the overall MAP as a function of the number of clusters (in increments of 1). It indicates that the maximum value of overall MAP is 18.44%, when the number of clusters was equal to 11.

Figure 3 indicates that the maximum MAP is obtained with 11 clusters, whereas the number of categories used to classify datasets in the LOD diagram is only 8. But if we construct just 8 clusters, our recommendation approach reaches an overall MAP of 16.0%, which is sub-optimal. That is, the LOD diagram is not a good starting point for our recommendation strategy. With only 8 clusters, many more non-relevant datasets end up being recommended, which decreases the overall MAP, as compared with the scenario that considers 11 clusters.

For the profiling-based strategy, Figure 4 presents the percentage of the total number of datasets for which the technique achieved a given MAP as a function of overall MAP intervals. It shows that this strategy: reached an overall MAP of 11-20% for 25% of the datasets; achieved an overall MAP higher than 20% for more than 37% of the datasets, reaching, in some cases, MAP values higher than 80%.

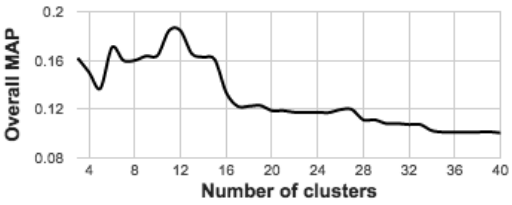

Fig. 3. Strategy 1: Overall Mean Average Precision *vs.* number of clusters.

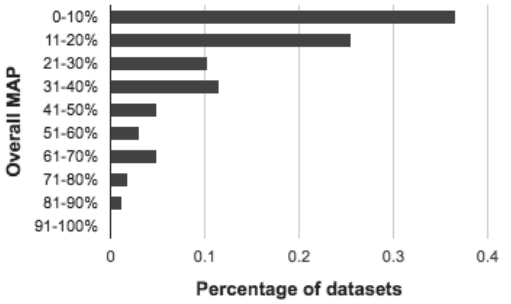

Fig. 4. Strategy 2: Percentage of datasets *vs.* Overall Mean Average Precision.

5.3. Discussion

The results reported in Section 5.2 should be assessed under several provisos, related to limitations of both the experiments and of the techniques the tool implements.

5.3.1. Profiling and interlinking issues

False positives. *Situation 1:* We first observe that we adopted as gold standard the LOD datasets represented in the Mannheim catalog that could be profiled. Furthermore, we considered that a source dataset d_j is correctly linked to a target dataset d_k iff the Mannheim entry for d_j contains a linkset for d_k. This may cause distortions on the results reported since DRX might correctly recommend a dataset d_m to be interlinked with d_j and no linkset is reported connecting d_j and d_m. This limitation has already been remarked in [4]. Therefore, d_m would be incorrectly considered as a false positive in the experiment, a situation that can only be uncovered by actually trying to interlink d_j and d_m, an expensive experiment that should be undertaken with care.

Situation 2: This scenario cannot be strictly considered as leading to a false positive, but the arguments are a continuation of the previous discussion. Consider again d_j and d_m, that is, DRX included d_m in the recommendation list for d_j. It might be the case that the user may try to interlink d_j and d_m without success because, although similar with respect to their profiles, these datasets contain different sets of resources that cannot be interlinked. The current implementation of the tool cannot automatically detect this situation, but its interface supports browsing the contents of a dataset so that the user may judge if the recommendation is likely to lead to interlinking d_j and d_m.

False negatives. *Situation 1:* The recommendation step depends heavily on the dataset profiling technique adopted. Let d_j be a dataset which is specific for a given area and, hence, whose profile has peaks for certain categories (such as agriculture, say). However specific d_j might be, it is common practice to interlink a dataset with a dataset d_m that contains generic data (such as the DBpedia). Since d_m is generic, its profile will tend to have high scores for most of the top categories. Hence, DRX will probably not rank d_m high in the recommendation list for d_j (which we assumed is not a generic dataset). Therefore, d_m would be considered a false negative in the experiment. This situation can be overcome by treating generic datasets (with high scores for most of the top categories) separately.

Situation 2: Conversely, d_j might be a dataset which is more generic than d_m. Hence, DRX will probably not rank d_m high in the recommendation list for d_j. Therefore, d_m would be considered a false negative in the experiment.

5.3.2. Third party tools issues

As part of the dataset profiling step, we use the Wikipedia Miner (WM) to extract entities from a given text literal. WM achieves good precision ($\approx 73\%$) and recall ($\approx 75\%$) rates for entity recognition as reported in [23]. Also, in a previous work [5], we conducted a similar experiment where we showed that misrecognized entities do not significantly impact in the resulting profile. In [5] we also show that the impact is related to the number of resources (sample size) extracted from a dataset to generate the profile and in this case we may need to find a trade-off between accuracy and scalability. Currently, DRX considers 10% of the resources from a dataset and obtained good accuracy and scalability.

Another reason for the low impact of the misrecognized entities in the resulting profile is that the process does not solely consider the entities but their parent categories. The entity categories are grouped and only the categories over a given threshold are kept in the process. So, the misrecognized entities/categories will mostly probably not survive.

An example could be given by the following enriched text: "Pelé began playing for Santos at 15 and the Brazil national football team at 16. He won three FIFA World Cups." Santos can be a city or a Brazilian football team. In this example, we may expect that Santos can be recognized as a resource of the category Sports. However, suppose that Santos was misrecognized as a city resource. Whilst the others well-recognized resources will contribute to the same categories (i.e. Sports), Santos, as a city, is expected to contribute to other categories not related. So, after running our process, the low contribution will not be considered, eliminating the contribution given by the misrecognized entity and hence its associated categories.

5.3.3. Examples of good and bad dataset profiles

A key factor to generate good dataset profiles is the availability of text literals. However, most datasets available in catalogs merely offer a small description with no associated data. So, a small sample of text literals may result in a not descriptive profile, increasing the probability of obtaining false positive recommendations.

Consider, for example, the *rkb-explorer-newcastle* dataset which obtained a high MAP value (70%). The *rkb-explorer-newcastle* entry in the Mannheim Catalog is richly described by five different resource types: (1) XML Sitemap; (2) VoID File; (3) a resource Example; (4) an RDF file for download; and (5) an SPARQL endpoint that provides direct access to the entire content of the dataset. With such amount of information available, the creation of the dataset profile tends to be more accurate.

Inspecting the profile generated for the *rkb-explorer-newcastle* dataset, we verified that the recognized entities were strictly related to the information the dataset provides. A sample of the recognized entities is: Programmer, Computer Software, Functional Programming, Neural Networks.

With enough resources available, DRX was able to generate a proper profile (fingerprint), selecting the best Wikipedia top-level categories to represent the dataset. In this case, DRX was able to generate a fingerprint with peaks at: Science, Technology and Mathematics, which are fully related to the dataset.

Unfortunately, as mentioned in [17], only a few datasets have SPARQL endpoints available and rich descriptions. An example of a dataset with low MAP is the *Statusnet-doomicile-de*. As mentioned in our previous work [5], and also in [17], this dataset lacks information and resources. For example, the only file available for *Statusnet-doomicile-de* is an example resource. No SPARQL endpoint is available. Unfortunately, with so few resources available, it is impossible to properly cover the content of a dataset. Moreover, without an SPARQL endpoint, DRX is unable to inspect its content. Hence, DRX does not generate a proper fingerprint resulting in a low MAP value.

Another issue found in datasets published in the Mannheim Catalog is that there are many datasets supposedly hacked. This is the case of: *HACKED BY SLAYERSHACKTEAM*, *admin*, *HacKeD By KingSkrupellos*, and many other datasets. The lack of quality, spam and curation lead us to low MAP values, not the method itself.

6. Conclusions and Future Work

We proposed a tool, called DRX, to assist data publishers in the process of dataset interlinking. DRX takes advantage of various methods including crawling, profiling, clustering and ranking modules to create ranked lists of datasets to be interlinked with a given dataset. The results obtained indicate that the proposed approach can indeed be used to facilitate the task of dataset interlinking in the LOD. They show that the profiling-based strategy achieves a better performance than the cluster-based strategy.

As for future work, we plan to thorough compare DRX with the TRT and TRTML tools, to further validate the results by actually running interlinking tools over selected pairs of datasets based on the recommendations DRX produced.

References

[1] A. A. M. Caraballo, B. P. Nunes, G. R. Lopes, L. A. P. Paes Leme, M. A. Casanova, and S. Dietze, *TRT-A Tripleset Recommendation Tool*, ISWC 2013 (Posters & Demos), pp. 105-108.

[2] A. A. M. Caraballo, N. M. Arruda Jr, B. P. Nunes, G. R. Lopes, and M. A. Casanova, *TRTML-A Tripleset Recommendation Tool Based on Supervised Learning Algorithms*, ESWC 2014 (Satellite Events), pp. 413-417.

[3] C. Bizer, T. Heath, and T. Berners-Lee. *Linked Data– The Story So Far.* Int. J. Semantic Web Inf. Syst., 5(3):1-22, 2009.

[4] M. Emaldi, O. Corcho, and D. López-de-Ipiña, *Detection of Related Semantic Datasets Based on Frequent Subgraph Mining Mikel*, IESD 2015 (in ISWC 2015).

[5] B. Fetahu, S. Dietze, B. P. Nunes, M. A. Casanova, D. Taibi and W. Nejdl, *A scalable approach for efficiently generating structured dataset topic profiles*, in The Semantic Web: Trends and Challenges, pp. 519-534, 2014.

[6] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, *WEKA data mining software: an update*, ACM SIGKDD explorations newsletter 11, no. 1 (2009): 10-18, 2009.

[7] T. Heath, and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space* (1st Edition), volume 1 of Synthesis Lectures on the Semantic Web: Theory and Technology, 2011.

[8] A. Jentzsch, R. Cyganiak, C. Bizer, *State of the lod cloud*, 2011.

[9] R. Kawase, P. Siehndel, B. P. Nunes, E. Herder and W. Nejdl, *Exploiting the wisdom of the crowds for characterizing and connecting heterogeneous resources*, 25th ACM Conf. on Hypertext and social media, pp. 56-65, 2014.

[10] S. Lalithsena, P. Hitzler, A. Sheth and Paril Jain, *Automatic domain identification for linked open data*, 2013 IEEE/WIC/ACM Int'l. Joint Conf. on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 1, pp. 205-212.

[11] L. A. P. P. Leme, G. R. Lopes, B. P. Nunes, M. A. Casanova, and S. Dietze, *Identifying candidate datasets for data interlinking*, in Web Engineering, pp. 354-366, 2013.

[12] G. R. Lopes, L. A. P. Paes Leme, B. P. Nunes, M. A. Casanova, and S. Dietze,*Recommending tripleset interlinking through a social network approach*, WISE 2013, pp. 149-161.

[13] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. *DBpedia spotlight: shedding light on the web of documents*, 7th International Conference on Semantic Systems, pp. 1-8, 2011.

[14] A. Nikolov, and M. d'Aquin, *Identifying relevant sources for data linking using a semantic web index*, in Linked Data on the Web, Workshop in WWW 2011, vol. 813.

[15] A. Ngomo, and S. Auer, *LIMES: a time-efficient approach for large-scale link discovery on the web of data*, IJCAI 2011, pp. 2312-2317.

[16] D. Pelleg and A. W. Moore, *X-means: Extending K-means with Efficient Estimation of the Number of Clusters*, ICML 2000, pp. 727-734.

[17] M. Schmachtenberg, C. Bizer, and H. Paulheim, *Adoption of the linked data best practices in different topical domains*, ISWC 2014, pp. 245-260.

[18] B., Sergey, and L. Page. *Reprint of: The anatomy of a large-scale hypertextual web search engine*, Computer networks 56, no. 18 (2012): 3825-3833.

[19] Volz, Julius and Bizer, Christian and Gaedke, Martin and Kobilarov, Georgi,*Silk-A Link Discovery Framework for the Web of Data*, LDOW 2009, 538.

[20] G. Tummarello, R. Cyganiak, M. Catasta, S. Danielczyk, R. Delbru, R, S. Decker, *Sig.ma: Live views on the Web of Data* , Web Semantics: Science, Services and Agents on the World Wide Web. 2010 Nov 30;8(4):355-64.

[21] *http://drx.inf.puc-rio.br/*

[22] *http://ckan.org/*

[23] Ellefi M.B., Bellahsene, Z., Scharffe, F., Todorov, K.. Towards semantic dataset profiling. Profiles Workshop 2014, ESWC 2014.