Linked Data Profiling

Identifying the Domain of Datasets Based on Data Content and Metadata

Andrejs Abele
«Supervised by Paul Buitelaar, John McCrae, Georgeta Bordea»
Insight Centre for Data Analytics, National University of Ireland, Galway
IDA Business Park,Lower Dangan
Galway, Ireland
andrejs.abele@insight-centre.org

ABSTRACT

Since the beginning of the Linked Open Data initiative, the number of published open datasets has gradually increased, but the datasets often do not contain description about content such as the dataset domain (e.g., medicine, cancer), when this information is available, it is usually coarse-grained e.g. `organic-edunet` contains the metadata about a collection of learning objects exposed through the *Organic.Edunet* portal, but it is classi ed as *Life science*. In this work we propose approaches that will provide a detailed description of existing datasets as well as linking assistance when publishing new datasets by generating detailed descriptions of the publishers dataset.

Keywords

Linked data pro ling, Linked data, domain identi cation

1. INTRODUCTION

Data pro ling is the process of creating descriptive information and collecting statistics about the dataset. It is the most important activity when facing an unfamiliar dataset [17] and can help to assess the importance of the dataset as a whole, nd out whether the dataset or part of the dataset can be easily reused, improve the user ability to query or search the dataset, and detect irregularities for improving data quality.

Moreover in the linked data paradigm, the datasets are connected to each other in a manner similar to how web pages are connected on the World Wide Web [3]. Data pro ling also provides information of these connections between datasets and this is what creates the Web of Data which allows to connect and reuse existing data instead of replicating the data.

Linked data pro ling consists of creating quantitative information of these datasets and of creating qualitative descriptions about the topics covered by the datasets. In our work, we focus on the qualitative description.

WWW'16 Companion, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
http://dx.doi.org/10.1145/2872518.2888603.

1.1 Motivation

Linked Open Data (LOD) has gained signi cant visibility and adoption since its inception. Starting with 12 datasets in 2007, currently it consists of at least 9,960 datasets[1]. The rapid growth in the number of LOD datasets reveals the interest of data publishers in publishing their data as structured data on the data cloud and this trend is likely to continue. Furthermore, the range of domains and topics covered by these datasets has also increased. When adding a new dataset to the LOD cloud, links should be identi ed to as many other relevant LOD datasets as possible, which calls for tools that support linked data search and discovery.

1.2 The Problem Statement

The number of open datasets is growing, but often they are not linked. From 10,632 datasets in DataHub[2] only 1,027[3] claim that they are connected and contain live links to other datasets. This highlights the problem, that publishers who want to publish their datasets do not have enough knowledge of existing datasets and do not provide metadata that correctly represents the content of their own datasets.

To solve this problem we propose two approaches, which are based on the same methodology. We will describe this methodology in detail in section 3.3 and 3.4.

In the rst approach we analyse existing datasets and provide a detailed description of these resources. In the second approach we provide metadata about the dataset that a publisher wants to publish, and suggestions for the existing datasets that the dataset could be linked to.

2. STATE OF THE ART

In the early days of linked data [4] the main focus of the community was on publishing data and nding good practices, but since the amount of the datasets was growing fast, so did the necessity for statistics and summaries about the existing datasets. RDFStats [16] and Semantic sitemaps [7] were one of the rst to deal with RDF data statistics and summaries. Based on their work there has been recently an explosion of tools for analysing linked data datasets.

2.1 Analytics systems

Tools like ExpLOD [14], LODStats [2], ProLOD++ [1], LODOP [11] and Aether [18] compute statistical informa-

[1] As of 22.11.15 based on statistics provided by LODstats
[2] `https://datahub.io/`
[3] As of 22.11.15 based on connected live links in DataHub

tion which is vital to the applications dealing with query optimization and answering, data cleansing, schema induction and data mining [13, 15].

There are also systems e.g., Project Open Data Dashboard [4] that tracks and measures how US government web sites implement the Open Data Principles to understand the progress and current status of their public data listings.

Bohm [5] presents an approach that exploits only the structure of an entity-relationship graph to address the problem of mining latent topics from graph-structured data.

One of the most recent tools for LOD analytics is LODVader[5], a system that provides similar statistics to what we want to provide e.g., number of triples, frequencies and distributions of distinct subjects, predicates, and objects, and a list of used vocabularies. LODVader crawls a dataset and extracts links using Bloom filters. They also calculate similarity between datasets by using `owl:Class`, `rdf:type` and predicates and visualise the results in an interactive diagram, however contrary to our approach, they do not classify datasets based on topics or domains.

2.2 Topical profiling

Most closely related to our research is topical profiling which focuses on the content-wise analysis at the instances and ontological levels. Lalithsena [15] performs automatic domain identification on the linked data by retrieving entity labels and labels of their classes, then they send the labels (of entity and classes) to Freebase[6] API and retrieve Freebase type and domain information. The results are merged to create a category hierarchy where only hierarchies with the most common root are kept. In the next step the most frequent category from all hierarchies is selected as the domain.

Similar work is done by Fetahu [9, 10] who describes a system that samples datasets, using DBpediaSpotlight[7] to identify entities and categories, followed by category filtering and ranking where the top ranked categories are considered topics.

Our approach can be considered a hybrid between these two approaches, as it differs from Lalithsena in that we use DBpedia[8] instead of Freebase and where as Lalithsena approach relies on class labels to identify the entities our approach does not require this data. Contrary to Fetahu approach we process the whole dataset, not just a sample, because our goal is to provide a description with different levels of granularity. Like Fetahu, we use DBpediaSpotlight, but instead of just retrieving the categories, we also retrieve the type of the entity as this provides us additional information and helps in identifying the relevant categories.

Figure 1: Domain identifications system

[4]http://labs.data.gov/dashboard/offices

[5]http://lodvader.aksw.org/#/home

[6]https://developers.google.com/freebase/?hl=en

[7]https://dbpedia-spotlight.github.io/demo/

[8]http://wiki.dbpedia.org/

3. METHODOLOGY

We propose two approaches, which are based on the same technology. In the first approach, we analyse the existing datasets and provide a detailed description of the datasets. In the second approach, we provide metadata about these datasets that a publisher wants to publish, and suggestions for possible datasets that the dataset could be linked to.

As both approaches are based on the same underlying technology, we will briefly describe the differences in the approaches, and focus on the common technology.

3.1 Approach for LOD resource discovery

In this approach we crawl the existing LOD portals (Publicdata[9], DataHub[2], Amsterdam Open Data[10], Europa[11]) to collect the same type of statistics as Lodstats [2] and ProLOD++ [1]. We do not use any of the existing systems because when we are extracting string literals from the dataset, we need to process the whole dataset and during this process we can easily collect the statistics that are relevant for us. If we would analyse the dataset using other systems then it would be an inefficient use of resources because we would have to resubmit the whole dataset to another system and depending on the size it can be time and resource consuming. As output we provide an interactive LOD cloud diagram with added metadata (e.g., number of triples, connections to other datasets, used vocabularies, domain of the dataset). This metadata is generated using approaches described later in this paper.

3.2 Approach for dataset publication assisting

Our approach provides recommendations for the publisher about what metadata to provide with the dataset and recommend related datasets to which the publisher's dataset could be linked based on domain, topics and LOD cloud diagram. This system processes RDF dumps and will provide a web interface where the publisher can inspect the metadata that we generate and modify it to precisely represent their dataset. Afterwards we generate an RDF metadata file using the VoID vocabulary, which the publisher can add to the dataset or provide it as a separate metadata file.

3.3 Statistics gathering

The statistics gathering method is shared by the two approaches. We provide statistics about frequencies and distributions of distinct subjects, predicates, and objects, a list of the different data types used for literals, and a list of used vocabularies. We use state of the art methods, similar to those used by existing systems - RDFStats [16], LODStats [2], ProLOD++ [1], LODOP [11], Aether [18].

3.4 Domain identification

To identify the domain of the dataset we analyse string literals from a given dataset and link them to DBpedia. We are using DBpedia categories because they cover large domains and we have not encountered situation when a dataset describes a domain which is not present in DBpedia categories. Our approach can be split in multiple subtasks i.e., (i) computing TF-IDF on extracted string literals, where we assume that each string literal is a separate document, then

[9]http://publicdata.eu/

[10]http://data.amsterdamopendata.nl/

[11]http://open-data.europa.eu/en/data/

we rank the terms based on their TF-IDF score and select the top results. As we identify topics in the dataset by linking them to the DBpedia concepts, we need contextual data, so in the next step we (ii) lter string literals containing the top terms. After collecting all the string literals that contain any of the top terms we identi ed, these literals are sent to (iii) DBpediaSpotlight [8]. We use the DBpediaSpotlight system because when it has recognised an entity, it provides a link to a DBpedia concept, and if string literal entity that links to the same DBpedia concept is discovered, we can be certain that it is the same entity. This is important because -rs tn aoi0s-not enough to reach a level of accuracy that would be sufficient for a fully automated approach.

We can assume with reasonable certainty that the problem lies in the dataset because in our internal experiments we ran combined 476,576 permutations of configurations for *C-SVC [6]* and the *tree classifier j84 [20]*, and we could not improve the results over those reported in this paper.

From the second experiment we can conclude that the tags provided by the creators of the datasets provide the most accurate classification, but they also require the most effort from the dataset creators and publishers. This means that this approach can be successfully used to classify existing dataset that have human-annotated tags. The disadvantage of this approach is that an annotated training set is required and if the new dataset contains tags that were not present in the training set then the classifier will not be able to identify the domain.

Our approach does not require any training data and can be run on any RDF dataset that contains literals. We have run our approach on the datasets contained in the LOD cloud diagram and we evaluated on the domain labels in the diagram. However the initial experiments have been inconclusive so far because the existing domain classification in the LOD cloud does not cover the whole range of information that actually is represented in the LOD cloud.

5.1 Evaluation plan

As mentioned before, there has been recent work that tries to automatically identify the domain of a dataset [15, 9]. Annotations used by different systems are incompatible and automated mapping would not be accurate. For this reason we will run these two existing approaches and our approach on the same datasets, then we will select top results from each approach and ask human evaluators to determine which system provided fitting domain description.

5.1.1 LOD resource discovery system evaluation

To evaluate the LOD resource discovery system and how helpful our provided description of the existing LOD cloud is, we will monitor user activity on our system (e.g. how long a user stays on our diagram, how many different resources are they selecting) and we will ask returning users if our previous recommendation was useful. This type of evaluation is long-term and depends on the amount of users that are using it. As there is an obvious possibility that we will not be able to collect enough user data to evaluate our system, we will contact the creators of the datasets that we will have processed and ask if they agree with our description of their datasets.

5.1.2 Recommender system evaluation

To evaluate our recommender system, we will collect statistics about how many of our recommendations the user followed and we also will collect user feedback about our recommendations. To increase the amount of data publishers using our system we will collaborate with linked data publishing portals e.g., DataHub[2] and ask them to recommend our system as metadata generating tool.

5.1.3 Individual component evaluation

Apart from evaluating the whole system as one, we will evaluate each of the components separately by comparing them to the alternative approaches based on the premises that the best solutions for the individual components will provide the best results for the whole system.

The component where we identify the most relevant terms using the TF-IDF algorithm, we will compare to the alternative approaches that are used in text summarization to identify the most important sentences e.g., TextRank, LexRank,SumBasic. Inouye [12] compares these algorithms on the Twitter datasets and considering that often the linked data datasets contain short textual descriptions just like tweets, we believe that this algorithms could be good alternatives for the TF-IDF algorithm we are using. To evaluate this step we will select popular LOD datasets e.g., BBCMusic, Foodalista, Medicare where for the human annotators it will be easy to identify if the selected terms are describing the datasets.

For evaluating the entity recognition component with alternative solutions e.g., Babelfy[19] we will use the dataset from the #Microposts2015 NEEL challenge[15]. We selected this dataset as it comprises tweets extracted from a collection of over 18 million tweets. They include event-annotated tweets. As mentioned before, limitations on the length of the tweets makes them similar to the linked data literals as often they are short.

To evaluate the category identification step we will use the same approach as for the TF-IDF evaluation, we will select popular LOD datasets and ask human annotators to identify if the categories that we have selected fit to the dataset.

5.2 Future work

As described earlier, we are using the DBpedia category structure to identify domains, but this structure is very large (960,039 nodes and 4,553,783 links), so we will create a simplified category structure that doesn't contain named entities with the assumption that it will provide faster and better domain recognition.

At this stage our approach does not take into account the domains of related datasets, but we are planning to extend it so that using information about other linked datasets could help to identify the domain.

As the size of the LOD cloud is growing, it becomes harder to visualise it, we will investigate alternative visualisation solutions, other then currently used in visualising the LOD diagram, e.g. hierarchical graph representation or chord diagram.

As noted by other researchers [15], the current LOD cloud domain classification in many cases does not make a lot of sense, for this reason we are planning to perform a study to identify what domain classifications are used by actual LOD applications in academia and industry.

We have some simple crawlers that can gather information about the datasets and retrieve data dumps, but it requires human supervision and interaction. Therefore we are planning to create a fault tolerant and more generic LOD crawler that could work autonomously.

6. ACKNOWLEDGEMENTS

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 (Insight) and MixedEmotions (H2020-644632).

[15]https://www.dropbox.com/s/8daewrvd1bz864g/microposts2015_neel_challenge_cfp.txt?dl=0

7. REFERENCES

[1] Z. Abedjan, T. Gruetze, A. Jentzsch, and F. Naumann. Pro ling and mining rdf data with prolod++. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 1198{1201. IEEE, 2014.

[2] S. Auer, J. Demter, M. Martin, and J. Lehmann. Lodstats{an extensible framework for high-performance dataset analytics. In *Knowledge Engineering and Knowledge Management*, pages 353{362. Springer, 2012.

[3] C. Bizer, T. Heath, and T. Berners-Lee. Linked data-the story so far. *International journal on semantic web and information systems 5.3*, pages 1{22, 2009.

[4] C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee. Linked data on the web (ldow2008). In *Proceedings of the 17th international conference on World Wide Web*, pages 1265{1266. ACM, 2008.

[5] C. Böhm, G. Kasneci, and F. Naumann. Latent topics in graph-structured data. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2663{2666. ACM, 2012.

[6] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[7] R. Cyganiak, H. Stenzhorn, R. Delbru, S. Decker, and G. Tummarello. Semantic sitemaps: e cient and exible access to datasets on the semantic web. In *Proceedings of the 5th European semantic web conference on The semantic web: research and applications*, pages 690{704. Springer-Verlag, 2008.

[8] J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes. Improving e ciency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*, 2013.

[9] B. Fetahu, S. Dietze, B. P. Nunes, M. A. Casanova, D. Taibi, and W. Nejdl. A scalable approach for e ciently generating structured dataset topic pro les. In *The Semantic Web: Trends and Challenges*, pages 519{534. Springer, 2014.

[10] B. Fetahu, S. Dietze, B. Pereira Nunes, M. Antonio Casanova, D. Taibi, and W. Nejdl. What's all the data about?: creating structured pro les of linked data on the web. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 261{262. International World Wide Web Conferences Steering Committee, 2014.

[11] B. Forchhammer, A. Jentzsch, and F. Naumann. Lodop-multi-query optimization for linked data pro ling queries. In *International Workshop on Dataset PROFIling and fEderated Search for Linked Data (PROFILES), Heraklion, Greece*, 2014.

[12] D. Inouye and J. K. Kalita. Comparing twitter summarization algorithms for multiple post summaries. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 298{306. IEEE, 2011.

[13] A. Jentzsch. Pro ling the web of data. *Proceedings of the 8th Ph. D. retreat of the HPI research school on service-oriented systems engineering*, page 101, 2014.

[14] S. Khatchadourian and M. Consens. Explod: Summary-based exploration of interlinking and rdf usage in the linked open data cloud. *The Semantic Web: Research and Applications*, pages 272{287, 2010.

[15] S. Lalithsena, P. Hitzler, A. Sheth, and P. Jain. Automatic domain identi cation for linked open data. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, volume 1, pages 205{212. IEEE, 2013.

[16] A. Langegger and W. Wö . Rdfstats-an extensible rdf statistics generator and library. In *Database and Expert Systems Application, 2009. DEXA'09. 20th International Workshop on*, pages 79{83. IEEE, 2009.

[17] H. Li. Data pro ling for semantic web data. In *Web Information Systems and Mining*, pages 472{479. Springer, 2012.

[18] E. Mäkelä. Aether{generating and viewing extended void statistical descriptions of rdf datasets. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 429{433. Springer, 2014.

[19] A. Moro, A. Raganato, and R. Navigli. Entity linking meets word sense disambiguation: a uni ed approach. *Transactions of the Association for Computational Linguistics*, 2:231{244, 2014.

[20] R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.