Enhancing Open Data to Linked Open Data with ODMiner

Francesco Poggi[1], Andrea Giovanni Nuzzolese[2], Gabriele Cigna[1]

[1] DASPLab, Department of Computer Science and Engineering, University of Bologna, Italy
[2] STLab, Institute of Cognitive Science and Technologies, National Research Council, Italy

Abstract. In this paper we introduce ODMiner, an automatic tool that enhances open datasets provided in heterogenous structured formats (e.g. JSON, CSV, XML, etc.) to Linked Open Data. ODMiner mines OD by recognising well known data types and formats (e.g., dates, emails, currencies, etc.) and by exploiting well known open linked datasets and vocabularies (e.g. DBpedia, WordNet, etc.) in order to extract named entities and relations between the open dataset elements. ODMiner is designed as modular and extensible software architecture and its process can be customised in order to address specific needs of final data representation and modelling. Finally, an evaluation of ODMiner with heterogenous multi-language OD datasets is provided in order to give evidence of its practical effectiveness.

1 Introduction

Over the last years, the volume of Open Data (OD) published on the Web has grown hugely, raising the joining interest of public institutions, private companies and citizens. Unfortunately, OD consumers have still to face a variety of challenges when trying to access, understand or use OD in order to develop innovative services for solving real world problems on top of them. In particular, a manual effort is still required for analysing and mining open datasets. For example, in most of the cases it is complex to understand their content and context, to extract information and transform it into an machine-understandable format, to filter the elements of interest, etc. This limits the exploitation of OD, posing serious threats to their real reuse. In this paper we introduce ODMiner, an automatic tool that enhances open datasets provided in heterogenous structured formats (e.g. JSON, CSV, XML, etc.) to Linked Open Data. ODMiner mines OD by recognising well known data types and formats (e.g., dates, emails, currencies, etc.) and by exploiting well known open linked datasets and vocabularies (e.g. DBpedia, WordNet, etc.) in order to extract named entities and relations between the open dataset elements.

We mention only two of the possible applications of these results. First of all, the linking of OD datasets to popular open linked datasets and vocabularies can be used to automatize the process of OD triplification and semantic enrichment, and their consequent use in Semantic Web-based applications. Another useful application we envision is using this information for automatic OD visualization. By leveraging rich OD description we could be able to automatically generate, for instance, visual analytics tools and panels that can be used by users to explore, analyse and make sense of OD datasets.

ODMiner is designed as modular and extensible software architecture and its process can be customised in order to address specific needs of final data representation and modelling. Finally, an evaluation of ODMiner with heterogenous multi-language OD datasets is provided in order to give evidence of its practical effectiveness. The structure of the paper is the following: (i) Section 2 reports the related work; (ii) Section 3 describe our system, i.e. ODMiner; (iii) Section 4 describes the evaluation we carried on to assess the effectiveness of ODMiner in the task of data type recognition; (iv) finally, Section 5 provides the conclusions and possible future work.

2 Related works

In recent years a lot of research has been carried on to populate the Semantic Web with Linked Open Data. Most of the approaches rely on fixed or mostly limited customisable solutions to convert structured sources to Linked Open Data. Worth mentioning tools are the D2R Server [1] and Krextor [5]. D2R is a tool for publishing relational databases on the Semantic Web. It enables RDF and HTML browsers to navigate the content of a database, and allows other applications to query a database through SPARQL. D2R Server uses the D2RQ Mapping Language to map the content of a relational database to RDF. A D2RQ mapping rule specifies how to assign URIs to resources, and which properties are used to describe them. Krextor is an extensible XSLT-based framework for extracting RDF from XML, supporting multiple input languages as well as multiple output RDF notations.

On the other hand, mining meaningful information from structured as well as semi-structured data sources (e.g. CSV, XML or JSON) remains a challenging task. In addition to OD quality [2], one of the main problem that limits OD reuse is a lack of tools that allow a nearly one-click process to mine relevant information from OD format in order to enhance them to Linked Open Data effectively. As a matter of fact most of the state of the art systems rely on machine learning platforms such as Weka [3] or RapidMiner[3]. A relevant example is [7] that introduces the RapidMiner Linked Open Data extension. This extension hooks into the powerful data mining platform RapidMiner, and offers operators for accessing Linked Open Data

[3] https://rapidminer.com/

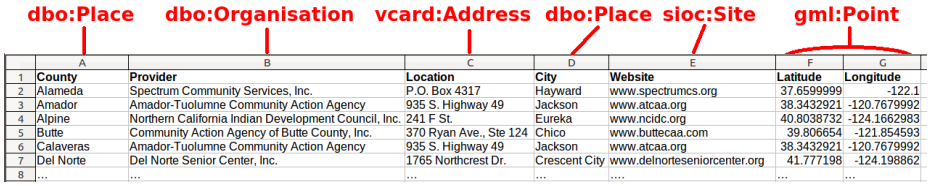
in RapidMiner, allowing for using it in sophisticated data analysis workflows without the need to know SPARQL or RDF.

Another approach is using a tool such as Wrangler [4] or OpenRefine[8] to mine, clean and transform semi-structured (e.g. tabular) OD datasets to Linked Data. The goal of these tools is to facilitate users in the task of analysing OD and enhancing them to LOD. Though very effective, the conversion provided by such tools is not completely automatic and still requires the human intervention.

To the best of our knowledge ODMiner is a novel and extensible solution to support the generation of Linked Open Data. It provides an HTTP REST interface that allows to automatically mining the data types from Open Data available in heterogeneous formats such as XML, JSON and CSV in a nearly one-click process.

3 ODMiner

ODMiner[4] is a web-based tool aimed at analysing and making sense of heterogeneous OD datasets. In particular, ODMiner takes as input the URL of an OD dataset (JSON, XML, CSV and TSV are the data formats currently supported), analyses its content, and infers which kind of data each field of the dataset is an instance of. For example, given the tabular dataset in Figure 1, ODMiner map each column to a meaningful class (the red text at the top of the table) of a well-known ontology.



	A	B	C	D	E	F	G
1	**County**	**Provider**	**Location**	**City**	**Website**	**Latitude**	**Longitude**
2	Alameda	Spectrum Community Services, Inc.	P.O. Box 4317	Hayward	www.spectrumcs.org	37.6599999	-122.1
3	Amador	Amador-Tuolumne Community Action Agency	935 S. Highway 49	Jackson	www.atcaa.org	38.3432921	-120.7679992
4	Alpine	Northern California Indian Development Council, Inc.	241 F St.	Eureka	www.ncidc.org	40.8038732	-124.1662983
5	Butte	Community Action Agency of Butte County, Inc.	370 Ryan Ave., Ste 124	Chico	www.buttecaa.com	39.806654	-121.854593
6	Calaveras	Amador-Tuolumne Community Action Agency	935 S. Highway 49	Jackson	www.atcaa.org	38.3432921	-120.7679992
7	Del Norte	Del Norte Senior Center, Inc.	1765 Northcrest Dr.	Crescent City	www.delnorteseniorcenter.org	41.777198	-124.198862
8	...	...	...	...	....	...	...

Fig. 1. An example of the output of ODMiner produced by the analysis of a tabular dataset.

ODMiner has a modular and extensible architecture. It provides some software modules, each of which is responsible of recognising if the data fields are instances of a specific class. The list of some of the basic classes that are identified by ODMiner is shown in Table 1, and has been derived by the analysis of the set of italian datasets described in Table 2. Other software modules can be implemented and loaded into the ODMiner engine, extending its analysis capabilities (e.g. extending the set of managed ontologies to which input data can be mapped).

ODMiner main engine is responsible of coordinating the analyses performed by its software modules. The overall process can be summarised as

[4] http://eelst.cs.unibo.it:8099/detector/api/url

Data type	Class URI
Email Address	vcard:Email
Gender	vcard:Gender
Location	dbo:Place
Gender	vcard:Gender
Location	dbo:Place
Person	dbo:Person
Number	xsd:Decimal
Geographic coordinates	geo:Point
Date	xsd:date, xsd:dateTime, etc.

Table 1. Some relevant classes identified by ODMiner built-in modules.

follows. First, ODMiner splits the dataset fields into homogeneous partitions. For example, the fields of tabular datasets are grouped by columns. Then, each module analyses the fields of each group separately. The kind of analysis differs from module to module: for example, the module responsible of recognising dates performs a simple pattern matching test, while a more sophisticated one such the place analyser computes Levenshtein distance [6] to estimate the similarity of data with DBpedia labels, etc. Finally, the main engine uses confidence thresholds to guarantee the statistical relevance of each analysis and, consequently, the accuracy of the results. The principle at the base of ODMiner logic is to leverage the homogeneity of structured or semi-structured datasets (e.g. of all the elements of a column in a tabular dataset) to infer meaningful information about their content.

ODMiner also provides users a mean to configure the analysis modules to define custom data analysers. For example, a user may be interested in recognising some specific types of time-related data (e.g. dates, times, durations, intervals, etc.). To do so, the user has to invoke ODMiner providing a configuration file[5] defining that he/she wants include an analysis based on a regular expression search, together with the regular expression to test and the ontological class that should be associated to the matching data.

Another configurable ODMiner module allows to identify data values that are instances of specific DBpedia classes. The basic idea is that ODMiner compares each field of the dataset (e.g. the cells of a column in a table) to the labels associated to DBpedia entities. When a match is found, ODMiner collects all the classes those entities are instances of. Each match is considered by ODMiner as a vote for that class. At the end of the analysis, all the classes that have not obtained a minimum number of votes (i.e. a config-

[5] The description of the format of the configuration, together with all the other information needed to download, compile and run ODMiner, is available at the address https://bitbucket.org/Cigna/semanticdetector

urable quorum threshold set by the user) are discarded. Among the remaining, the majority principle is applied, and the class with the higher number of matches is chosen. The DBpedia classes users are interested in can be specified by users, together with the number of records to analyse and the confidence threshold (i.e. the quorum). As discussed in Section 4, this principle has proven to be effective for the datasets and the classes taken into account in our evaluation.

4 Evaluation

We designed an experiment to assess the capability of ODMiner to enhance Open Datasets by effectively recognising the correct types associated with data. We randomly selected 16 datasets from Italian Open Data archives[6] and from some US data portals. Table 2 reports the full list of datasets used for the evaluation of ODMiner. Each row of the table identifies an open dataset and specifies the topic of the dataset and the URL that can be used to retrieve the dataset. The datasets were selected according to the following rationale: (i) homogeneous distribution of heterogeneous formats (e.g., CSV, XML, etc.); (ii) multilinguality of datasets, i.e. we did not focus on a single language (e.g. Italian); (iii) not negligible size of source data; (iv) heterogeneity of topics.

Topic	**Source URL**	Size (KBs)	Format
Doctors	http://gabriele.cigna.web.cs.unibo.it/dati/dottori.xml	70	XML
Nursery schools	http://gabriele.cigna.web.cs.unibo.it/dati/asili.csv	24	CSV
Dr. Who universe of doctors, actors, etc.	http://gabriele.cigna.web.cs.unibo.it/dati/drwho.csv	24	CSV
Drugs	http://gabriele.cigna.web.cs.unibo.it/dati/farmaci.xml	327	XML
Pharmacies	http://gabriele.cigna.web.cs.unibo.it/dati/farmacie.xml	127	XML
Employees of Emilia Romagna (italian administrative region)	http://gabriele.cigna.web.cs.unibo.it/dati/people.xml	45	XML
Plants	http://gabriele.cigna.web.cs.unibo.it/dati/plants.xml	9	XML
Music	http://gabriele.cigna.web.cs.unibo.it/dati/songs.xml	32	XML
Musical bands of Rome	http://85.18.173.117/opendata/ElencoBandeMusicali.csv	12	CSV
Libraries of Rome	http://85.18.173.117/mappe/Biblioteche.csv	20	CSV
Absestos licensed professionals of Illinois	https://data.illinois.gov/api/views/2gzr-9awy/rows.csv	230	CSV
Absestos licensed contractors of Illinois	https://data.illinois.gov/api/views/5vh3-wnad/rows.csv	18	CSV
Child health plus program enrollment of New York State	https://health.data.ny.gov/api/views/izdx-gtc9/rows.csv	8041	CSV
Contact information of the Dept. of Rehabilitation office of California	https://chhs.data.ca.gov/api/views/exxu-vffk/rows.csv	8	CSV
Service providers of California	https://chhs.data.ca.gov/api/views/jwfv-ersg/rows.csv	444	CSV
Public health services of the City of Chicago	https://data.cityofchicago.org/api/views/cjg8-dbka/rows.csv	19	CSV

Table 2. Topics, source URLs, sizes and formats of the open datasets used for the evaluation of ODMiner.

We constructed a ground truth by manually annotating the correct data type for each field (e.g. a column of a CSV table) of the datasets presented in Table 2. The ground truth is available on-line as CSV for consultation[7]. Hence, we used ODMiner to mine data types against the same datasets in order to compare the output of ODMiner with respect to the ground truth in

[6] http://www.dati.gov.it/ and http://85.18.173.117/

[7] http://eelst.cs.unibo.it/ld4ie/goldenstandard

terms of precision, recall and F-measure. We set up ODMiner with the configuration provided in Table 3. This configuration was provided to ODMiner as a JSON file[8] and allowed us to specify the confidence threshold and the matching criteria for data type recognition. The confidence thresholds reported in Table 3 are expressed in a range between 0 and 1, and have been selected through an empirical analysis, trying to find a good trade-off between the accuracy of the results and the analysis duration. The meaning associated with matching criteria is the following: (i) RegEx is a regular expression that is used to perform the matching; (ii) pattern matching identifies a set of cases or patterns from a list that are used to test whether a set of data matches those cases; (iii) Exact and Contain-based matches on DBpedia allows ODMiner to perform the classic "exact match" and "contains" operations on the values (either literals or URIs) coming from a Linked Dataset (e.g. DBpedia), which is used as background knowledge.

Data type	**Threshold**	**Matching criterion**
Email Address	0.1	RegEx
Tax Code	0.1	RegEx
Blood group	0.3	Matching wrt. to fixed set of elements
Gender	0.3	Matching wrt. to fixed set of elements
Location	0.1	Exact match on DBpedia labels of places
Person	0.1	Contain-based match on DBpedia labels of people
Number	0.1	RegEx
Geographic coordinates	0.1	RegEx
Date time	0.1	RegEx

Table 3. Types, confidence thresholds and matching criteria used to configure ODMiner for the evaluation experiment.

Finally, we ran the experiment by configuring ODMiner to recognise data types by using only a small slice (i.e. 20%) of the whole dataset. The resources part of the slice were picked up randomly. Figure 2 shows the final results we recorded.

Although ODMiner requires a further and more detailed evaluation that should be performed on a larger set of OD datasets and a broader set of classes to recognise, the preliminary results of these tests are encouraging and show that ODMiner has a good accuracy in performing the linking task.

5 Conclusions

In this paper we present ODMiner, a novel tool to support the process of generating of Linked Open Data from Open Data format such as XML, JSON and CSV. It provides a HTTP REST interface and it is easy to extend by adding new algorithms and supported formats. The evaluation that we carried on demonstrates the effectiveness of our tool, though further investigation is needed. Possible future work includes the extension of ODMiner in order to

[8] This JSON file is available on-line at http://eelst.cs.unibo.it/ld4ie/config.json.

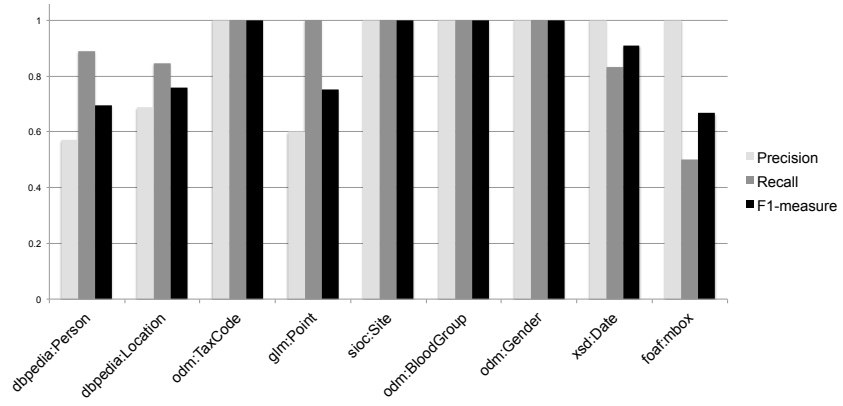


Fig. 2. Precision, recall and F1-measure computed data type recognition against the 20% of the whole dataset.

support more formats like XSLX or OpenOffice spreadsheets and the study of solutions to mine meaningful information that are distributed across multiple fields. For example, the latter is the case of dates represented as multi-column values in a CSV having a column for the day, another for the month and another for the year.

References

1. C. Bizer and R. Cyganiak. D2R server - publishing relational databases on the semantic web. In *Proceedings of ISWC2006 posters*, volume 175. CEUR-WS, 2006.
2. P. Ciancarini, F. Poggi, and D. Russo. Big data quality: a roadmap for open data. In *2016 IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService)*, pages 210–215. IEEE, 2016.
3. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
4. S. Kandel, A. Paepcke, J. Hellerstein, and J. Heer. Wrangler: Interactive visual specification of data transformation scripts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3363–3372. ACM, 2011.
5. C. Lange. Krextor - an extensible XML->RDF extraction framework. In *Proceedings of 5th Workshop on Scripting and Development for the Semantic Web at ESWC 2009*, volume 449. CEUR-WS, 2009.
6. V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Cybernetics and Control Theory*, 8(10):707–710, 1966.
7. P. Ristoski, C. Bizer, and H. Paulheim. Mining the Web of Linked Data with RapidMiner. *Journal of Web Semantics*, 35:142–151, 2015.
8. M. Verlic. Lodgrefine-lod-enabled google refine in action. In *I-SEMANTICS (Posters & Demos)*, pages 31–37, 2012.