Increasing Quality of Austrian Open Data by Linking them to Linked Data Sources: Lessons Learned*

Tomáš Knap[1,2]

[1] Charles University in Prague, Faculty of Mathematics and Physics
Malostranské nám. 25, 118 00 Praha 1, Czech Republic
knap@ksi.mff.cuni.cz
[2] Semantic Web Company
Mariahilfer Straße 70 / 8
A - 1070 Vienna, Austria
t.knap@semantic-web.at

Abstract. One of the goals of the ADEQUATe project is to improve the quality of the (tabular) open data being published at two Austrian open data portals by leveraging these tabular data to Linked Data, i. e., (1) classifying columns using Linked Data vocabularies, (2) linking cell values against Linked Data entities, and (3) discovering relations in the data by searching for evidences of such relations among Linked Data sources. Integrating data at Austrian data portals with existing Linked (Open) Data sources allows to, e. g., increase data completeness and reveal discrepancies in the data. In this paper, we describe lessons learned from using TableMiner+, an algorithm for (semi)automatic leveraging of tabular data to Linked Data. In particular, we evaluate TableMiner+'s ability to (1) classify columns of the tabular data and (2) link (disambiguate) cell values against Linked Data entities in Freebase. The lessons learned described in this paper are relevant not only for the goals of the ADEQUATe project, but also for other data publishers and wranglers who need to increase quality of open data by (semi)automatically interlinking them to Linked (Open) Data entities.

Keywords: Open Data, Linked Data, Data quality, Data linking, Data integration, Entity disambiguation

1 Introduction

The advent of Linked Data [1] accelerates the evolution of the Web into an exponentially growing information space where the unprecedented volume of data offers information consumers a level of information integration that has up to now not been possible. Consumers can now mashup and readily integrate information for use in a myriad of alternative end uses.

* This work has been supported in part by the Austrian Research Promotion Agency (FFG) under the project ADEQUATe (grant no. 849982).

In the recent days, governmental organizations publish their data as open data (most typically as CSV files). To fully exploit the potential of such data, the publication process should be improved, so that data are published as Linked Open Data. By leveraging open data to Linked Data, we increase usefulness of the data by providing global identifiers for things and we enrich the data with links to external sources.

To leverage CSV files to Linked Data[3], it is necessary to 1) classify CSV columns based on its content and context against existing knowledge bases 2) assign RDF terms (HTTP URLs, blank nodes and literals) to the particular cell values according to Linked Data principles (HTTP URL identifiers may be reused from one of the existing knowledge bases), 3) discover relations between columns based on the evidence for the relations in the existing knowledge bases, and 4) convert CSV data to RDF data properly using data types, language tags, well-known Linked Data vocabularies, etc.

To introduce an illustrative example of leveraging CSV files to Linked Data, if the published CSV file would contain names of the movies in the first column and names of the directors of these movies in the second column, the leveraging of CSV files to Linked Data should automatically 1) classify first and second column as containing instances of classes 'Movie' and 'Director', 2) convert cell values in the movies' and directors' columns to HTTP URL resources, e. g., instead of using 'Matrix' as the name of the movie, URL `http://www.freebase.com/m/02116f` may be used pointing to Freebase knowledge base[4] and standing for 'Matrix' movie with further attributes of that movie and links to further resources, and 3) discover relations between columns, such as relation 'isDirectedBy' between first and second column[5].

In this paper, we focus on the CSV files available at two Austrian data portals – `http://www.data.gv.at` and `http://www.opendataportal.at`. The first one is the official national Austrian data portal, with lots of datasets published by the Austrian government.

Our goal is not to find a solution, which automatically leverages tabular data to Linked Data, as this is really challenging and we are aware of that, but our goal is to help data wranglers to convert tabular data to Linked Data by suggesting them (1) concepts classifying the columns and (2) entities the cell values may be disambiguated to. To realize these steps, we evaluate TableMiner+, an algorithm for (semi)automatic leveraging of tabular data to Linked Data. By successfully classifying columns and disambiguating cell values, we immediately increase the quality of the data along the *interlinking* quality dimension [6].

The main contributions of this paper are lessons learned from evaluating TableMiner+ to classify columns and disambiguate cell values in CSV files obtained from the national Austrian open data portal. In [7], they also evaluate

[3] By leveraging the data we mean improving the way how data is published by converting it from CSV files to Linked Data, with all the benefits Linked Data provides [1].

[4] `http://freebase.com`

[5] The classes 'Movie' and 'Director' and the relation 'isDirectedBy' mentioned above should be reused from some well know Linked Data vocabulary

TableMiner+, nevertheless, (1) they do not evaluate TableMiner+ on top of CSV files and (2) they do not evaluate TableMiner+ on top of governmental open data, containing, e. g., lots of statistical data.

The rest of the paper is organized as follows. Section 2 discusses possible approaches for leveraging tabular data to Linked Data and justifies selection of TableMiner+ as the most promising algorithm for leveraging CSV files to Linked Data. Section 3 evaluates TableMiner+ algorithm on top of the data obtained from the national Austrian data portal. Section 4 summarizes lessons learned and we conclude in Section 5.

2 Related Work and TableMiner+

TableMiner+ is an algorithm for (semi)automatic leveraging of tabular data to Linked Data. TableMiner+ consumes table as the input. Further, it (1) discovers subject column of the table (the 'primary key' column containing identifiers for the rows), (2) classifies columns of the table to concepts (topics) available in Freebase, (3) links (disambiguates) cell values against Linked Data entities in Freebase, and (4) discovers relations among the columns by trying to find evidence for the relations in Freebase. TableMiner+ uses Freebase as its knowledge base; as the authors in [7] claim, Freebase is currently the largest knowledge base and Linked Data set in the world, containing over 2.4 billion facts about over 43 million topics (e. g., entities, concepts), significantly exceeding other popular knowledge bases such as DBpedia[6] and YAGO [5]. TableMiner+ is available under an open license – Apache License v2.0. There are also other algorithms with similar goals, such as Tabel [4] or the algorithm introduced in [3]. Extensive related work to TableMiner+ algorithm can be found in [7].

In [2], the authors present an approach for enabling the user-driven semantic mapping of large amounts tabular data using MediaWiki system. Although we agree that user's feedback is important when judging about the correctness of the suggested concept for classification or suggested entity for disambiguation, and completely automated solutions leveraging tabular data to Linked Data are very challenging, the approach in [2] relies solely on the user-driven mappings, which expects too much effort from the users.

Open Refine[7] with RDF extension provides a service to disambiguate cell values to Linked Data entities, e. g., from DBpedia. Nevertheless, the disambiguation is not interconnected with the classification as in case of, e. g., approach introduced in [7], so either user has to manually specify the concept restricting the candidate entities for disambiguation or all entities are considered during disambiguation, which is inefficient. Also the disambiguation is based just on the comparison of labels, without taking into account the context of the cell - further row cell values, column values, column header, etc.

[6] `http://dbpedia.org`

[7] `http://openrefine.org/`

We decided to use TableMiner+ to leverage CSV data from national Austrian data portal to Linked Data, because it outperforms similar algorithms, such as Tabel [4] or the algorithm presented in [3] and is available under an open license.

3 Evaluation

In this section, we describe the evaluation of TableMiner+ algorithm on top of CSV files obtained from the national Austrian data portal `http://data.gv.at`. First we provide basic statistics about the data we use in the evaluation and then we describe evaluation metrics and results obtained during evaluation of (1) subject column detection, (2) classification, and (3) disambiguation. We do not evaluate in this paper the process of discovering binary relations among columns of the input files.

Since the standard distribution of TableMiner+ algorithm[8] expects HTML tables as the input, we extended the algorithm, so that it supports also CSV files as the input.

3.1 Data and Basic Statistics

We evaluated TableMiner+ on top of 753 files out of 1491 CSV files (50.5%) obtained from the national Austrian data portal `http://data.gv.at`. The files processed were randomly selected from the files having less than 1 MB in size and having correct non-empty headers for all columns. We processed at most first 1000 rows from every such file. The processed files had in average 8.46 columns and 1.47 named entity columns.

3.2 Subject Column Detection

From all the processed files, we selected those for which TableMiner+ algorithm identified more than one named entity column and for those, we evaluated precision of the subject column detection by comparing the subject column selected by the TableMiner+ algorithm for the given file and the subject column manually annotated as being correct by a human annotator[9].

Results In 97.15% of cases, the subject column was properly identified by the TableMiner+ algorithm. There were couple of issues, e. g., considering column with companies, rather than with projects as the subject column in the CSV file containing list of projects. In case of statistical data containing couple of dimensions and measures, every dimension (except of the time dimension) was considered as a correctly identified subject column.

[8] `https://github.com/ziqizhang/sti`

[9] When talking about a human annotator here and further in the text, we always mean a person who has at least university master degree and at least basic knowledge of German language (to understand data within the Austrian portals).

3.3 Classification

In TableMiner+ algorithm, candidate concepts classifying certain column are computed in couple of phases. First, a sample of cells of the processed column is selected, disambiguated and the concepts of the disambiguated entities vote for the initial winning concept classifying the column. Further, all cells within that column are disambiguated, taking into account restrictions given by the initial winning concept, and, afterwards, all disambiguated cells vote once again for the concept classifying the column. If the winning concept classifying the column changes, disambiguation and voting is iterated. Lastly, candidate concepts for the given column are reexamined in the context of other columns and their candidate concepts, which may once again lead to the change of the winning concept suggested by TableMiner+ algorithm for the column. At the end, TableMiner+ algorithm reports the winning concept for every named entity column and also further candidate concepts, together with their scores (winning concept has the highest score).

To evaluate precision of such classification, for each processed file and named entity column, we marked down the candidate concepts for the classification together with the scores computed by TableMiner+ algorithm, sorted by the descending scores. Then we selected candidate concepts having 5 highest scores (since more candidate concepts may have the same score, this may include more than 5 candidate concepts). Afterwards, we selected a random sample of these selected candidate concepts (containing 100 columns) and let annotators to annotate for each file and column the classifications suggested by the TableMiner+ – annotators marked the suggested column classification either with *best*, *good* or *wrong* labels. Label *best* means that the candidate concept is the best concept which may be used in the given situation – it must properly describe the semantics of the classified column and it must be the most specific concept as possible as the goal is to prefer the most specific concepts among all suitable concepts; for example, instead of the concept *location/location*, the concept *location/citytown* is the preferred concept for the column containing list of Austrian cities. Label *good* means that the candidate concept is appropriate (it properly describes the semantics of the cell values in the column), but it is not necessarily the most suitable concept. Label *wrong* means that the candidate concept is inappropriate, it has a different semantics.

Let us denote $\#Cols$ as the number of columns annotated by annotators. Further, let us define function $top_N(c)$, which is equal to 1 if the candidate concept c annotated as *best* for certain column was also identified by TableMiner+ as a concept having up to N-th highest score, $N \in 1, 2, 3, 4, 5$. If $N = 1$ and $top_1(c) = 1$ for certain concept c, it means that the winning concept suggested by TableMiner+ is the same as the concept annotated as *best* by the annotators. Further, let us define metric $best_N$ which computes the percentage of columns in which the candidate concept c annotated as *best* for certain column was also identified by TableMiner as a concept having N-th highest score at worst; divided by total number of annotated named entity columns:

$$best_N = 100 \cdot \sum_c top_N(c)/\#Cols$$

So, for example, $best_1$ denotes the percentage of cases (columns) for which the concept annotated as *best* is also the winning concept suggested by TableMiner+.

The formula above does not penalize situations when more candidate concept share the same score. Since our goal is not to automatically produce Linked Data or column classification from the result of the TableMiner+, but we expect that user is presented with couple of candidate concepts (s)he verifies/selects from, it is not important whether (s)he is presented with 5 or 8 concepts, but it is important to evaluate how often the concept annotated as *best* is among the highest scored concepts.

Results The winning concepts (Freebase topics) discovered by the TableMiner+ algorithm running on top of all 753 files from the portal which were suggested for at least 20 columns and the number of columns for which these concepts were suggested as winning concepts are depicted in Table 1.

Freebase Concept	Number of Columns
location/location	478
music/recording	166
music/single	51
organization/organization	48
people/person	45
music/artist	35
location/statistical_region	34
location/dated_location	26
base/aareas/schema/administrative_area	25
fictional_universe/fictional_character	25
film/film_character	25
business/employer	22
location/citytown	22
music/release_track	22

Table 1. The winning concepts (Freebase topics) as discovered by TableMiner+

As we can see, majority of the columns were classified with the Freebase concept *location/location*. Although this is correct in most cases, typically, there is a better (more specific) concept available, such as *location/citytown*. There are also concepts, such as *music/recording* or $film/film\_character$, which are in most cases results of the wrong classification due to low evidence for correct concepts during disambiguation of the sample cells.

Selected results of the $best_N$ measure are introduced in Table 2. As we can see, 20% of concepts annotated as *best* were properly suggested by the TableMiner+

algorithm as the winning concepts; 36% of concepts annotated as *best* for certain columns were among concepts suggested by TableMiner+ and having highest or second highest score, etc. In other words, there is 76% probability that the concept annotated as being *best* will appear within candidate concepts suggested by TableMiner+ having 5th highest score at worst.

Furthermore, in 68% of the analyzed columns, only concepts annotated as *best* and *good* appear among concepts suggested by TableMiner+ and having 3rd highest score at worst.

In 24% of the analyzed columns, all concept candidates suggested by TableMiner+ were wrongly suggested. The reasons for completely wrong suggested classifications are typically two-fold: (1) low disambiguation recall due to low evidence for the cell values within the Freebase knowledge base or (2) wrong disambiguation due to short named entities having various unintended meanings.

We did not evaluated recall of the concept classification, as there was always a suggested concept classifying the column, although the precision could have been low.

N	$best_N$ (in percentage)
1	20
2	36
3	64
4	74
5	76

Table 2. Results of the $best_N$ measure

3.4 Disambiguation

For selected concepts from Table 1, we computed precision and recall of the entities disambiguation. Precision is calculated as the number of distinct entities (cell values) being correctly linked to Freebase entities divided by the number of all distinct entities (cell values) linked to Freebase (restricted to the given concept). Recall is computed as the number of distinct entities being linked to Freebase divided by number of all distinct entities (restricted to the given concept). To know which entities were correctly linked to Freebase, we again asked annotators to annotate, for the columns classified with the selected concepts, each distinct winning disambiguation of the cell value to Freebase entity – annotators could have marked the winning disambiguated entity either as being *correct* or *wrong*. The disambiguation is *correct* if the disambiguated entity represents correctly the semantics of the cell value. Otherwise, it is marked as *wrong*.

Results In case of *location/citytown* concept, we analyzed disambiguation of cities in 16 files, where the concept *location/citytown* was suggested as the winning concept by the TableMiner+ algorithm. The precision of the disambiguation was 95.2%; the recall 88.1%. We also analyzed other 24 files, where there was a column containing cities and one of the concepts (but not the winning concept) suggested by TableMiner+ classifying that column was *location/citytown* concept with the score being above 1.0. In this case, precision was 99% and recall 99.8%, taking into account more than 500 distinct disambiguated entities. It is also worth mentioning that TableMiner+ algorithm properly disambiguates and classifies cell values based on the context of the cell; thus, in case of the column with the cities, the cell value *Grambach* is properly classified as the city and not the river.

We analyzed 23 files where there was a column containing districts of Austria classified with the winning concept *location/location*. The precision was 38.3% and recall 100%. The precision is lower because in this case, more than half of the districts (e. g. *Leibnitz*, *Leoben*) were classified as cities. The reason why these columns were classified with the rather generic concept *location/location* and not with a more appropriate *location/administrative_division* is that some values within that column were disambiguated to cities and voted for *location/citytown*, some were disambiguated correctly to districts and voted for the best concept *location/administrative_division* and, since both these types of entities also belong to the concept *location/location*, this concept was chosen as the winning one.

Concept *base/aareas/schema/administrative_area* has high precision 88% and 100% recall, but there were only 17 distinct districts of Linz processed.

Concept *organization/organization* has reasonable precision for columns holding schools – it links faculties to the proper universities with precision 75% and recall 81%. For other types of organizations, such as pharmacies, hospitals, etc., disambiguation does not work properly, because there are no corresponding entities to be linked in Freebase.

Disambiguation of *people/person* concept has very low precision. The reason for that is that vast majority of people are not in the knowledge base. Also the precision of the concept *business/employer* is very low.

4 Lessons Learned

There is a high correlation between precision of the disambiguation and classification, which is caused by the fact that initial candidate concepts for the classification of a column are based on the votes of the disambiguated entities for the selected sample set of cells.

If the recall of the disambiguation is low (not much entities are disambiguated), it does not make sense to classify the column, as it will be in most cases misleading. In these cases, it is better to report that there is not enough evidence for the classification, rather than trying to classify the column somehow, because this ends up by suggesting completely irrelevant concepts, which confuses users.

Row context used by TableMiner+ algorithm proofed its usefulness in many situation. For example, it allowed to properly disambiguate commonly named cities having more than one matching entities in Freebase, i. e., the cities were properly disambiguated w.r.t. to the countries to which they belong.

If the cell values to be disambiguated are too short (e. g., abbreviations) and the precision of the subject column disambiguation, defining the context for these abbreviations, is low, it does not make sense to disambiguate these short cell values as the precision of such disambiguation will be low.

Classification/disambiguation in TableMiner+ has higher precision when the processed tabular data have subject column, which is further described by other columns, thus, classification/disambiguation may use reasonable row context. In case of statistical data, which merely involves measurements and dimension identifiers, the row context is not that beneficial and the precision of the classification/disambiguation is lower.

In many cases, the generic knowledge base, such as Freebase, is not sufficient as it does not include all needed information, e. g., it does not include information about all schools, hospitals, playgrounds, etc., in the country's states/regions/cities. So apart from generic knowledge bases, such as Freebase, also the focused knowledge bases should be used. Nevertheless, such focused knowledge bases must be available or must be constructed upfront.

TableMiner+ algorithm should use knowledge bases defining hierarchy of concepts within the knowledge base, as in many cases, more generic concepts were denoted as the winning concepts. Using hierarchy of concepts would improve performance and increase precision of the classification/disambiguation algorithm.

4.1 Contributions to Data Quality

Paper [6] provides a survey of Linked Data quality assessment dimensions and metrics. In this section, we discuss how successful classification and disambiguation conducted by TableMiner+ contribute towards higher quality of the resulting Linked Data along the quality dimensions introduced in [6].

Successful classification and disambiguation increase number of links to external (linked) data sources, thus, increase the quality of the data along the *interlinking* dimension [6]. By having links to external (linked) data sources, it is then possible to improve the quality of the data along the following quality assessment dimensions: [10]

- *Completeness*: It is possible to increase completeness of the data by introducing more facts about the entities from other (linked) data sources.
- *Semantic accuracy*: It is possible to reveal discrepancies in the data by comparing the resulting data with the data introduced in external (linked) data sources.
- *Trustworthiness*: It is possible to increasing trustworthiness of the data by providing further evidence for the data from external (linked) data sources.
- *Interoperability*: By reusing existing identifiers in external (linked) data sources, it is possible to increase interoperability of the data set.

[10] The names of the dimensions are taken from [6], where further description of the dimensions may be found.

5 Conclustions and Next Steps

We evaluated TableMiner+ algorithm on top of the Austrian open data obtained from the Austrian national open data portal `http://www.data.gv.at`.

We showed that in 76% of cases the concept annotated by humans as being the *best* in the given situation appears within the candidate concepts suggested by TableMiner+ with 5th highest score at worst. This is a promising result, as our main purpose is to provide to the data wranglers not only the winning concepts, but also certain number of alternative concepts.

Classification and disambiguation had very high precision for concept of cities (95%+) and reasonable precision for certain other concepts, such as districts, states, organizations. Nevertheless, for certain columns/cell values, the precision of the classification/disambiguation was rather low, which was caused either by (1) missing evidence for the disambiguated cell values in the Freebase knowledge base or (2) by trying to disambiguate cell values which have various alternative meanings. We showed that in 24% cases, the analyzed columns had irrelevant classification, which is rather confusing for users and in these cases it would be better not to produce any classification at all.

Although the first results are promising, we plan to experiment further (1) with different knowledge bases, such as WikiData[11], and (2) also plan to improve TableMiner+ algorithm, so that it behaves, e. g., more conservative in cases of low evidence for the classification/disambiguation.

References

1. C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1 – 22, 2009.
2. I. Ermilov, S. Auer, and C. Stadler. Csv2rdf: User-driven csv to rdf mass conversion framework. *Proceedings of the ISEM '13, September 04 - 06 2013, Graz, Austria*, 2013.
3. G. Limaye, S. Sarawagi, and S. Chakrabarti. Annotating and searching web tables using entities, types and relationships. *PVLDB*, 3(1):1338–1347, 2010.
4. V. Mulwad. *TABEL - A Domain Independent and Extensible Framework for Inferring the Semantics of Tables.* PhD thesis, University of Maryland, Baltimore County, January 2015.
5. F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA, 2007. ACM.
6. A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. Quality assessment for linked data: A survey. *Semantic Web Journal*, 2015.
7. Z. Zhang. Effective and efficient semantic table interpretation using tableminer+. *Semantic Web Journal*, 2016.

[11] `www.wikidata.org`