T2WML: A Cell-Based Language To Map Tables Into Wikidata Records

Pedro Szekely, Daniel Garijo, Jay Pujara, Divij Bhatia, and Jiasheng Wu

Information Sciences Institute
University of Southern California
4676 Admiralty Way, Marina del Rey, CA 90292, USA
{pszekely, dgarijo, jpujara}@isi.edu, {divijbha, jiashenw}@usc.edu

Abstract. The web contains millions of useful spreadsheets and CSV files, but these files are difficult to use in applications because they use a wide variety of data layouts and terminology. We present Table To Wikidata Mapping Language (T2WML), a language that makes it easy to map and link arbitrary spreadsheets and CSV files to the Wikidata data model. The output of T2WML consists of Wikidata statements that can be loaded in the public Wikidata, or loaded in a Wikidata clone, creating an augmented Wikidata knowledge graph that application developers can query using SPARQL.[1]

Keywords: Knowledge Graphs, RDF, Entity Linking, Wikidata

1 Introduction

The web contains millions of useful spreadsheets and CSV files, including data from many government and international organizations. Organizations that offer data often have web sites where users can search, browse and download data on a large number of topics. Most institutions offer their data in Excel and CSV formats. The downloaded data is seldom directly usable because, unlike databases, which use one column per variable, spreadsheets often arrange the data in different layouts.

Fig. 1 illustrates the problem using data about homicide rates in different countries, downloaded from the United Nations web site[2]. We truncated and colored the files for ease of presentation. The cells with the homicide numbers are highlighted in green, the cells that provide contextual information for the value are highlighted in blue, and header cells are highlighted in dark blue. Fig. 1a shows the layout of the data provided in the UN website; Fig. 1b shows a more compact representation using multi-level headers; Fig. 1c shows a layout

 This material is based upon work supported by United States Air Force under Contract No. FA8650-17-C-7715.

[2] https://dataunodc.un.org/crime/intentional-homicide-victims

a) Original

Country	Source		2000	2001	2002
Burundi	SDG	Females	1	2	2
Burundi	SDG	Males	4	5	6
Comoros	GHD Estimate	Females	2	1	
Comoros	GHD Estimate	Males	4	-	8
Djibouti	GHD Estimate	Females	2	1	3
Djibouti	GHD Estimate	Males	1	1	

b) Compact

Country	Source	Males			Females		
		2000	2001	2002	2000	2001	2002
Burundi	SDG	4	5	6	1	2	2
Comoros	GHD Estimate	4	-	8	2	1	
Djibouti	GHD Estimate	1	1		2	1	3

c) Database

Country	Source	Population	Year	Homicides
Burundi	SDG	Females	2000	1
Burundi	SDG	Females	2001	2
Burundi	SDG	Females	2002	2
Burundi	SDG	Males	2000	3
Burundi	SDG	Males	2001	4
Burundi	SDG	Males	2002	6
Comoros	GHD Estimate	Females	2000	2
Comoros	GHD Estimate	Females	2001	1
Comoros	GHD Estimate	Females	2002	

d) By year

Country	Source	Population	Homicides
2000			
Burundi	SDG	Females	1
	SDG	Males	3
Comoros	GHD Estimate	Females	2
	GHD Estimate	Males	4
Djibouti	GHD Estimate	Females	2
	GHD Estimate	Males	1
2001			
Burundi	SDG	Females	2
	SDG	Males	3
Comoros	GHD Estimate	Females	1
	GHD Estimate	Males	-

Fig. 1. Intentional Homicide Data (Excel file downloaded from dataunodc.un.org)

that could be used to store the data in a database, and that can be used directly in tools such as Pandas; Fig. 1d illustrates a common convention for arranging data by topic, by creating stacked tables that share common headings. All tables present the same homicide data. The interpretation of each value is defined by four cells (country, year, population and source) that identify the context for a value. In each table, the context cells are located in different parts of the data. Only in Fig. 1c (Database) the context cells are in the same row as the value; in the other tables, context cells appear in different rows, in header rows (examples a and b), or in visually distinct rows within the table (example d).

Existing languages for mapping structured data to RDF, including R2RML[3], RML [1], Karma [3] and CSV2RDF [2] process tabular data row by row, requiring tabular data to be in database format (Fig. 1c). RML supports non-tabular formats (JSON and XML) and Karma provides folding and unfolding operators to rearrange data for row-based processing. None support complex layouts such as those in examples b or d.

T2WML is a mapping language designed to meet three objectives: **1)** Identify and map data and their context qualifiers in arbitrary data layouts found in Excel and CSV files without the need of complex preprocessing steps to transform tables into a canonical "Database" representation; **2)** Enable users who are not familiar with RDF to map spreadsheets and CSV files to knowledge graphs so that they can augment knowledge graphs with data useful in downstream applications. **3)** Integrate mapping and linking so that the resulting output is linked to a reference knowledge graph, avoiding the need for a separate linking process that is often neglected in the current workflows.

[3] https://www.w3.org/2001/sw/wiki/R2RML

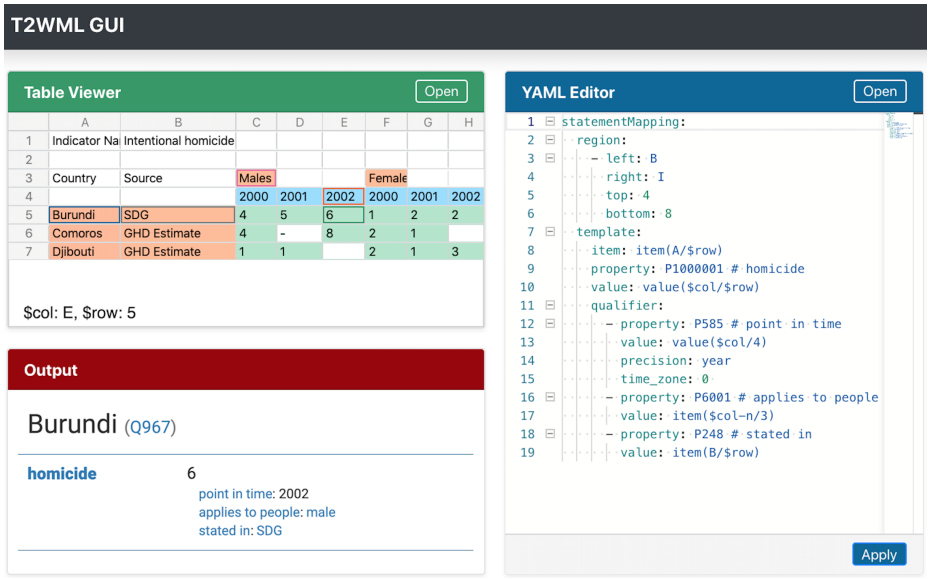

Fig. 2. T2WML user interface to map the spreadsheet in Fig. 1b to Wikidata

2 T2WML

T2WML uses two concepts to model spreadsheets and CSV files: **data blocks**, possibly non-contiguous blocks of cells that contain values of interest (green cells in Fig. 1), and **templates** to construct knowledge graph statements. T2WML uses the Wikidata item/statement data model [4] where items are described using *statements* consisting of a property, a value, qualifiers and references[4].

For ease of use, T2WML specifications are defined in YAML (see Fig. 2), a popular language for writing structured data. Another difference with existing mapping tools is that T2WML adopts a "link first" strategy where cells that identify entities in the target knowledge graph (Wikidata in our implementation) must be linked to the corresponding entities *before* the input data is processed to produce statements in the knowledge graph. Multiple tools exist to automatically link cells in tables to knowledge graphs[5]. The standard format for recording the linking results developed for this challenge, enables our T2WML tool to use the results from multiple linking tools, in addition to our own.

The T2WML user interface (Fig. 2) colors cells with clickable links in orange. Users can use the T2WML interface to curate (modify, add or delete) links before invoking our T2WML processor to produce statements in the target knowledge graph. The YAML Editor panel in Fig. 2 shows a T2WML specification to map the data in Fig. 1b to Wikidata. The `region` section identifies cell blocks containing the values of a statement (i.e., homicide counts, highlighted in green

[4] The Wikidata data model can be translated into RDF and queried using SPARQL
[5] www.aicrowd.com/challenges/iswc-2019-cell-entity-annotation-cea-challenge

in the Table Viewer). T2WML defines blocks in terms of the edges that surround the data. In the example, the edges are defined using constants, but it is possible to define blocks using predicates (e.g., "two empty rows" or "row with a value in column A, but otherwise empty".)[6]

The `template` section defines the mapping of cell values to elements of a statement. The T2WML tool instantiates the template once for every cell defined in the `region` section, binding the variables *$col* and *$row* to the coordinates of the cell being processed. To facilitate understanding of the template instantiation procedure, users can click on a data cell in the Table Viewer to see how it is mapped. The interface shows the values of *$col* and *$row*, highlights the cell containing the item (subject) of the statement, and the cells containing the qualifiers, and shows the resulting statement in an Output panel.

Users define the relationships between a value cell and other parts of a statement using a simple expression language. Fig. 2 illustrates several use cases: line 8 defines the item for the value in *$col*/*$row* as the item in column A and the same row (`A/$row`); line 13 identifies the "point in time" qualifier as the value in the current column in row 4 (`$col/4`); line 17, illustrates a more complex expression to define gender as the values of the "applies to people" qualifier. The value is located in row 3, in the closest non-empty column on the left. This is specified using the expression `$col-n/3`. The "-n" part directs the processor to substitute n with values 0, 1, ... until it reaches a non-empty cell. Simpler arithmetic expressions such as `$col+2/$row-4` are also supported.

3 Discussion

We tested the expressivity of T2WML using 19 variants of the homicides table (Fig. 1 shows 4 variants). We also created T2WML specifications for several country indicator files from the World Bank[7], generated Wikidata-compliant RDF and loaded it on a Wikidata clone. We are now working to enhance T2WML to support definition of new entities and properties, and constructing a corpus of mappings for use in research to automate the generation of mappings.

References

1. Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., and Van de Walle, R. Rml: A generic language for integrated rdf mappings of heterogeneous data.
2. Ermilov, I., Auer, S., and Stadler, C. Csv2rdf: User-driven csv to rdf mass conversion framework. In *Proceedings of the ISEM* (2013), vol. 13, pp. 04–06.
3. Gupta, S., Szekely, P., Knoblock, C. A., Goel, A., Taheriyan, M., and Muslea, M. Karma: A system for mapping structured sources into the semantic web. In *Extended Semantic Web Conference* (2012), Springer, pp. 430–434.
4. Vrandečić, D., and Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM 57*, 10 (Sept. 2014), 78–85.

[6] T2WML provides a rich expression language documented in http://anonymopus.

[7] For example, http://api.worldbank.org/v2/en/indicator/NY.GDP.PCAP.CN