3.1 T2WML Statement Mappings

We describe the language for mapping tables to statements using Fig. 2, which shows the user interface for the T2WML system. The panel on the left shows the table in Fig. 1a, and the YAML Editor panel shows a T2WML specification to map this data to Wikidata.

T2WML specifications consist of a data **`region`** to identify the cells containing the values of statements (e.g., homicide counts, highlighted in green in the Table Viewer in Fig. 2), and a **`template`** section to define the statements for each cell in the data region.

T2WML defines data regions using column and row expressions. Data regions confine an iterator to visit all cells containing data values. In our example, the data region is defined by columns `C` and right edge of the table (the default) and rows `4` and `14`. Using constant row and cell expressions yields correct results for this table, but we use a predicate to define the bottom edge, as the number of countries can change, and footnotes may be present at the bottom of the table. The predicate `bottom: value(B:/$row) = ""` states that the bottom edge is the first row where columns after `B` are empty.[6] The example also illustrates the ability to skip rows and cells using Boolean expressions. The cells in the data region are highlighted in green.

The `template` section defines the mapping of cell 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 (blue), the cells containing the qualifiers (pink), and shows the resulting statement in an output panel (bottom right of Fig. 2).

Users define the relationships between a value cell and other parts of a statement in the YAML editor using expressions defined above. Fig. 2 illustrates the definition of the subject, predicate, value and qualifiers of a statement, which we summarize below:

Subject: line 11 defines the subject of the statement for a value cell as the item in the same row in column A. For the value in cell F6, the item is Burundi, also shown in the output panel.

Predicate: the predicate of the statement is specified as a constant (`P100024`, defined in our clone of Wikidata). It is also possible to define the properties using the `item` function, a convenient feature to map spreadsheets where different columns contain information about different properties.

Value: the value of a statement is usually defined using the expression `value($col/$row)`, or function that transforms the value. T2WML offers a library of string, data and numeric transformation functions similar to those provided in spreadsheet software.

Qualifiers: the qualifiers of a statement are defined in the `qualifiers` section of the YAML file. Each qualifier consists of a predicate and a value. Lines 15 through 19 illustrate the definition of a time qualifier including specification of the value, calendar, precision and time zone.

References: references are defined in the `references` section of the YAML file (not shown in the figure), and are defined similarly to qualifiers.

[6] The : operator is a shortcut for the Boolean *and* operator.

4 DISCUSSION

T2WML is a new language and system under active development. While our experience with T2WML is limited, the results so far are encouraging.

We evaluated the expressivity of T2WML by creating 19 variants[7] of the homicides table downloaded from dataunodc.un.org. We created the variants from the Database layout (Fig. 1c), moving the qualifiers into header rows to emulate common layouts, including multiple header rows (Fig. 1b). We created stacked table variants (Fig. 1d) as tables of this type are also common in web sites. T2WML can map all variants except one where we combined the female and male homicide values in a single cell, separated by comma.

We used T2WML to create Wikidata statements for 9 World Bank indicators (all countries, all available years), and for the Fragile States Index indicators (https://fragilestatesindex.org). Political scientists are using these statements to build models querying our Wikidata clone.

We also used T2WML to map county-level crime data from the FBI.[8] We had originally written a 400-line Python script to map this data, and were able to replicate the results using a T2WML file of the same complexity as the one shown in Fig. 2.

Finally, we evaluated the usability of T2WML with a second-year undergraduate student who is creating Wikidata statements for public crime records in Los Angeles. The dataset contains over 2 million records and is updated weekly. Each crime record is highly detailed, including information such as the address, type of crime, time of day and location, source, etc. Despite not being an expert in mapping languages or RDF, the student was able to use T2WML to link crime records to Wikidata locations using Wikidata properties.

Our current and future work will focus on four directions. First, T2WML can create statements, but currently cannot create new Wikidata items or properties, or define new labels, aliases and descriptions. We will extend T2WML to support these tasks. Second, we want to integrate T2WML more tightly with Wikidata to support semantic operators to enable users to refer to elements of a table semantically (e.g., the column containing countries). Third, we are investigating machine learning approaches for property recommendation as we found that identifying the correct properties to use is the most difficult part of the mapping process. Finally, we are investigating ideas for publishing T2WML files in Wikidata to crowdsource the creation of mapping files for the significant number of spreadsheet and CSV files that exist on the web.

REFERENCES

[1] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. [n. d.]. RML: A Generic Language for Integrated RDF Mappings of Heterogeneous Data.
[2] Ivan Ermilov, Sören Auer, and Claus Stadler. 2013. Csv2rdf: User-driven csv to rdf mass conversion framework. In *Proceedings of the ISEM*, Vol. 13. 04–06.
[3] Shubham Gupta, Pedro Szekely, Craig A Knoblock, Aman Goel, Mohsen Taheriyan, and Maria Muslea. 2012. Karma: A system for mapping structured sources into the Semantic Web. In *Extended Semantic Web Conference*. Springer, 430–434.
[4] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM* 57, 10 (Sept. 2014), 78–85. https://doi.org/10.1145/2629489

[7] available at https://github.com/usc-isi-i2/t2wml/tree/master/Datasets
[8] https://bit.ly/2YdBrpn