SemInt at SemTab 2022

Abhisek Sharma[1,*,†], Sumit Dalal[1,†] and Sarika Jain[1,†]

[1] *National Institute of Technology Kurukshetra, India.*

Abstract

In this paper we present SemInt, for SemTab 2022 challenge of ISWC 2022. This is SemInt's first participation to the challenge. This challenge is about annotating tabular data from publically available knowledge graphs (such as Wikidata/DBPedia). We propose a model named as SemInt that runs iterative SPARQL query over Wikidata/DBPedia SPARQL endpoints for each term available a given table. For handling misformed or differing representations of terms or entities in the table, SemInt queries the Wikidata or DBPedia API's and find the suitable matches for them. It also employs a search engine to address typos in the terms. This year SemInt participated for CTA task and got some encouraging results with 0.794 Precision and F-measure. We plan to extend it for CEA and CPA as well.

Keywords

Entity annotation, Table interpretation, Knowledge graph, SemInt, SemTab

1. Introduction

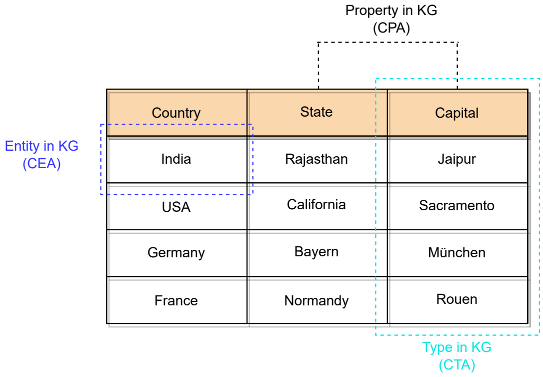
Web pages contains information of various dimensions. However, most of this information is present in the tables. Tables occupies relational data in various fields and are sources of high-quality data with lesser noise than unstructured text which is useful for various tasks knowledge graph augmentation [1] and knowledge extraction [2]. Hence tables can not be ignored while moving to the Web 3.0. Simple data (without any annotation) from tables don't have much meaning, but annotated tables are valuable sources and has critical research value. Semantic annotation of the tabular data has gained much attention in recent years. Most of the works employs probabilistic graphical models for the annotation purpose [3], [4]. There are several units in a table which can be annotated like cells, columns. A column or pair of columns can be assigned to entities, while relationship between two columns can be annotated to two cells from these columns. Though there are many benefits of annotating tables and employing them in knowledge extraction assignments. However, due to diverse languages and noise mentions, interpreting semantic data from tables by machines is not easy. SemTab chellenge is organized every year since 2019 on tabular data to Wikidata or DBpedia matching [5]. This year's challenge is to match the tabular data to Wikidata, DBpedia and Schema.org properties or classes depending on the rounds of the challenge. A new set of difficulties such as larger-

[*] Corresponding author.

[†] These authors contributed equally.

abhisek_61900048@nitkkr.ac.in (A. Sharma); sumitdalal9050@gmail.com (S. Dalal); jasarika@nitkkr.ac.in (S. Jain)

0000-0003-1568-2625 (A. Sharma); 0000-0002-8736-2148 (S. Dalal); 0000-0002-7432-8506 (S. Jain)



Country	State	Capital
India	Rajasthan	Jaipur
USA	California	Sacramento
Germany	Bayern	München
France	Normandy	Rouen

Figure 1: Tasks in SemTab 2022

scale knowledge graph setting, knowledge graph data shifting, and noisy schema structure of multiple knowledge graphs have followed. Additionally, this year's challenge also has a more challenging dataset (the tough tables [6]), which is manually curated, offering realistic issues than the last challenge. The Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2022) aims at benchmarking tabular data to knowledge graph matching systems. The challenge consists of three tasks: Column Type Annotation (CTA), Cell Entity Annotation (CEA) and Column Property Annotation (CPA). The CTA task is assigning a semantic type to a column, the CEA task is matching cells to entities in a specific KG, and the CPA task is assigning a KG property to the relationship between two columns. These three tasks and their formal definitions can be illustrated by Figure 1.

We have proposed an approach to solve the CTA task, where internally as insights we have used approach that gives some results for the CEA task, though we have not individually participated for CEA. For CTA task, we have used Wikidata/DBPedia SPARQL endpoint to query individual entities from each column and proceed from there.

Outline. The rest of the paper is organised as follows: Section 2 of the paper presents work from previous year SemTab challenges. Section 3 defines the proposed approach to solve the CTA task while Section 4 discusses the results for one rounds. Conclusion and future direction of this work is given in the last Section number 5.

2. Related Work

MTab tool supports multilingual tables and could process various table formats [7]. Referent entity for a cell in table is detected using a graphical model with iterative probability propagation algorithm in [8]. MTab4Wikidata [9] considers statement search and fuzzy search to handle noise mentions which improves entity search. Some works provided new formula for ranking the matching results such as DAGOBAH [10], MantisTable SE [11]. MTab system [12] is based on an aggregation of multiple cross-lingual lookup services and probabilistic graphical model. CSV2KG (IDLab) also uses multiple lookup services to improve matching performance [13]. Tabular ISI implements the lookup part with Wikidata API and Elastic Search on DBpedia

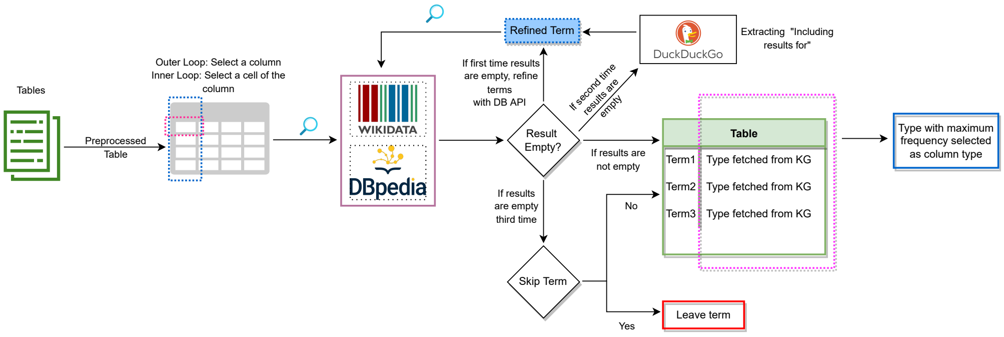

Tables
Preprocessed Table
Outer Loop: Select a column
Inner Loop: Select a cell of the column
WIKIDATA
DBpedia
Refined Term
DuckDuckGo
Extracting "Including results for"
If first time results are empty, refine terms with DB API
If second time results are empty
Result Empty?
If results are not empty
If results are empty third time
Skip Term
No
Yes
Leave term
Table
Type with maximum frequency selected as column type

Table	
Term1	Type fetched from KG
Term2	Type fetched from KG
Term3	Type fetched from KG

Figure 2: SemInt Architecture

labels, and aliases [14]. ADOG [15] system also uses Elastic Search to index knowledge graph. LOD4ALL first checks whereas there is available entity which has a similar label with table cell using ASK SPARQL, else perform DBpedia entity search [16]. DAGOBAH system performs entity linking with a lookup on Wikidata and DBpedia; the authors also used Wikidata entity embedding to estimate the entity type candidates [17]. Mantis Table provides a Web interface and API for tabular data matching [18].

3. Proposed Model

This section describes the architecture of our proposed system, named SemInt, whose various components are depicted in Figure 2. We have participated for the first time in SemTab, in the CTA task only. SemInt follows a simple, yet with decent results, majority-voting-based lookup approach: Cell contents are looked up in the SPARQL endpoint of the target KG, and in case of null results, looked up again on a search engine (DuckDuckGo) for fixing typos. The returned entity type with the highest number of votes per column is assigned as the type of that column.[1]

Assumptions

SemInt is developed keeping some assumptions in mind.

1. **Assumption 1** We assume that the input table contains values horizontally, i.e., column represent values of same type.
2. **Assumption 2** The cell and column types defined in Wikidata/DBPedia uses rdf:type and are of type owl:class.

3.1. Loading of tables and Selection of terms

A set of file with tables are provided in the beginning. Iteratively single files are selected and loaded as dataframe. SemInt then iterate over columns of loaded table selecting one at a time. Terms are then selected out of the selected column.

[1] Can be accessed through: https://github.com/abhiseksharma/SemInt

3.2. Lookup

The chosen term is supplied through a SPARQL query to retrieve various term types from the online DBpedia/Wikidata repository. If no result is received from the knowledge graph for any term then that term will be passed via respective API (DBpedia API or Wikidata API) to obtain the candidate representation of the term. This is done because an empty result may be caused by a difference in representation between the term stored in DBPedia/Wikidata and the representation in the table(like lowercase or camelcase, use of punctuations). Out of all the returned terms, first term is selected as in The query is then executed again once the candidate term has been obtained. If the result is still empty, the term is passed through a search engine (this version of SemInt uses DuckDuckGo search engine) to catch any typos by extracting ”including results for” part of the search result. DBPedia/Wikidata may have some representations that are accurately listed in the table but on which search engines may become confused, because of which this was not done in the first place. After the search engine has corrected any typos, the query is run one last time to seek for results that aren't empty. SemInt skips it and proceeds on to the following term in the line if the result is still empty.

When a result is not empty, it is saved as a table with terms in one column and types returned by the repository in the other.

We have used following SPARQL Query for the above lookup:

```
select DISTINCT ?o where
    {?s rdfs:label <term> @en . ?s
        wdt:P31 ?o .}
```

The <term> in the above query is the entry/concept/term in the cell of the dataset which will be queried for its type in DBPedia or WikiData (based on the dataset).

3.3. Type Selection

The frequency of entity types in the saved term-type table is taken into consideration while choosing the column type (for the CTA task). The column type is determined by the entity type with the highest frequency.

4. SemInt Performance and Results

This sections presents the performance and result of SemInt at SemTab 2022 in 1 out of the 3 rounds (i.e., Round 1) in which SemInt participated.

SemInt did went through the execution on dataset of round 2 and 3. In round 2, SemInt was able to get partial results locally, though was unable to execute completely due to some external factors. So, we had to skip submission for round 2. For round 3, SemInt ran completly on the dataset and produced some results, though after submission the evaluation scores (F1, recall, precision) came out as 0, we suspect the output KG types were represented in wrong format in the submitted CSV file.

Round 1

This year first round has 3 tasks, CTA-WD(Column Type Annotation using Wikidata), CEA-WD (Cell Entity Annotation using Wikidata), and CPA-WD (Annotating two columns with property on Wikidata). SemInt submitted results for CTA-WD task of Round 1 this year. The comparative results are presented in table 1

Table 1
Result of Round 1 for CTA-WD task

System	Precision	F1
DAGOBAH	0.975	0.975
s-elBat	0.951	0.957
Kepler-aSI	0.944	0.944
KGCODE-Tab	0.944	0.942
JenTab	0.940	0.938
AMALGAM	0.793	0.786
Laurent	0.785	0.770
SemInt	**0.794**	**0.794**

5. Conclusion

This paper presented the first version of SemInt approach. We are participating in this challenge for the first time. We have used a combination of strategies and treatment to tackle the tasks of SemTab 2022 and achieved encouraging performance. We have performed preprocessing,iterative term improvement techniques, and then iterative querying over SPARQL endpoint of Wikidata/DBPedia.

SemInt injects cell contents of a table into a generic SPARQL query. SemInt at SemTab 2022 is a promising approach, but which will be further improved. Our focus will be to decrease the complexity of the system in terms of space and time requirements. We will try to incorporate some Big Data or machine learning approaches to improve data processing. To speed up the process and handle the problem of large data we will employ parallel processing techniques and varying search strategies. Eventually, we want to cater the system for all the tasks i.e., CTA, CEA, and CPA over all the data sources.

References

[1] D. Ritze, O. Lehmberg, Y. Oulabi, C. Bizer, Profiling the potential of web tables for augmenting cross-domain knowledge bases, in: Proceedings of the 25th international conference on world wide web, 2016, pp. 251–261.

[2] D. Wang, P. Shiralkar, C. Lockard, B. Huang, X. L. Dong, M. Jiang, Tcn: Table convolutional network for web table interpretation, in: Proceedings of the Web Conference 2021, 2021, pp. 4020–4032.

[3] G. Limaye, S. Sarawagi, S. Chakrabarti, Annotating and searching web tables using entities, types and relationships, Proceedings of the VLDB Endowment 3 (2010) 1338–1347.

[4] V. Mulwad, T. Finin, A. Joshi, Semantic message passing for generating linked data from tables, in: International Semantic Web Conference, Springer, 2013, pp. 363–378.

[5] E. Jiménez-Ruiz, O. Hassanzadeh, V. Efthymiou, J. Chen, K. Srinivas, Semtab 2019: Resources to benchmark tabular data to knowledge graph matching systems, in: European Semantic Web Conference, Springer, 2020, pp. 514–530.

[6] V. Cutrona, F. Bianchi, E. Jiménez-Ruiz, M. Palmonari, Tough tables: Carefully evaluating entity linking for tabular data, in: International Semantic Web Conference, Springer, 2020, pp. 328–343.

[7] P. Nguyen, I. Yamada, N. Kertkeidkachorn, R. Ichise, H. Takeda, Semtab 2021: Tabular data annotation with mtab tool., in: SemTab@ ISWC, 2021, pp. 92–101.

[8] L. Yang, S. Shen, J. Ding, J. Jin, Gbmtab: A graph-based method for interpreting noisy semantic table to knowledge graph., in: SemTab@ ISWC, 2021, pp. 32–41.

[9] P. Nguyen, I. Yamada, N. Kertkeidkachorn, R. Ichise, H. Takeda, Mtab4wikidata at semtab 2020: Tabular data annotation with wikidata., SemTab@ ISWC 2775 (2020) 86–95.

[10] V.-P. Huynh, J. Liu, Y. Chabot, T. Labbé, P. Monnin, R. Troncy, Dagobah: Enhanced scoring algorithms for scalable annotations of tabular data., in: SemTab@ ISWC, 2020, pp. 27–39.

[11] M. Cremaschi, R. Avogadro, A. Barazzetti, D. Chieregato, E. Jiménez-Ruiz, O. Hassanzadeh, V. Efthymiou, J. Chen, K. Srinivas, Mantistable se: an efficient approach for the semantic table interpretation., in: SemTab@ ISWC, 2020, pp. 75–85.

[12] P. Nguyen, N. Kertkeidkachorn, R. Ichise, H. Takeda, Mtab: matching tabular data to knowledge graph using probability models, arXiv preprint arXiv:1910.00246 (2019).

[13] B. Steenwinckel, G. Vandewiele, F. De Turck, F. Ongenae, Csv2kg: Transforming tabular data into semantic knowledge, SemTab, ISWC Challenge (2019).

[14] A. Thawani, M. Hu, E. Hu, H. Zafar, N. T. Divvala, A. Singh, E. Qasemi, P. A. Szekely, J. Pujara, Entity linking to knowledge graphs to infer column types and properties., SemTab@ ISWC 2019 (2019) 25–32.

[15] D. Oliveira, M. d'Aquin, Adog-annotating data with ontologies and graphs, in: SemTab@ ISWC, 2019.

[16] H. Morikawa, Semantic table interpretation using lod4all., SemTab@ ISWC 2019 (2019) 49–56.

[17] J. Liu, R. Troncy, Dagobah: an end-to-end context-free tabular data semantic annotation system, SemTab@ ISWC (2019).

[18] M. Cremaschi, R. Avogadro, D. Chieregato, Mantistable: an automatic approach for the semantic table interpretation., SemTab@ ISWC 2019 (2019) 15–24.