Extracting New Knowledge from Web Tables: Novelty or Confidence?

Benno Kruit
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
kruit@cwi.nl

Peter Boncz
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
boncz@cwi.nl

Jacopo Urbani
Vrije Universiteit Amsterdam
Amsterdam, The Netherlands
jacopo@cs.vu.nl

ABSTRACT

To extend the coverage of Knowledge Bases (KBs), it is useful to integrate factual information from public tabular data. Ideally, the extracted information should not only be correct, but also novel. So far, the evaluation of state-of-the-art techniques for this task has focused primarily on the correctness of the extractions, but the novelty is less well analysed. To fill this gap, we replicated the evaluation of two state-of-the-art techniques and analyse the amount of novel extractions using two new metrics. We observe that current techniques are biased towards confidence, but this comes at the expense of novelty. We sketch a possible solution for this problem as part of our ongoing research.

ACM Reference format:
Benno Kruit, Peter Boncz, and Jacopo Urbani. 2018. Extracting New Knowledge from Web Tables:
Novelty or Confidence?. In *Proceedings of KBCOM 2018 First Workshop on Knowledge Base Construction, Reasoning and Mining - Short Paper, Los Angeles, February 2018 (KBCOM2018),* 5 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Motivation. Knowledge bases (KBs) are large repositories of factual knowledge which are (typically) available on the Web. Modern KBs (e.g., DBPedia [1]) contain millions of facts and are valuable assets in many tasks like semantic search, reasoning, etc. Unfortunately, despite their large sizes, they remain highly incomplete.

Much of the world's information exists as tabular data. On the Web, tables are available in web pages, as spreadsheets, or as publicly available datasets in many different formats. Because of their relational nature, tabular data is suitable for supporting entity search [20] or for answering specific factual queries [15]. Moreover, tables are used for structuring factual knowledge because the tables' cells often contain entities related to each other through some semantic relationships. Thus, tables represent an important source of knowledge for augmenting current KBs with useful knowledge.

Problem. So far, a significant amount of research has focused on the integration of tables with popular KBs like DBPedia or Freebase [3, 5, 7–10, 13, 14, 17, 21]. Broadly speaking, the integration process consists of two phases: First, the table at hand must be interpreted by associating its content with entities, types, and relations from the knowledge base. Then, the integration takes place by adding the information in the table to the KB, possibly filtering out low-quality extractions. This operation is also known as *slot-filling*, as the empty "slots" in the KB are filled with new facts [13].

The accuracy of the second phase relies on whether the first phase returns a sufficiently large number of links between the table and KB. On one extreme, if every cell and column can be linked 1:1 to concepts in the KB, then new facts can be extracted with high confidence but are likely to be redundant for slot-filling. On the other extreme, if the system is unable to make any link, then it cannot produce any new fact without introducing new KB concepts. This introduces a dilemma: On the one hand, links increase the confidence of extracted facts, but on the other one they hinder the novelty of the extraction.

Contribution. In this paper, we argue that current approaches for table interpretation rely strongly on the initial links to the KB and this introduces a sort of "bias" that encourages predictions of correct but redundant information which is useless for slot filling. More concretely, we formulate the following hypotheses about existing systems:

(H1) *Correctly extracted facts are more redundant than the unextracted ones.*
(H2) *Novel facts are extracted less often than redundant facts.*

In order to verify these hypotheses, first we introduce a new set of metrics, called *ReNew* metrics (*ReduntantNew*), to evaluate the performance of table interpretation systems w.r.t. the amount of redundant information they produce. Then, we present a replication study of two state-of-the-art methods for the integration of tabular data with KBs, T2K Match [12] and TableMiner+ [21], and analyse the amount of redundant extractions using our newly-introduced metrics. Finally, we sketch a potential solution for overcoming such "bias" as part of ongoing research.

2 FROM TABLES TO KNOWLEDGE BASES

Tables represent an important source of new knowledge for the KB, but the extraction is not trivial. First, it is necessary to run an interpretation step that maps the meaning of table cells, rows, headers, and columns to the concepts used in the KB. This task consists of identifying (1) which entities are present in the table, (2) which types those entities have, and (3) which relations are expressed between columns (if any) [8, 9, 12, 18, 21].

To describe this process, let us assume for instance that we have a KB with five entities {`Netherlands`, `Country`, `Amsterdam`, `City`, `capitalOf`} and a table X which contains a row r with cell values $r[1]$ = "Holland" and $r[2]$ = "A'dam". The first cell value should be linked to the entity `Netherlands`, while the second should be linked to the entity `Amsterdam`. The mapping is not trivial because

KBCOM2018, Los Angeles

DOI: 10.1145/nnnnnnn.nnnnnnn

a string can link to zero or multiple entities, e.g. "Holland" can map either to the county or to 19 different cities in the US, and the system has to disambiguate the correct meaning. Furthermore, if the other rows in the table also contain countries and their capital cities, then the system should infer that all these entities are instances of classes such as `Country` and `City`, and the relation between the columns should be `capitalOf`. Finally, after the interpretation is finished, we can use the links to construct facts to be added to the KB, e.g., ⟨`Amsterdam`, `capitalOf`, `Netherlands`⟩.

Several research directions have been explored to solve this task, with multiple systems focusing on cross-domain KB extension. The first system that integrated web tables with a KB was introduced in [8]. The system uses a probabilistic graphical model that combined a large number of features for making supervised predictions. Subsequent work approached the problem with a task-specific knowledge base [16, 18, 19], by limiting the feature set to speed up predictions [9], using distributed processing [6], or focusing on limited domains [10, 11, 17, 22]. Recently, state-of-the-art results have been obtained with the T2K Match [12] and TableMiner+ [21] systems.

To the best of our knowledge, T2K Match and TableMiner+ represent the most promising and mature systems for populating knowledge bases with the content of tables. They are open-source and available online[1]. The T2K Match system [12] implements a series of matching steps that match table rows to entities, using similarities between entity property values and the table columns. Beginning with entity candidate selection from cell values, the value-based similarities between cells and entity properties are then used to filter the candidate set and property predictions, after which they are recomputed on the new selection. This is iteratively repeated until the similarities stop changing and, if it exceeds a confidence threshold, a final prediction is chosen. The TableMiner+ system [21] consists of two phases that are alternated until a certain confidence level has been reached. The forward-learning phase builds up predictions on a row-by-row basis, after which the backward-update phase uses these to guide the interpretation of the rest of the data. This process is repeated until convergence.

These two systems were designed to work with different KBs, thus no comparison between them was even made. Moreover, the systems were evaluated against a set of manual annotations, and scored on the individual subtasks in terms of precision and recall. Such evaluation did not consider the facts that the system has extracted, but only the classification accuracy on the entity linking, type prediction, and relation prediction tasks.

In other words, no difference was made between predictions of already known facts or actual new knowledge. In order to fill this gap, we first need to define some new metrics that take into account the amount of redundant knowledge.

3 MEASURING REDUNDANCY

We are interested in using tables to expand a knowledge base, which we represent as a set of facts KB over a set of entities E_{KB}. The table extraction technique is expected to yield a new set of facts F_P over E_{KB}. For a set of tables in an held-out set, it is standard practice to manually annotate a gold standard set of facts F_G and use them for evaluating how many facts in F_P are correct. Notice that F_G might contain facts that are either in KB or not.

So far, current techniques have been evaluated w.r.t. the set of true positives $F_G \cap F_P$ (correctly extracted facts) and false negatives as $F_G \setminus F_P$ (valid facts that were missed). These measures do not capture the *redundant* information that was extracted. We propose two additional metrics to capture it. The first, which we refer to as *positive redundancy* (R^+), is the fraction of correctly extracted facts that are already in the knowledge base, and the second, *negative redundancy* (R^-), is the fraction of annotated but unextracted facts that are in the knowledge base:

$$R^+ = \frac{|(F_G \cap F_P) \cap KB|}{|F_G \cap F_P|} \qquad R^- = \frac{|(F_G \setminus F_P) \cap KB|}{|F_G \setminus F_P|} \tag{1}$$

In other words, R^+ is the redundancy of the true positives, and R^- is the redundancy of the false negatives. Notice that these measures work only if $F_G \setminus F_P \neq \emptyset$ and $F_G \cap F_P \neq \emptyset$ but these are conditions largely satisfied in practice. For example, imagine a table of 3 columns and 10 rows yielding $|F_G| = 20$ relational facts, of which 13 are already in the KB. If the technique at hand predicts only 10 correct facts but 8 of these are already in the KB, then $|F_G \cap F_P| = |F_G \setminus F_P| = 10$, $R^+ = 0.8$, and $R^- = 0.5$. Intuitively, R^+ reports the ratio of redundant information that was predicted, while R^- reports the ratio of redundant information that was not predicted. The two measures do not complement each other because they depend on both the predictive power of the technique and on the amount of novel information we can extract from the table. For instance, if the table yields only novel facts, than both R^+ and R^- will be zero regardless how good the extraction technique is.

Therefore, in order to have a more fine-grained view on the actual performance of the technique, we introduce also two *recall* scores that are sensible to the redundancy. The first, *novel recall* ($Q^\star$), is the fraction of new facts that is correctly extracted, and the second, *redundant recall* ($Q^\dagger$), is the fraction of redundant facts that is correctly extracted:

$$Q^* = \frac{|F_P \cap (F_G \setminus KB)|}{|F_G \setminus KB|} \qquad Q^\dagger = \frac{|F_P \cap (F_G \cap KB)|}{|F_G \cap KB|} \tag{2}$$

In other words, Q^* is the recall of novel annotations, and $Q^\dagger$ is the recall of known annotations. For the example above, $|F_G \setminus KB| = 7$, $|F_G \cap KB| = 13$, $Q^* \approx 0.29$, and $Q^\dagger \approx 0.62$. We argue that the measures $R^+, R^-, Q^\star, Q^\dagger$, which we call the *ReNew* measures, offer a better view of the performance than the used precision and recall because they take into account the actual number of novel knowledge that we can extract. Moreover, we can use them to formally state our hypotheses as follows:

$$R^+ > R^- \tag{H1}$$

$$Q^* < Q^\dagger \tag{H2}$$

Note that we are specifically interested in quantifying the extent to which table interpretation systems will extract redundant facts, and not in the general performance of the systems with regard to novel extractions. If we were only interested in the systems performance on the quality of their extracted facts, we could discard all redundant facts, and measure precision and recall of the remaining set of novel extractions. While these measures are useful

[1] http://dws.informatik.uni-mannheim.de/en/research/T2K, https://github.com/ziqizhang/sti/

Task	System	T2D-instance Precision	Recall	F_1
Entities Pr.	T2K Match	0.96	0.75	0.84
	TableMiner+	0.97	0.70	0.81
Type Pr.	T2K Match	0.93	0.92	0.92
	TableMiner+	0.94	0.91	0.93
Relations Pr.	T2K Match	0.83	0.60	0.70
	TableMiner+	0.75	0.40	0.51

Task	System	T2D-complete Precision	Recall	F_1
Relations Pr.	T2K Match	0.74	0.33	0.46
	TableMiner+	0.65	0.21	0.32

Table 1: Precision, recall and their harmonic mean F_1 for all datasets, tasks and systems.

Task	System	T2D-instance R^+	R^-	$Q^\star$	$Q^\dagger$
Entities Pr.	T2K Match Match	1.00	0.71	0.00	0.77
	TableMiner+	1.00	0.74	0.00	0.73
Types Pr.	T2K Match	1.00	0.71	0.00	0.77
	TableMiner+	1.00	0.74	0.00	0.73
Relations Pr.	T2K Match	0.81	0.22	0.10	0.63
	TableMiner+	0.83	0.32	0.04	0.36

Task	System	T2D-complete R^+	R^-	$Q^\star$	$Q^\dagger$
Relations Pr.	T2K Match	0.82	0.15	0.12	0.78
	TableMiner+	0.83	0.29	0.07	0.47

Table 2: ReNew metrics: positive redundancy, negative redundancy, novel recall and redundant recall for all datasets, tasks and systems.

for tuning systems for performance, in this work we are interested in analysing the behaviour of existing systems with regard to both novel and redundant extractions.

4 PRELIMINARY EMPIRICAL EVIDENCE

As mentioned earlier, we evaluate the systems T2K Match and TableMiner+ since they represent the current state-of-the-art for our task. For our experiments, we used two datasets from [12], which contain HTML tables from a large, cross-domain web scrape that are known to express relational data (i.e., not used for HTML layout purposes). These datasets contain a realistic selection of tables from the web, with manual annotations from DBPedia [1], a popular up-to-date KB. Thus, they are ideal for our purpose.

We evaluate the performance of the three key operations performed during the table interpretation process: 1) *Entity prediction*, which calculates the entity associated to each cell value. This process yields facts of the type $\langle entity, label, cell\_value\rangle$; 2) *Type Prediction*, which is the process to associate classes to the table's columns. This process yields facts of the type $\langle entity, type, class\_name\rangle$; 3) *Relations Prediction*, which is the process that determines the relationships between two different cells. This process yields facts of the type $\langle entity, relation, entity\rangle$. In order to evaluate the performance of the system, we need manual annotations for each of these three tasks.

The first dataset, called *T2D-instance* gold standard, consists of 233 tables with manual annotations of 25703 entities, 233 types and 420 relations from DBpedia. Using these annotations we could extract 75216 facts. The second (much larger) dataset, called *T2D-complete* gold standard, consists of 1748 tables. In this case, the manual annotations were limited to types (i.e., columns) and relations (between columns) from DBpedia. Entities (e.g. cell values) are not annotated. The lack of entity annotations precluded the usage of this dataset of our purposes. To fix this problem, we created a silver-standard set of entity annotations for each system by leveraging class predictions. If a class was correctly predicted for a column, then we assumed that the matching with the entities was correct. Using this method, we were able to extract 56509 and 48173 facts for T2K Match and TableMiner+, respectively. This still allows us to extract facts and calculate the redundancy scores. However, by definition in this case the entity and type matchings will be ideal. Therefore, we report the results only for the relations.

4.1 Accuracy

Initially, our goal was to reproduce the experiments presented in literature and compare the two systems using the standard precision, recall, and F_1. Running the T2K Match system was not particularly challenging since the implementation was already configured to use DBPedia. However, the TableMiner+ system [21] was originally designed for the Freebase knowledge base, and used services that have been discontinued. To provide a meaningful comparison, we minimally altered the system to use the same KB as the one used by the T2K Match framework. Moreover, we replaced the Freebase module by a label index and KB query index in Lucene, using the same interface. In this way, we could provide a meaningful comparison of the two systems.

The precision and recall were calculated following the definitions in [12] and [21]. Predictions of equivalent classes and relations were considered correct, and so were single-level superclasses [12].

The results we obtained are presented in Tab. 1. We can see that both systems perform very similarly on the T2D-instance dataset, particularly regarding entity and type prediction. The scores for T2K Match are comparable to the scores published in [12] which means we were able to reproduce the experimental analysis presented in literature. With the T2D-complete dataset, T2K Match significantly outperforms TableMiner+ on the relation prediction task. This may be due to the coherence that T2K Match calculates between all columns of a table and relations of a class, but a further error analysis is outside the scope of this paper.

4.2 Redundancy

We report in Tab. 2 the four ReNew metrics on the set of facts from entity, types, and relation predictions for both systems. First, we observe that R^+ and Q^* is 1 and 0 for the entity and types predictions respectively. These values are expected since by design both systems only accept mappings that are already in the KB. Thus

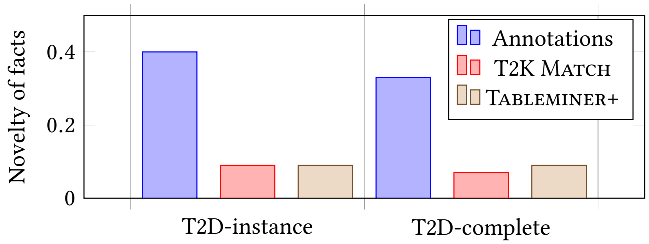

Figure 1: Avg. fraction of facts that is new, for facts extracted from annotations ($\frac{|F_G \setminus KB|}{|F_G|}$) and predictions ($\frac{|F_P \setminus KB|}{|F_P|}$).

the positive redundancy is maximal while novel recall is minimal. Notice however that R^- and $Q^\dagger$ do not have ideal values, which means that the systems do miss valid entity and type predictions because of this policy.

Furthermore, we can see that both hypotheses hold in every case and with both systems. This means that a large part of correct extracted facts is redundant (R^+ close to 1) and that a large part of unextracted facts is novel (low R^-). Moreover, the ratio of novel facts that is extracted (Q^*) is lower than the ratio of redundant extractions ($Q^\dagger$). If we compare the two systems, then we observe that negative redundancy (R^-) is higher with TableMiner+, which indicates that a larger fraction of missed facts are known. Also, novel recall (Q^*) is lower, which means that the system retrieved a smaller fraction of all novel facts that could have been extracted.

One could argue that since the tables do contain some redundant information, then it should be expected that the system also returns redundant predictions. To consider this case, Fig. 1 reports the average fraction of facts that is not in the KB for facts extracted from the gold standard ($\frac{|F_G \setminus KB|}{|F_G|}$) and for predictions returned by the two systems ($\frac{|F_P \setminus KB|}{|F_P|}$). This figure clearly illustrates that the ratio for the two systems is smaller than the amount of redundant information from the tables, which confirms (from an empirical perspective) our conclusion that the state-of-the-art is biased towards the prediction of already-known knowledge rather than novel one.

5 OVERCOMING REDUNDANCY

In this paper, we are concerned with the amount of redundant facts produced as a result of the integration of Web tables with existing KBs. To address this issue, we introduced a set of new metrics to evaluate whether and to what extent these systems are biased towards extracting data that is already in the knowledge base. These metrics concern the redundancy of extracted versus unextracted facts, and the recall of novel versus redundant facts. We used these metrics to formally capture the bias with two hypotheses, and verified them with an empirical comparison of two state-of-the-art systems. Our analysis indicates that correctly extracted facts are redundant more often than unextracted facts, and novel facts are indeed extracted less often than redundant facts.

How can we overcome this bias? We take inspiration from existing works and hint to a number of techniques which can be potentially used to reduce the redundancy of extractions. We divide them into three groups: extended feature sets, probabilistic KBs, and knowledge fusion.

Extended feature sets can be used in supervised systems to guide the interpretations away from redundant extractions by representing implicit ways that entities interact. In the original graphical model of [8], some features made use of the knowledge base ontology, using the type hierarchy and the range and domain of relations. Alternatively, the work of [11] models the incompleteness in a domain-specific subset of the knowledge base by estimating class probabilities based on relations between entities, which the limited domain makes tractable. However, to most effectively exploit these features for novel fact extraction, the objective function of the supervised model should account for redundancy.

Going a step further, some approaches quantify uncertainties using probabilistic KBs. The systems of [19] and [18] use a probabilistic KB created from a web corpus for supporting table search. This type of KB offers many strategies for improving the recall of new knowledge because it allows for an explicit model of unknown facts. This existing work however does not evaluate whether this approach actually leads to more novel extractions.

In data fusion approaches, systems explicitly aim for high recall, and use a post-processing filter to improve precision. In [10], the extracted facts are filtered using several machine learning models, and in [2] they are filtered using a statistical model of the KB. However, the first system does not disambiguate entities in cells but relies on hyperlinks in the table that point to Wikipedia pages, while the second relies heavily on an estimation of the trustworthiness of multiple data sources, which is not always available. In [13], the system of [12] is used to interpret a large collection of web tables, after which the extracted facts are filtered using several strategies. However, only 2.85% of web tables can be matched, which is attributed to a topical mismatch between the tables and the knowledge base. While such a post-processing step can be explicitly tuned to favor novel facts, it is still necessary for the extraction step to cover a very wide spectrum of topics.

Inspired by these techniques, we plan to explore strategies for overcoming the extraction bias that we found towards known facts. Our goal is to explicitly incorporate metrics of redundancy into a fusion system that first performs interpretations with high recall, and then filters extracted facts with high precision.

To overcome the topical mismatch of tables and knowledge bases, we plan to enrich the KB with contextual data from other sources, such as textual data, linked data from other sources, and anchor links on the web. While these data sources might be noisy, the enrichment will increases the coverage of domains and surface forms that can be used for table interpretation. For knowledge fusion, we will employ existing link prediction models to model the probability of novel fact extractions. This approach can be naturally combined with a model of the incompleteness of the KB [4].

To conclude, our work has shown that there is a tradeoff between the extraction of novel knowledge and the requirement of high confidence. While current systems appear to give more weight to confidence rather than novelty, our hope is that a combined usage of metrics that explicitly capture the redundancy, like our ReNew ones, and (some of the) techniques highlighted before will lead to more novel extractions.

REFERENCES

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. *The semantic web* (2007), 722–735.

[2] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. *Proceedings of KDD* (2014), 601–610.

[3] Ivan Ermilov and Axel Cyrille Ngonga Ngomo. 2016. TAIPAN: Automatic property mapping for tabular data. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10024 LNAI (2016), 163–179.

[4] Luis Galárraga, Simon Razniewski, Antoine Amarilli, and Fabian M Suchanek. 2017. Predicting completeness in knowledge bases. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 375–383.

[5] Xiaoyan Guo, Yueguo Chen, Jinchuan Chen, and Xiaoyong Du. ITEM: extract and integrate entities from tabular data to RDF knowledge base. In *Web Technologies and Applications* (2011). 400–411.

[6] Oktie Hassanzadeh, Michael J Ward, Mariano Rodriguez-Muro, and Kavitha Srinivas. Understanding a large corpus of web tables through matching with knowledge bases: an empirical study. In *OM* (2015). 25–34.

[7] Yusra Ibrahim, Mirek Riedewald, and Gerhard Weikum. Making Sense of Entities and Quantities in Web Tables. In *Proceedings of CIKM* (2016). 1703–1712.

[8] Girija Limaye, Sunita Sarawagi, and Soumen Chakrabarti. 2010. Annotating and Searching Web Tables Using Entities, Types and Relationships. *PVLDB* 3, 1-2 (2010), 1338–1347.

[9] Varish Mulwad, Tim Finin, and Anupam Joshi. Semantic message passing for generating linked data from tables. In *Proceedings of ISWC* (2013). 363–378.

[10] Emir Muñoz, Aidan Hogan, and Alessandra Mileo. 2014. Using linked data to mine RDF from wikipedia's tables. *Proceedings of WSDM* (2014), 533–542.

[11] Chenwei Ran, Wei Shen, Jianyong Wang, and Xuan Zhu. Domain-specific knowledge base enrichment using Wikipedia tables. In *Proceedings of ICDM* (2015). 349–358.

[12] Dominique Ritze, Oliver Lehmberg, and Christian Bizer. Matching HTML Tables to DBpedia. In *Proceedings of WIMS* (2015). 10:1–10:6.

[13] Dominique Ritze, Oliver Lehmberg, Yaser Oulabi, and Christian Bizer. Profiling the Potential of Web Tables for Augmenting Cross-domain Knowledge Bases Categories and Subject Descriptors. In *Proceedings of WWW* (2016). 251–261.

[14] Yoones A Sekhavat, Francesco Di Paolo, Denilson Barbosa, and Paolo Merialdo. Knowledge Base Augmentation using Tabular Data. In *LDOW* (2014).

[15] Huan Sun, Hao Ma, Xiaodong He, Wen-tau Yih, Yu Su, and Xifeng Yan. Table cell search for question answering. In *Proceedings of WWW* (2016). 771–782.

[16] Zareen Syed, Tim Finin, Varish Mulwad, and Anupam Joshi. Exploiting a web of semantic data for interpreting tables. In *Proceedings of the Second Web Science Conference* (2010), Vol. 5.

[17] Mohsen Taheriyan, Craig A. Knoblock, Pedro Szekely, and Jos Luis Ambite. 2016. Learning the semantics of structured data sources. *Web Semantics: Science, Services and Agents on the World Wide Web* 37 (2016), 152–169.

[18] Petros Venetis, Alon Halevy, Jayant Madhavan, Marius Pasca, Warren Shen, Fei Wu, Gengxin Miao, and Chung Wu. 2011. Recovering Semantics of Tables on the Web. *PVLDB* 4 (2011), 528–538.

[19] J Wang, Bin Shao, and Haixun Wang. 2010. Understanding tables on the web. In *ER*, Vol. 1. Springer, 141–155.

[20] Mohamed Yakout, Kris Ganjam, Kaushik Chakrabarti, and Surajit Chaudhuri. Infogather: entity augmentation and attribute discovery by holistic matching with web tables. In *Proceedings of SIGMOD* (2012). 97–108.

[21] Ziqi Zhang. 2016. Effective and efficient semantic table interpretation using tableminer+. *Semantic Web* Preprint (2016), 1–37.

[22] Stefan Zwicklbauer, Christoph Einsiedler, Michael Granitzer, and Christin Seifert. 2013. Towards disambiguating web tables. In *ISWC-PD'13 Proceedings of the 2013th International Conference on Posters and Demonstrations Track*. CEUR-WS. org, 205–208.