Web Table Taxonomy and Formalization

Larissa R. Lautert
Universidade Federal de Santa Catarina
Florianópolis, SC, Brazil
llautert@inf.ufsc.br

Marcelo M. Scheidt
Universidade Federal de Santa Catarina
Florianópolis, SC, Brazil
marcelo.scheidt@inf.ufsc.br

Carina F. Dorneles
Universidade Federal de Santa Catarina
Florianópolis, SC, Brazil
dorneles@inf.ufsc.br

ABSTRACT

The Web is the largest repository of data available, with over 150 million high-quality tables. Several works have combined efforts to allow queries on these tables, but there are still challenges, like the various different types of structures found on the Web. In this paper, we propose a taxonomy for the tabular structures and formalize the ones used with relational data and show, through an experimental evaluation, that WTClassifier, our supervised framework, classifies Web tables with high accuracy. Additionally, we use WTClassifier to categorize more than 300 thousand Web tables into our taxonomy and found that 82.25% are not formatted similarly to relational structure.

1. INTRODUCTION

According to Cafarella et al. [2], there are over 150 million HTML tables on the Web with high-quality relational data. The main incentive for working with this information is enabling analysis and integration of data on the Web, since it is the larger corpus of table ever seen. Several works have combined their efforts to properly identify [15, 16, 8], extract [9, 3, 12] and label [7, 14] this data, so it can be used as basis for structured queries. However, most of them have formatting different from relational databases. Hence, they should be individually analyzed to understand their structures.

Fig. 1 shows a Web table with high-quality relational data found in Wikipedia. Its structure addresses some challenges faced in table understanding. First of all, attribute names are disposed vertically, and not in a row, as in relational tables. Besides, there are two tables nested and attributes *Developed by* and *Language(s)* are multivalued. Data interpretation within these characteristics is easily understood by humans, but not by machines.

In face of the many structure types found in tables on the Web, it is necessary to catalog and interpret most used categories. Thus, we can simplify understanding and processing of Web structured data. Crestan et al. [6] have already proposed a taxonomy with nine heterogeneous structures, but we argue that some important types were ignored.

General information	
Title	Game of Thrones
Developed by	David Benioff D. B. Weiss
Language(s)	English, Dothraki
Broadcast	
Original channel	HBO
Picture format	1080i (HDTV)
Audio format	Dolby Digital 5.1

Figure 1: Web table with heterogeneous formatting.

In this paper, we propose Relational Knowledge Web table categories and present WTClassifier, an Artificial Neural Network-Based Classifier, which learns by analyzing features from each category. In summary, this paper makes the following contributions: (i) definition of a taxonomy for heterogeneous Web tables; (ii) formalization of its Relational Knowledge categories; and (iii) a framework for classification of heterogeneous Web tables.

The remainder of this paper is organized as follows. In Section 2, we propose a Web table taxonomy and formalize Relational Knowledge categories. Later, in Section 3, our classification framework is described. Its experimental evaluation is shown in Section 4, along with a classification of 342,795 tables collected from the Web. Section 5 discusses related work and compares our framework with a similar one. Finally, Section 6 presents conclusions e future work.

2. WEB TABLES TAXONOMY AND FORMALIZATION

The Relational Model [5] considers *relations* as data structures. Given n sets of domains $S_1, \cdots, S_n$, R is a relation on these n sets if it is a set of n-tuples each of which has its first element from S_1, its second element from S_2, and so on [5]. In view of this definition, we propose the subsequent defini-

tions to formalize Web tables, having the relational model definition as base. As these Web tables have relational purpose, i.e., are composed of relational data, they should also be treated as relations.

The early part of this section brings general concepts of Web tables. They are important to properly define each Web table category later, in Subsection 2.2. Web tables used without relational purposes are briefly described in Subsection 2.3.

2.1 General Concepts

Structure presented on Table 1, with x rows and y columns, will be used to illustrate some concepts.

Table 1: Structure of a Web table.

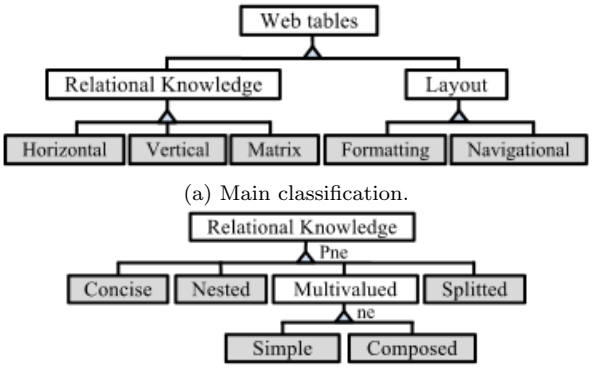
v_{11}	v_{12}	$\cdots$	v_{1y}
v_{21}	v_{22}	$\cdots$	v_{2y}
$\vdots$	$\vdots$		$\vdots$
v_{x1}	v_{x2}	$\cdots$	v_{xy}

Definition 2.1 (Web Table). *We call* WEB TABLES WT *tabular structures found in Web pages, composed of an ordered set of x rows and y columns.*

Definition 2.2 (Cell, Label, Data *and* Multivalued Data). *Let* WT *be a* WEB TABLE *with x rows and y columns, represented on Table 1. Each intersection between a row and a column determines a* CELL c_{ij}*, which has a value v_{ij}, where $1 \leqslant i \leqslant x$ and $1 \leqslant j \leqslant y$. Values v_{ij} come from set $L = \{l_1, ..., l_y\}$, composed of* LABELS*, or from set $D = \{d_1, ..., d_{(x-1)\cdot y}\}$, composed of* DATA*. The elements domain in L is string, while the elements domain in D might be strings,* WEB TABLES*, null values or a set of atomic values. A value v_{ij} is said to be* MULTIVALUED DATA *iff it is composed of a set of* DATA *values.*

Comparing with Relational Model [5], LABELS are equivalent to domain names, while DATA values corresponds to elements of these domains. According to notation presented in [1], LABELS correspond to *attributes*. The major difference between Relational Tables and WEB TABLES is that the first one has a unique structure, in which the first row corresponds to attribute names and the others, to tuples. Fig. 3a shows an example of WEB TABLE, where LABELS (*Title* and *Year*), located in the first row, correspond to attribute names for DATA below.

In order to better understand some WEB TABLES characteristics, it is important to introduce the concept of SEQUENTIAL CELLS. It will be used in the definition of MERGED CELLS, found in some categories presented later.



(a) Main classification.

(b) Secondary classification.

Figure 2: Classification of Web table types.

Definition 2.3 (Sequential Cells). *Let c_{ab} and c_{de} be two* CELLS *of a* WEB TABLE WT*, where a and d are indices for the rows and b and e, for the columns.* CELLS c_{ab} *and c_{de} are said to be* SEQUENTIAL CELLS *iff only one of the following conditions is true: $d = a + 1$ or $e = b + 1$.*

Let m, n and p be three arbitrary CELLS. If m is sequential to n and n is sequential to p, then m, n and p compose a set of SEQUENTIAL CELLS. For instance, in Fig. 5a, observe CELLS in the first ROW. The one containing value *PLANT* is sequential to CELL with *COLOR*, and this second is sequential to CELL with *HEIGHT*. Together, all the three compose a set of SEQUENTIAL CELLS.

Definition 2.4 (Merged Cell). *Let* MC *be a set of* SEQUENTIAL CELLS *of a* WEB TABLE WT*.* MC *is said to be* MERGED CELL *iff all their elements are associated to the same value v, with $v \in L$ or $v \in D$.*

An example can be seen in Fig. 5a, where values v_{21}, v_{22} and v_{23} are associated to *SHRUBS*. As they have the same value, they are presented in one condensed CELL. It is important to note that only SEQUENTIAL CELLS can form a MERGED CELL.

2.2 Relational Knowledge Web Tables Categorization

Considering all these definitions presented, we will now introduce and categorize WEB TABLE structures, whose hierarchical classification is presented in Fig. 2. Note that, as we are interested on tables with relational purpose, only Relational Knowledge ones, which can generate data in Relational Model, are formalized. The others are informally described in Subsection 2.3.

Definition 2.5 (Horizontal Web Table). *A* WEB TABLE WT *is said to be* HORIZONTAL *iff values v_{ij}*

($b \leqslant i \leqslant x$; $j = 1$) come from the domain S_1, values v_{ij} ($b \leqslant i \leqslant x$; $j = 2$), from domain S_2, and so on. b corresponds to index of the first row where $\exists\ v_{ij} \in D\ (\forall\ i; \forall\ j)$. If $\exists\ v_{ij} \in L\ (i = 1; \forall\ j)$, b assumes value 2 and $v_{ij} \in D\ (b \leqslant i \leqslant x; \forall\ j)$. Otherwise, $b = 1$ and values $v_{ij} \in D\ (\forall\ i; \forall\ j)$.

In Codd's definition [5] for the Relational Model, LABELS correspond to domain names on which the WEB TABLE WT is defined. The LABELS set is located on the first row, and the values of the remaining rows are DATA, likewise in a relational table. Fig. 3a exemplifies this type of structure.

Definition 2.6 (Vertical Web Table)**.** *A* WEB TABLE WT *is said to be* VERTICAL *iff values v_{ij} ($i = 1$; $b \leqslant j \leqslant y$) come from the domain S_1, values v_{ij} ($i = 2$; $b \leqslant i \leqslant y$), from domain S_2, and so on. b corresponds to index of the first column where $\exists\ v_{ij} \in D\ (\forall\ i; \forall\ j)$. If $\exists\ v_{ij} \in L\ (\forall\ i; j = 1)$, b assumes value 2 and $v_{ij} \in D\ (\forall\ i; b \leqslant j \leqslant y)$. Otherwise, $b = 1$ and values $v_{ij} \in D\ (\forall\ i; \forall\ j)$.*

In other words, WT is said to be VERTICAL if its tuples are disposed vertically. LABELS set, when present, is located in first column. In the structure shown on Table 1, LABEL represented by v_{11} would be associated to DATA located on the first row, i.e., from v_{12} until v_{1n}. This structure was informally defined as *Horizontal Listing* in [6] and as *Vertical Table* in [14]. Fig. 3b shows an example of this WEB TABLE, where LABELS (*Born*, *Residence*, *Nationality*, etc.) are located in the first column.

Definition 2.7 (Matrix Web Table)**.** *Let S_1, S_2 and S_3 be three different domains. A* WEB TABLE WT *is said to be* MATRIX *iff values $v_{ij} \in S_1$ ($i = 1$; $2 \leqslant j \leqslant y$), $v_{ij} \in S_2$ ($2 \leqslant i \leqslant x$; $j = 1$) and $v_{ij} \in S_3$ ($2 \leqslant i \leqslant x$; $2 \leqslant j \leqslant y$). Let V be a set with all values $v_{ij} \in$ WT. $v_{11} \in L$ and $(V - v_{11}) \in D$. Each value v_{ij} ($2 \leqslant i \leqslant x$; $2 \leqslant j \leqslant y$) belongs to the same tuple as v_{mj} ($m = 1$) and v_{in} ($n = 1$).*

In other words, each value that are not in the first row/column is associated to a value in row header and to another in column header. Fig. 4 shows statistics for car accident, where domain S_1 is *decades*, S_2 is *causes* and S_3 is the *number of accidents*. A human observer can easily note that CELL c_{22} content, the number 26, is not an instance of row header (*1980s*) neither column header (*Pilot error*), as they are no LABELS. It corresponds to the *Number of accidents* that happened on *1980s* by *Pilot error*. Value in CELL c_{11} is the only LABEL in this WEB TABLE, and in this case, is associated to the first column DATA values. There are no LABELS for *Decades* and *Number of accidents* values.

Year	Title
1925	*The Freshman*
1931	*Maker of Men*
1932	*Horse Feathers*

(a) Horizontal Web table.

Robert De Niro	
Born	August 17, 1943 New York, NY
Nationality	American
Occupation	Actor and director

(b) Vertical Web table.

Figure 3: Examples of Web tables

Cause	1980s	1990s	2000s
Pilot Error	26	27	30
Weather	14	10	8
Mechanical Failure	20	18	24

Figure 4: Example of Matrix Web table.

Definition 2.8 (Concise Web Table)**.** *Let* MC *be any* MERGED CELL. *A* WEB TABLE WT *is said to be* CONCISE *iff* WT $\supset$ MC.

In a CONCISE WEB TABLE, there is occurrence of MERGED CELLS to avoid repetition of values, so then it becomes more compact, i.e., concise. Challenges on interpreting this structure where already mentioned in recent work [14], where MERGED CELL is considered a sub-header for the rows below it. We see the problem in a most general way. In the case illustrated in Fig. 5a, MERGED CELL values represent common DATA for rows below and can be seen as sub-headers. It could be also represented as a new attribute posed in another column, with value *shrubs* for plants *azalea* and *buddleia*; and *cultivated annuals* for plant *alyssum*.

However, in the situation of Fig. 5b, CELLS disposed vertically were merged and do not act as sub-headers. In order not to repeat equal year values, which are the same for the films *Death at a Funeral*, *I Love You Too* and *Pete Smalls is Dead* (*2010*), original CELLS were merged into one.

Definition 2.9 (Nested Web Table)**.** *A* WEB TABLE WT *is said to be* NESTED *iff* $\exists\ v_{ij} \in$ WT *that is another* WEB TABLE.

The WEB TABLE presented in Fig. 1 is classified as NESTED. It can be observed that there are two WEB TABLES nested in one, separated for MERGED CELLS containing each WEB TABLE title.

PLANT	COLOR	HEIGHT
SHRUBS		
Azalea	variable	shrub
Buddleia	blue, pink, white	shrub
CULTIVATED ANNUALS		
Alyssum	violet, white	4 inches

(a)

Year	Title
2010	*Death at a Funeral*
	I Love You Too
	Pete Smalls Is Dead
2011	*A Little Bit of Heaven*

(b)

Figure 5: Examples of Concise Web tables

Definition 2.10 (Splitted Web Table). *A* WEB TABLE WT *is said to be* SPLITTED *iff its* LABELS *present sequential ordered repetitions in the row/column header. Let s be the number of these repetitions. Hence, each* LABEL *is repeated every* $\frac{z}{s+1}$ CELL(S)*, where $z = y$ if the* WT *is* HORIZONTAL*; and $z = x$ if it is* VERTICAL.

Comparing with Relational Model, we can say that each DATA row of a SPLITTED WEB TABLE is composed of $s+1$ tuples. This fact can be observed in Fig. 6, where the SPLITTED WEB TABLE has $s = 1$, i.e., it was horizontally splitted once. Thus, the set of LABELS consisting of *Rank*, *City name* and *Pop.* appears repeated once. In analogy with the Relational Model, it can be said that rows from 2 to 6 are composed of two tuples.

Rank	City name	Pop.	Rank	City name	Pop.
1	São Paulo	11,316,149	6	Belo Horizonte	2,385,639
2	Rio de Janeiro	6,355,949	7	Manaus	1,832,423
3	Salvador	3,093,605	8	Curitiba	1,764,540
4	Brasília	2,609,997	9	Recife	1,536,934
5	Fortaleza	2,476,589	10	Porto Alegre	1,413,094

Figure 6: Example of Splitted Web table.

Definition 2.11 (Multivalued Web Table). *Let v_{ij} be any value of* WEB TABLE WT. WT *is said to be* MULTIVALUED *iff* $\exists$ $v_{ij} \in$ WT *that is a* MULTIVALUED DATA*, composed of a set of k* DATA *values $\{m_1, ..., m_k\}$, which come from the domains $\{S_1, ..., S_k\}$, respectively.*

In this case, some DATA values are sets of other DATA values, as we will see in subsequent definitions.

Definition 2.12 (Simple Multivalued Web Table). *Let v_{ij} be a* MULTIVALUED DATA *of* WEB TABLE WT *with k* DATA *values.* WT *is said to be* SIMPLE MULTIVALUED *iff $S_1 = ... = S_k$, i.e., all k* DATA *values of v_{ij} come from the same domain.*

The WEB TABLE in Fig. 1 has MULTIVALUED DATA in two situations. The first one is in DATA value for *Developed by*, which is composed of two names (*David Benioff* and *D. B. Weiss*). The other case occurs in *Language(s)*, where the DATA value is a set of two strings (*English* and *Dothraki*).

Definition 2.13 (Composed Multivalued Web Table). *Let v_{ij} be a* MULTIVALUED DATA *of* WEB TABLE WT. WT *is said to be* COMPOSED MULTIVALUED *iff $S_1 \neq ... \neq S_k$, i.e., all k* DATA *values of v_{ij} are from different domains.*

The WEB TABLE in Fig. 3b has COMPOSED MULTIVALUED DATA in *Born* value, which consists of information about date and city of birth.

2.3 Layout Web Tables

Most of the HTML tables found on the Web are used only for layout purpose. According to [6], they can be divided in FORMATTING and NAVIGATIONAL. The first one is used in order to organize elements (text, images, videos, tables, etc.) in the page, while the second category disposes items of menus containing hyperlinks for navigating purpose.

3. WTCLASSIFIER, A NEURAL NETWORK-BASED CLASSIFIER

In order to automate the Web tables classification process, we have developed WTCLASSIFIER, a supervised Neural Network classifier using Neuroph[1]. Our framework learns from analyzing each category patterns, represented by a list of layout, HTML and lexical features, likewise approach used in [6]. Along with 20 features they described, we added 5 new ones: position of inner HTML tables and ratio of cells containing unordered lists, ordered lists, commas and brackets. The first feature helps on identification of Formatting Web table, while the other four often characterize MULTIVALUED DATA.

For the training phase, we have provided a list of 25 features extracted from 4,000 Web tables and their categories (golden collection). As output, we have 1 neuron per category, with value ranging from 0 to 1. We have used Multilayer Perceptron Network, with 1 hidden layer and resilient propagation. Classification process steps are described below.

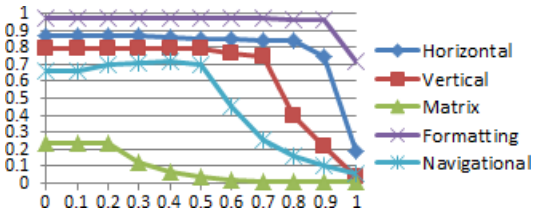
Main Classification separates Web tables in five categories: HORIZONTAL, VERTICAL, MATRIX, FORMATTING and NAVIGATIONAL. These classes are mutually exclusive, i.e., a Web table cannot be in more than one of these classes at a time.

Secondary Classification categorizes HORIZONTAL, VERTICAL and MATRIX in CONCISE, NESTED, SPLITTED, SIMPLE MULTIVALUED and/or COMPOSED MULTIVALUED. As input, we consider 25 features mentioned before plus the category obtained in Main Classification. Output categories are not mutually exclusive.

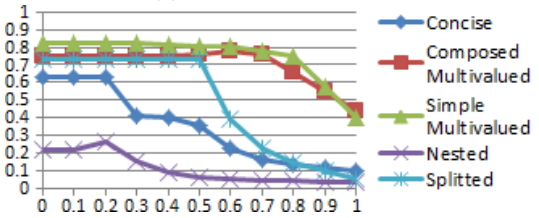
4. EXPERIMENTS

We have developed a crawler focused on extracting Web tables. As seeds, we have used Wikipedia, e-commerce, news and university sites, visiting a total of 174,927 pages, in which 104,261 contained Web tables. From these pages, we have extracted 631,382 HTML tables. Discarding repeated ones, 342,795 were left.

[1]Neural Network Framework (neuroph.sourceforge.net)

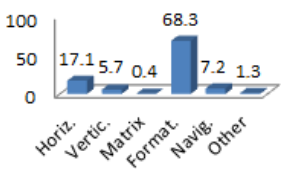


(a) Main Classification.

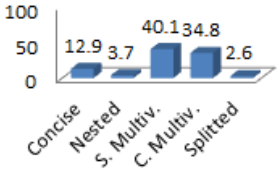


(b) Secondary Classification.

Figure 7: Recall vs precision graphs.



(a) Main Classific.



(b) Secondary Classific.

Figure 8: Distribution of Web tables.

4.1 Web Tables Distribution

First, we have ran WTClassifier to categorize the set with 342,795 Web tables in Main Classification. Latter, the ones first categorized as *Relational Knowledge* (75,233), went through Secondary Classification. Observing Fig. 8a, we note that the most used structure to represent relational data is HORIZONTAL (17.1%), followed by VERTICAL (5.7%). A total of 1.3% of our set was not classified in any of our categories. Observing Secondary Classification distribution, it can be seen that MULTIVALUED is the most often type, occuring in 74.9% of Relational Knowledge Web tables (40.1% INCLUSIVE plus 34.8% COMPOSED).

In order to verify how many Relational Knowledge Web tables were formatted similarly to relational structure, we have counted the number of HORIZONTAL occurences that do not fit in any category of Secondary Classification. We found that only 17.75% present this homogeneous structure.

4.2 WTClassifier Quality Evaluation

For each classification phase, there were generated 10 non-overlapping sets from our golden collection (90% of examples used for training and 10% for testing purpose). Using these values, through a 10-fold cross-validation process, we obtained precision and recall values represented in Fig. 7. MATRIX, the category with the lowest representation (Fig. 8a), is the one with worst precision values. This happens because there are too few cases of matrix in the training set (14). In Secondary Classification, it can be seen that WTClassifier is presenting difficulties on distinguishing categories where MERGED CELLS are more often (CONCISE and NESTED), and therefore, they present lower precision values. On the other hand, SIMPLE and COMPOSED MULTIVALUED have the best results due to their representatively in training set, plus full cover of their main patterns in features list described in Section 3.

Comparing F-measure values in Table 2 for overlapping categories, we can see that WTClassifier outperforms TabEx [6] in HORIZONTAL, VERTICAL, FORMATTING and NAVIGATIONAL categories.

	TabEx	WTClassifier
Horizontal	0.72	0.83
Vertical	0.24	0.71
Matrix	0.30	0.22
Formatting	0.86	0.95
Navigational	0.45	0.60

Table 2: F-measure for TabEx and WTClassifier.

5. RELATED WORK

Since Cafarella [2] attested the wealth of knowledge present in HTML tables, some studies have focused on them [7, 10, 14, 3, 13, 11]. The main issue with this data structure comes from the fact of them being made for human consumption, and therefore, machines have difficulty on interpreting some kinds of formatting. Heterogeneous structures for HTML tables, like vertical tables and lists presented in two dimensions were already noticed and dismissed in previous work [4, 14]. While most of them only worried about detecting HTML tables similar to relational tables [2, 4, 16], Yoshida et al. [17] came up with a method to integrate tables of the Web and proposed nine categories. However, these categories do not reflect the most common cases found on the Web today. A more complete taxonomy was proposed by Crestan et al. [6], where categories range from tables used only for layout purpose to structures similar to relational tables.

The most recent taxonomy proposed [6] has 7 categories of Relational Knowledge Tables. Even though this classification is useful, it presents some conceptual issues. One should note that, planning algorithms for bringing heterogeneous Web tables

to a unique structure, *Attribute/Value* (Web table with only one entity) and *Enumeration* (Web table with only one attribute) categories should be simply classified in *Vertical* or *Horizontal* categories; while *Form* type should not be classified as *Relational Knowledge Table*, as they have no high-quality relational data. Furthermore, they consider only inner HTML tables in classification process, assuming all ones are FORMATTING; while we add this characteristic as input on Neural Network and let it decide how important this feature is for each category. Another point is that since we are dealing with tables, which are, formally, *relations* [5], it would be appropriate to formally define them according to the Set Theory.

Comparing results in Fig. 8a with the one reported by Crestan et. al [6], we note that *Layout* categories (FORMATTING and NAVIGATIONAL) do not present similar values for Web table distribution. We suppose this is due to different seeds used for crawling. As we are most interested in analyzing Relational Knowledge structures, many of our Web tables come from Wikipedia, known for its high-quality data. Thereby, we obtained higher occurrences of HORIZONTAL and VERTICAL WEB TABLES. Moreover, tables classified as *Enumeration* and *Attribute/value* in *TabEx* are classified as *Horizontal* or *Vertical* in WTCLASSIFIER.

6. CONCLUSIONS AND FUTURE WORK

We have presented formalizations for Relational Knowledge Web tables, essential for defining algorithms to deal with them. Besides, we propose a taxonomy for Web tables categories, with five not mentioned in previous work. Comparing F-measure values of WTCLASSIFIER with reported results of TabEx [6], we note that our framework outperforms on identifying four, in a total of five categories in common. As future work, we highlight the importance of defining algorithms for bringing all categories to a unique structure. Thus, applications which deal with Web table data would not worry about their heterogeneous characteristics. Another issue consists of Web tables which subject is in surrounding text, and not within its structure.

7. REFERENCES

[1] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Add.-Wesley, 1995.

[2] M. J. Cafarella, A. Halevy, D. Z. Wang, E. Wu, and Y. Zhang. Webtables: exploring the power of tables on the web. *Proc. VLDB Endow.*, 1(1):538–549, Aug. 2008.

[3] M. J. Cafarella, J. Madhavan, and A. Halevy. Web-scale extraction of structured data. *SIGMOD Records*, 37(4):55–61, Mar. 2009.

[4] M. J. Cafarella and E. Wu. Uncovering the relational web. In *WebDB*, 2008.

[5] E. F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, 1970.

[6] E. Crestan and P. Pantel. Web-scale table census and classification. In *WSDM '11*, pages 545–554, New York, NY, USA, 2011. ACM.

[7] A. S. da Silva, D. Barbosa, J. M. B. Cavalcanti, and M. A. S. Sevalho. Labeling data extracted from the web. In *ODBASE*, pages 1099–1116. Springer, 2007.

[8] D. W. Embley, C. Tao, and S. W. Liddle. Automating the extraction of data from html tables with unknown structure. *Data Knowl. Eng.*, 54(1):3–28, July 2005.

[9] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Krüpl, and B. Pollak. Towards domain-independent information extraction from web tables. In *WWW*, pages 71–80, New York, NY, USA, 2007. ACM.

[10] G. Limaye, S. Sarawagi, and S. Chakrabarti. Annotating and searching web tables using entities, types and relationships. *Proc. VLDB Endow.*, 3(1-2):1338–1347, Sept. 2010.

[11] S. Mergen, J. Freire, and C. A. Heuser. Querying structured information sources on the web. In *iiWAS*, pages 470–476, New York, NY, USA, 2008. ACM.

[12] G. Nagy, S. C. Seth, D. Jin, D. W. Embley, S. Machado, and M. S. Krishnamoorthy. Data extraction from web tables: The devil is in the details. In *ICDAR*, pages 242–246, 2011.

[13] S. L. Sardi Mergen, J. Freire, and C. A. Heuser. Indexing relations on the web. In *EDBT*, pages 430–440, USA, 2010. ACM.

[14] P. Venetis, A. Halevy, J. Madhavan, M. Paşca, W. Shen, F. Wu, G. Miao, and C. Wu. Recovering semantics of tables on the web. *Proc. VLDB*, 4(9):528–538, June 2011.

[15] Y. Wang and J. Hu. Detecting tables in html documents. In *DAS*, pages 249–260, London, UK, UK, 2002. Springer-Verlag.

[16] Y. Wang and J. Hu. A machine learning based approach for table detection on the web. In *WWW*, pages 242–250, New York, NY, USA, 2002. ACM.

[17] M. Yoshida and K. Torisawa. A method to integrate tables of the world wide web. In *WDA*, pages 31–34, 2001.