Column Concept Determination for Chinese Web Tables via Convolutional Neural Network

Jie Xie[1,2], Cong Cao[2(✉)], Yanbing Liu[2], Yanan Cao[2], Baoke Li[1,2] and Jianlong Tan[2]

[1] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[2] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{xiejie,caocong,liuyanbing,caoyanan,libaoke,tanjianlong}@iie.ac.cn

Abstract. Hundreds of millions of tables on the Internet contain a considerable wealth of high-quality relational data. However, the web tables tend to lack explicit key semantic information. Therefore, information extraction in tables is usually supplemented by recovering the semantics of tables, where column concept determination is an important issue. In this paper, we focus on column concept determination in Chinese web tables. Different from previous research works, convolutional neural network (CNN) was applied in this task. The main contributions of our work lie in three aspects: firstly, datasets were constructed automatically based on the infoboxes in Baidu Encyclopedia; secondly, to determine the column concepts, a CNN classifier was trained to annotate cells in tables and the majority vote method was used on the columns to exclude incorrect annotations; thirdly, to verify the effectiveness, we performed the method on the real tabular dataset. Experimental results show that the proposed method outperforms the baseline methods and achieves an average accuracy of 97% for column concept determination.

Keywords: Column Concept Determination, Table Semantic Recovery, Chinese Web Tables.

1 Introduction

The Web contains a wealth of high-quality tabular data. Cafarella et al. [1] extracted 14.1 billion HTML tables from Google's general-purpose web crawl. Lehmberg et al. [6] presented a large public corpus of web tables containing over 233 million tables. Recently, the tabular data has been utilized in knowledge base expansion [7], table searching [4, 8] and table combining.

Normally, a table may have an entity column that contains a set of entities. Each row in the table is composed of the correlation attribute values of an entity. Each column is called an attribute that describes a feature of the entity set. Cells in a single column contain similar content. Compared with free-format text, tables usually contribute to valuable facts about entities, concepts and relations, which can usually favor the automatic extraction of data.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

However, web tables do not have any uniform schema and they tend to lack explicit key semantic information such as column headers, entity column notations and inter-column relations, which, if present, do not use controlled vocabulary. Taking the table in Fig. 1 as an example, it does not have a column header, so the information in such tables is difficult for machines to use. A key challenge is to make such information processable for machines, which usually contains but not limited to three tasks: identify entity columns, determine column concepts and annotate column relations.

萨乌丁 (Dmitry Sautin)	男 (Male)	俄罗斯 (Russia)	奥运会七枚跳水奖牌 (won seven diving medals in the Olympic Games)
高敏 (GaoMin)	女 (Female)	中国 (China)	第 24、25 届奥运会女子跳水 3 米板金牌得主 (won the women's 3m springboard gold medal at the 24th and 25th Olympic Games)
田亮 (TianLiang)	男 (Male)	中国 (China)	雅典奥运会男子双人十米跳台冠军 (won the men's diving synchronized 10m platform gold medal at the Athens Olympics)
郭晶晶 (Guo Jingjing)	女 (Female)	中国 (China)	2004 年、2008 年奥运会女子 3 米板金牌 (won the women's 3m springboard gold medal in 2004 and 2008)

Fig. 1. A web table without a column header.

In this paper, we focus on column concept determination task in Chinese web tables, which determines the most appropriate concept for each column in Chinese web tables. To the best of our knowledge, this is the first work to recover the semantics of Chinese web tables. For English tables, most of the state-of-art methods were based on knowledge bases [2, 3, 7, 10, 11, 13], or databases extracted from the Web [4]. These methods can only annotate tables with facts existing in the knowledge bases, but have difficulty to discover new (or unknown) knowledge. Due to the fact that cells in Chinese tables do not have uniform specifications and may contain a certain amount of long sentences (see Fig. 1), the knowledge base-based approaches are not applicable. Therefore, we assumed column concept determination as a classification problem which we solved by leveraging convolutional neural network (CNN). In summary, we made the following contributions:

- We used the infoboxes in Baidu Encyclopedia to automatically construct datasets for text classifier.
- We trained a classifier based on CNN to annotate cells in Chinese web tables and used majority vote to exclude cells with incorrect annotations, and then we determined the concept for each column in the tables.
- We verified the effectiveness of our method on the real tabular dataset.

The rest of this paper is organized as follows. After reviewing the related works in Section 2, we give the problem definition in Section 3. In Section 4, we describe the

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

proposed method in detail and experimental results are reported in Section 5. Finally we make a conclusion in Section 6.

2 Related Work

WebTables [1] showed that the World-Wide Web consisted of a huge number of data in the form of HTML tables and the research of using web tables as a high quality relational data source was initiated. Recently, a number of studies have appeared with the goal of recovering the semantics of tables to make fully use of tabular data.

For English tables, most of the state-of-art methods were based on knowledge bases [2, 3, 7, 10, 11, 13] or databases extracted from the Web [4]. Limaye et al. [3] proposed a graphical model to annotate tables based on YAGO. Wang et al. [2] used Probase [9] to generate headers for tables and identify entities in tables. Ritze et al. [7] proposed an iterative matching method (T2K) which combined schema and instance matching based on DBpedia [15]. Deng et al. [10] determined column concept by fuzzily matching its cell values to the entities within a large knowledge base. These methods have difficulty to discover new (or unknown) knowledge that do not exist in the knowledge bases.

Different from the methods mentioned above, Quercini et al. [5] utilized support vector machine (SVM) to annotate entities in English web tables. It searched information on the Web to annotate entities. However, it only focused on entity identification task in English web tables.

Considering the fact that many existing Chinese knowledge bases either are for internal use or contain insufficient knowledge for the annotation task, this paper takes the idea of text classification to deal with column concept determination task in Chinese web tables, and leverages the good feature extraction and classification performance of the convolutional neural network.

3 Problem Definition

This paper aimed to study the problem of determining column concept for Chinese web tables. The formal description of the problem will be shown in this section.

- **Web Tables**: Let T be a table with n rows and m columns. Each cell in the table can be represented as $T(i, j)$, $1 \le i \le n$ and $1 \le j \le m$ being the index of the row and column respectively. $T(i, j)$ can be a long sentence besides a word or a few words, just like the fourth column in Fig. 1. In addition, we assume that the contents in web tables are mainly Chinese. In fact, our method works as well when the table contains several English words. In this research, we model T as a bi-dimensional array of $n \times m$ cells, limiting the research scope to tables with no column branches into sub columns.
- **Column Concept Determination**: Given a table T, the method must classify a type for each cell in T and then determine the concept for each column. More formally, for each cell $T(i, j)$, the method firstly annotates it with a type $t_{i,j}^{(k)}$.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

Then for the j_{th} column, if the majority of cells are annotated with type $t^{(k)}$ in the type set, we choose $t^{(k)}$ as the concept of this column.

4 Column Concept Determination

We firstly present the process of our method and then describe our model based on convolutional neural network.

Since this paper is not focus on how to identify entities in tables, we assume that the entity column is already known and presented in the first column of a table. The column concept determination problem can be viewed as a classification problem. The principle of the method is to make use of the wealthy information available on the web to enrich the scarce context of tables. Our method uses Baidu Encyclopedia to obtain related text for attributes in tables. Compared with search engine used in [5], it is easier to get useful information for attributes from Baidu Encyclopedia. The process of the method is presented in Fig. 2:

1. Submit the entities of the table (in the entity column) to Baidu Encyclopedia.
2. Extract related text for the attribute values in the tuple from their corresponding entity pages.
3. Put the related text into classifier and find out a type for each cell.
4. Use majority vote to determine the most appropriate concept for each column.

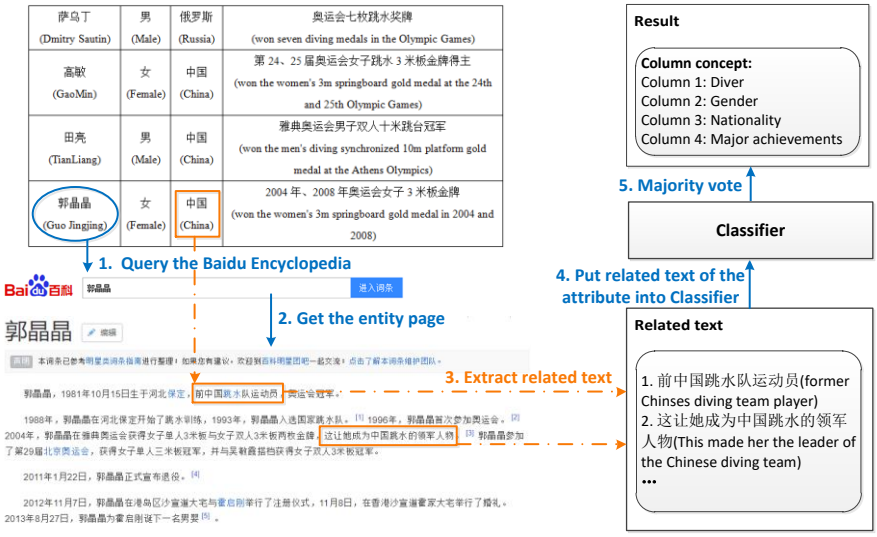

Fig. 2. The process of column concept determination.

The proposed method consists of three steps: pre-processing, annotating and post-processing. We will detail the three steps in the remainder of this section.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

4.1 Pre-processing

Pre-processing enriches the context of attribute cells in tables with the related text from the Baidu Encyclopedia. Each row R_i in the table can be represented as $(e, p_1, p_2, ..., p_m)$, where e is the entity of the row and $p_1, p_2, ..., p_m$ are attribute values of e.

For each row R_i in the table, the method uses the entity e to query Baidu Encyclopedia and get the article returned. It iterates through the article and segment the document into sentences. Then we use each attribute value p_i as a keyword to extract sentences containing p_i to form its related text $RT(p_i)$. If an attribute value is composed of several sub-values or a long sentence (e.g., the last column of table in Fig. 1), we split it into several words, remove words that conflict with other attributes and then find the sentences that contain one or more of these keywords.

We are tending to solve the problem of determining column concepts for Chinese web tables. Since Chinese text does not use spaces to separate words as English text does, the method firstly uses the Chinese word segmentation technology to process the related text $RT(p_i)$ of each attribute value p_i and get a set of words L_{p_i}.

4.2 Annotating

The classification model is summarized in Fig. 3. The method feeds the word set L_{p_i} obtained from the Pre-processing step to the CNN model. The lookup table layer converts these words into word embedding. The convolutional layer extracts features of input data followed by an average pooling layer. Finally, we use a fully connected layer with dropout and Softmax classifier in our output layer. We will introduce the model layer by layer in the following parts.

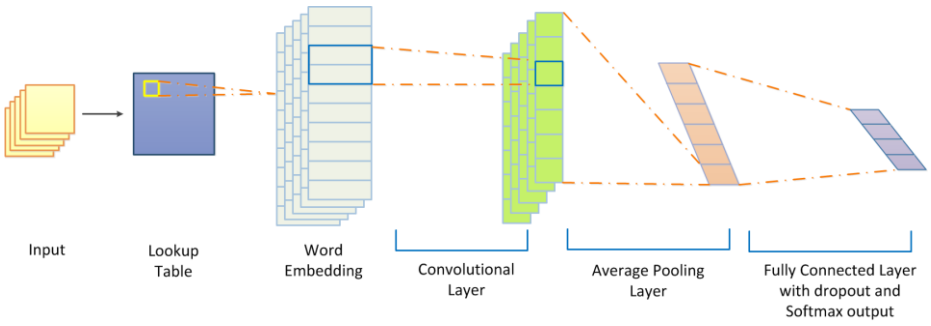

Fig. 3. Classification model based on CNN.

Firstly, we want to give a notation. A neural network with k layers can be considered as a composition of functions f, corresponding to each layer:

$$f_\theta(\cdot) = f_\theta^k\left(f_\theta^{k-1}\left(\cdots f_\theta^1(\cdot)\cdots\right)\right) \tag{1}$$

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

Lookup Table Layer. Convolutional neural network (CNN) is essentially a mathematical model. When using CNN for text classification, we firstly use word representation to convert the input words into vectors by looking up embedding. We have already got the related text of attribute p_i, which was represented by a set of words L_{p_i}. Then each word in L_{p_i} is passed through the lookup table layer to produce a numeric vector of ML_{p_i}. These numeric vectors can be viewed as the initial input of the standard CNN. More formally, the initial input numeric vector ML_{p_i} fed to the convolutional layer is given by the lookup table layer $LTF(\cdot)$:

$$f_\theta^1(\cdot) = ML_{p_i} = LTF\left(L_{p_i}\right) \tag{2}$$

There are many freely available word representation models, such as Google word2vec. We utilize word2vec to train a lookup table for representing the words.

Convolutional Layer. This layer contains several convolutional operations. Convolution is an operation between a weight vector W and the numeric vector ML_{p_i} from the Lookup Table Layer (2). The weights matrix W is regarded as the filter for the convolution:

$$f_\theta^k(\cdot) = Conv\left(f_\theta^{k-1}(\cdot), W\right) \tag{3}$$

The convolutional layer is used to extract higher level features between the attribute values and their related text.

Average Pooling Layer. The size of the convolution output depends on the number of words in L_{p_i} fed to the network. To apply subsequent standard affine layers, the features extracted by the convolutional layer have to be combined such that they are independent of the length of word set L_{p_i}. In convolutional neural networks, average or max pooling operations are often applied for this purpose. The max operation does not make much sense in our case. Since we use an attribute value to extract related sentences in the article, in general, several sentences will be matched. These sentences are complementary to each other and determine the type of the attribute value jointly. So in this work, we use an average approach, which forces the network to capture the average value of local features produced by the convolutional layer:

$$\left[f_\theta^k\right]_i = \underset{t}{mean}\left[f_\theta^{k-1}\right]_{i,t} \tag{4}$$

Where t is the number of output of the $k-1$ layer. The fixed size global feature vector can be then fed to the output layer.

Output Layer. The output of the average pooling layer is fed into the output layer through a fully connected network with dropout:

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

$$f_\theta^k(\cdot) = ReLU\left(Wf_\theta^{k-1}(\cdot) + b\right) \tag{5}$$

Where W is the weight matrix and b is the bias. We use $ReLU$ as the active function. A Softmax classifier is used to compute a score of each possible type if we give the final weight matrix W and bias b. Formally, the final output can be interpreted as follow:

$$f_\theta^k(\cdot) = Softmax\left(Wf_\theta^{k-1}(\cdot) + b\right) \tag{6}$$

Softmax is a multiclass classifier. For each type in the type set $Type = \left\{t^{(1)}, t^{(2)}, ..., t^{(k)}\right\}$, Softmax outputs the probability that a sample belongs to different types in the form of $S = \left\{S^{(1)}, S^{(2)}, ..., S^{(k)}\right\}$. Then the type $t^{(k)}$ with the largest score is selected as the type of this attribute value.

4.3 Post-processing

Since the cells in a single column have similar contents, we therefore leverage the column coherence principle to rule out the cells annotated incorrectly. For the j_{th} column in a table, our method combined the annotation of the cells in column j based on majority vote. If most of the cells in a column are assigned with a type $t^{(k)}$, we choose the type $t^{(k)}$ as the concept of the j_{th} column.

5 Experiment

We performed several experiments to evaluate the proposed method. This is the first work focused on column concept determination task in Chinese web tables and it is infeasible for us to compare our method with the methods designed for English web tables for the following reasons:

- The inputs are different. We can't use the English web tables as input for our proposed method;
- The previous methods are not reproducible. It is impossible for us to reproduce the knowledge base-based method designed for English tables and then use them for our task. Since many existing Chinese knowledge bases either are for internal use or contain insufficient knowledge, it is hard for us to find a Chinese knowledge base as large as Probase [9] and DBpedia [15].

We use the Naïve Bayes classification techniques (BAYES) and support vector machine (SVM) as baselines. In section 5.1, we describe the method to construct datasets and present the training and test sets obtained for classifier training. In section 5.2, we evaluate the performance of three text classifiers based on BAYES, SVM and CNN separately. In section 5.3, we discuss the evaluation results obtained by running our method on a set of web tables extracted from the web pages.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

5.1 Dataset Construction

To make our approach scalable, we need to construct the training and test datasets that involves as little manual intervention as possible.

Inspired by [14], we found that the infoboxes in Baidu Encyclopedia contained tabular summaries of objects' key attributes. So they can be used as a source of data. We used a web crawler to extract entity pages containing infoboxes and selected the most common attributes as target attributes. For each entity page with an infobox mentioning one or more target attributes, we segmented the document into sentences using word segmentation technology. Then for each target attribute, our method used the attribute value to search for the corresponding sentences in the article. Our implementation used two heuristics to match sentences to attributes as follows.

- If an attribute value is mentioned by one or more sentences in an article, we use these sentences and the attribute name to form a positive example.
- If there is no sentence containing the attribute value exactly, we use the word segmentation technology to split the value into several words and remove words that have exactly the same value as other attribute values. Then we find the sentences containing one or more of these words, finally form a positive example.

80% of the resulting labeled sentences were used to form the training set TR and the remaining 20% formed test set TE . We used TR and TE to train CNN classifier model after the sentences were processed by word segmentation technology.

We conducted our experiment on the category of people and selected six common attribute types—Date of Birth, Nationality, Birthplace, Profession, Graduate Institution and Major Achievement. These target attributes were used to construct datasets. We automatically obtained a large number of data that have already been labeled in the end. The following Table 1 shows a summary of the datasets.

Table 1. Training and test datasets.

Type	TR	TE
Date of Birth	13620	3431
Nationality	12210	3000
Birthplace	13062	3317
Profession	12302	3005
Graduate Institution	8048	2018
Major Achievement	7200	1774
Total	66442	16545

5.2 Setup of the Classifier

We trained and tested three text classifiers based on BAYES, SVM and CNN, following the grid-search procedure along with 10-fold cross validation to select the optimal parameters in the process of training. For the BAYES classifier, the parameter α was set to 1. For the SVM classifier, we used a RBF kernel, the parameter cost was set to

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

0.5 and the γ was set to 2. For the CNN classifier, the convolutional layer contained 48 convolution cores with a height of 2 and a width of the word vector. We used an average pooling to combine the local feature vectors to obtain a global feature vector. Then the vector was fed into a fully connected network with 50 neurons and 10% dropout.

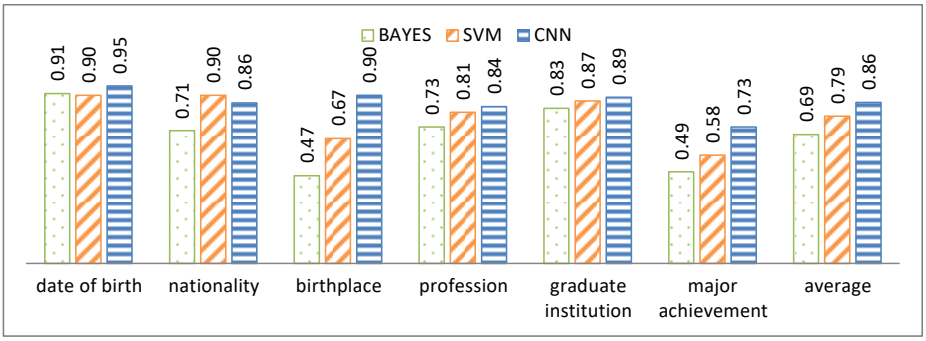

Fig. 4. Evaluation results of CNN, BAYES and SVM classifiers.

Fig. 4 shows the accuracy obtained while testing BAYES, SVM and CNN classifiers respectively. The results show that our classifier coupled with CNN outperforms the BAYES and SVM methods on most types. Especially, there is a significant accuracy increase in the types of birthplace and major achievement.

5.3 Evaluation of the method

We randomly selected 104 tables containing entities of the category of people from a large number of tables crawled from web pages. Each row of these tables contained an entity and several attribute values. Totally we got 2820 references for the six selected attributes. There were 126 references for Date of Birth, 833 references for Nationality, 353 references for Birthplace, 346 references for Profession, 149 references for Graduate Institution and 1013 references for Major Achievement. We manually annotated these tables for our method to compare with.

We firstly ran our method without post-processing operation on these web tables to evaluate the accuracy of our classifier in handling real tabular data, and then we used the post-processing operation to rule out cells that were annotated incorrectly. The experiments were compared with two baseline methods—BAYES and SVM.

Table 2 shows a comparison of experimental results with and without post-processing operation. For all the three methods—BAYES, SVM and CNN, the post-processing dramatically increases the accuracy of the classifiers. This proves that the method used in the post-processing phase can effectively remove incorrect annotations. We can also notice that the BAYES and SVM methods have low accuracy rates on the type major achievement and their accuracy rates are not improved even after using post-processing operations. The reason is that the majority vote does not work well in the post-processing phase based on the low accuracy results given by the clas-

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

sifiers. However, our method coupled with CNN still performs well on the type of major achievement and the post-processing also results in a significant improvement in accuracy.

Table 2. Evaluation of the algorithm with and without post-processing.

type	BAYES		SVM		CNN	
	BAYES	BAYES + Post-Proc	SVM	SVM+ Post-Proc	CNN	CNN+ Post-Proc
Date of Birth	0.83	0.85	0.77	0.77	0.91	1.00
Nationality	0.67	1.00	0.74	1.00	0.71	1.00
Birthplace	0.64	0.82	0.68	0.89	0.79	0.95
Profession	0.74	0.96	0.76	0.96	0.74	0.96
Graduate Institution	0.78	1.00	0.84	1.00	0.79	0.93
Major Achievement	0.21	0.20	0.34	0.32	0.76	0.96
average	0.65	0.81	0.69	0.82	0.78	0.97

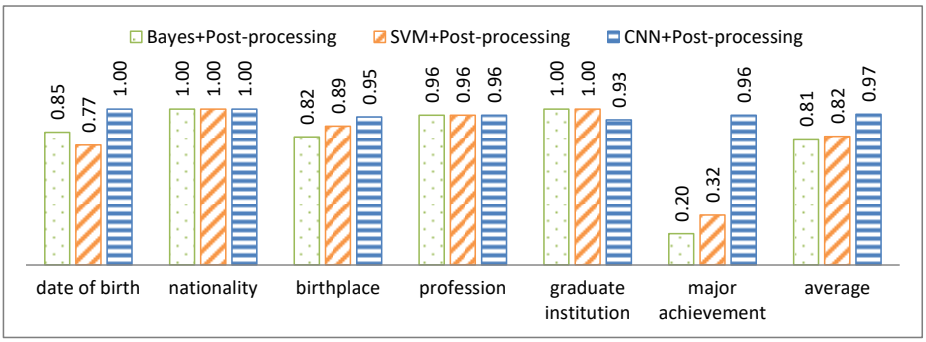

Fig. 5. Results of column concept determination in web tables (with Post-processing)

Fig. 5 shows a comparison of the accuracy between the three methods when using post-processing. Our method coupled with CNN shows its superiority as it outperforms the other two baseline methods on most types. The average accuracy improves by 15% compared to the best baseline. Moreover, we can notice that the accuracy rate on the type of major achievement has a great improvement (0.96 versus 0.32 and 0.20). This shows that the convolutional neural network can effectively capture the features for attribute values from their related text, especially for the ones which consist of long sentences and are hard to classify. We observe a slight drop in the type of graduate institution, but the average accuracy of our proposed method is higher over the other two methods totally.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

6 Conclusion

This paper described a method that determines the column concept for Chinese web tables using convolutional neural network. The proposed method can be used for the construction and expansion of the Chinese knowledge graph and can also contribute to applications such as data extraction and table searching. To the best of our knowledge, this is the first study of semantic recovery for Chinese Web tables. Another advantage which is different from the previous works is that our method is able to handle cells with long sentences. This means we can get a large amount of descriptive knowledge (e.g., major achievement) from the web tables to expand the knowledge bases. The evaluation shows our proposed method outperforms the BAYES and SVM methods and reaches an average accuracy of 97%.

For the future work, we intend to improve the scalability of our algorithm. Although our algorithm can automatically retrieve and select target attributes and training data from Baidu Encyclopedia, it is limited. We need a larger data source to get more attribute types and training data to annotate web tables.

7 Acknowledgement

This work was supported by the National Key R&D Program of China (No. 2017YFC0820700), the Fundamental theory and cutting edge technology Research Program of Institute of Information Engineering, CAS (Grant No.Y7Z0351101), Xinjiang Uygur Autonomous Region Science and Technology Project (No.2016A030007-4).

References

1. Michael J. Cafarella, Alon Y. Halevy, Daisy Zhe Wang, Eugene Wu, Yang Zhang: WebTables: exploring the power of tables on the web. PVLDB 1(1), 538-549 (2008).
2. Jingjing Wang, Haixun Wang, Zhongyuan Wang, Kenny Qili Zhu. Understanding Tables on the Web. ER 2012, 141-155 (2012).
3. Girija Limaye, Sunita Sarawagi, Soumen Chakrabarti: Annotating and Searching Web Tables Using Entities, Types and Relationships. PVLDB 3(1), 1338-1347 (2010).
4. Petros Venetis, Alon Y. Halevy, Jayant Madhavan, Marius Pasca, Warren Shen, Fei Wu, Gengxin Miao, Chung Wu: Recovering Semantics of Tables on the Web. PVLDB 4(9), 528-538 (2011).
5. G. Quercini and C. Reynaud: Entity discovery and annotation in tables. In EDBT, 2013, 693-704 (2013).
6. Oliver Lehmberg, Dominique Ritze, Robert Meusel, Christian Bizer: A Large Public Corpus of Web Tables containing Time and Context Metadata. WWW (Companion Volume) 2016, 75-76 (2013).
7. Dominique Ritze, Oliver Lehmberg, Christian Bizer: Matching HTML Tables to DBpedia. WIMS 2015, 10:1-10:6 (2015).
8. Tam, N. T.; Hung, N. Q. V.; Weidlich, M.; and Aberer, K: Result selection and summarization for web table search. In ICDE, 231–242 (2015).

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48

9. Wentao Wu, Hongsong Li, Haixun Wang, Kenny Qili Zhu:Probase: a probabilistic taxonomy for text understanding. SIGMOD Conference 2012, 481-492 (2012).
10. Dong Deng, Yu Jiang, Guoliang Li, Jian Li, Cong Yu: Scalable Column Concept Determination for Web Tables Using Large Knowledge Bases. PVLDB 6(13), 1606-1617 (2013).
11. Dominique Ritze, Christian Bizer: Matching Web Tables To DBpedia - A Feature Utility Study. EDBT 2017, 210-221 (2017).
12. Oktie Hassanzadeh, Michael J. Ward, Mariano Rodriguez-Muro, Kavitha Srinivas: Understanding a large corpus of web tables through matching with knowledge bases: an empirical study. OM 2015, 25-34 (2015).
13. Ziqi Zhang: Towards Efficient and Effective Semantic Table Interpretation. Semantic Web Conference (1) 2014, 487-502 (2014).
14. Fei Wu, Daniel S. Weld: Autonomously semantifying wikipedia. CIKM 2007, 41-50 (2007).
15. Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, Christian Bizer: DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. Semantic Web 6(2), 167-195 (2015).

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93713-7_48