Extracting Structured Data from Web Pages with Maximum Entropy Segmental Markov Model

Susan Mengel and Yaoquin Jing

Texas Tech University, Computer Science,
Box 43104, Lubbock, TX
susan.mengel@ttu.edu, jing.andy@gmail.com

Abstract. Automated techniques can help to extract information from the Web. A new semi-automatic approach based on the maximum entropy segmental Markov model, therefore, is proposed to extract structured data from Web pages. It is motivated by two ideas: modeling sequences embedding structured data instead of their context to reduce the number of training Web pages and preventing the generation of too specific or too general models from the training data. The experimental results show that this approach has better performance than Stalker when only one training Web page is provided.

Keywords: HTML extraction, Markov Model.

1 Introduction

As part of the Semantic Web effort, finding a way to extract structured data from Web pages and integrating the data with uniform schemes would assist Web users with common tasks, such as searching and product comparisons. Because the source file of a Web page consists of a sequence of content, such as structured data and tags, the problem addressed in this paper can be defined as follows: given the source file of a Web page, find the subsequences which contain structured data, and then extract these data from the original Web page and from similar Web pages.

2 Approaches for Structured Data Extraction

The approaches for extracting structured data from Web pages can be classified into three categories: manual, semi-automatic, and automatic. As the manual approaches [1] need users to write extraction rules in the special rule languages after investigating the characteristics of the Web pages embedding structured data, they are time-intensive and error-prone. Therefore, researchers currently focus on the approaches belonging to the latter two categories. Semi-automatic approaches [2], [3], [4] adopt machine learning technologies, such as instance-based learning or inductive learning, to generate extraction rules based on the

G. Vossen, D.D.E. Long, and J.X. Yu (Eds.): WISE 2009, LNCS 5802, pp. 219–226, 2009.

provided training examples, which contain extracted structured data labeled by users. One limitation of these approaches is that their performance depends on the coverage of the training Web pages for a set of Web pages embedding similar structured data; the more training Web pages, the better performance of the approaches. However, more training Web pages means more work for users. The automatic approaches [5], [6], [7] are based on the assumption that the similar structured data are embedded with similar sequences of tags and content on Web pages. Hence, structured data can be automatically found by searching the similar subsequence of a Web page without the users involvement. One limitation of these approaches is that they only process the Web pages containing at least two similar structured data. In addition, unexpected structured data can be extracted by these approaches and users still need to post-process them.

Learning extraction rules also exists in developing general purpose information extraction systems for natural language text using Markov models [8], [9], [10]. The model presented in this paper is based on these models.

3 Extracting Structured Data Using a Maximum Entropy Segmental Markov Model

The new approach is based on the following two initial ideas: unlike other semi-automatic approaches based on the inductive learning paradigm, which generate extraction rules based on the context of structured data (or data items) of a Web page, it constructs a model to describe the sequence (instead of its context) of embedded structured data (or data items). Secondly, as the semi-automatic approaches based on the instance-based learning paradigm usually generate templates by combining differences among the training data, the learned templates may be too general or specific to extract structured data from similar Web pages correctly. The cause of this problem is that extra assumptions are made from training data. For example, one training data contains a distinct symbol not existing in other ones; a learned template containing this symbol makes an assumption that all sequences embedding similar structured data contain this symbol. This problem can be solved by enforcing a model only describes the characteristics of training data without extra assumptions. To satisfy the requirements of the above two ideas, a maximum entropy segmental Markov model approach is proposed to extract structured data from Web pages.

3.1 Maximum Entropy

The principle of maximum entropy [11] is a framework to estimate the distribution of data. Its underlying principle is that the best model for data is the one satisfying certain constraints derived from training data with the fewest possible assumptions. More explicitly, given a set of training data, the probability distribution p should be consistent with the known evidence or partial information and maximize the entropy

$$H(a) = -\sum_{x \in \epsilon} p(x) \log p(x) \tag{1}$$

where ϵ is the event space, such as all sequences of similar Web pages embedding similar structured data.

To apply the maximum entropy principle to estimate the distribution of data, a critical step is the representation of the facts (or evidences) about data. For example, how to represent the known fact that student names embedded on Web pages are enclosed with $< table >$ and $< /table >$ tags. In most applications, this fact is represented with a binary function $f_i : \epsilon \rightarrow \{0, 1\}$, called a feature (function). For example, the feature function $f_{<table>}(x)$ denotes a sequence enclosed by the $< table >$ and $< /table >$ tags.

3.2 MESMM

To establish an maximum entropy segmental Markov model (MESMM) for sequences embedding structured data, users are asked to highlight one structured data on a Web page. The approach then automatically generates training data, which are used to generate a segmental Markov model (SMM). For each state in the SMM, feature functions are generated based on its training data. At the same time, the generalized iterative scaling (GIS) procedure [12] is used to learn the state transition distribution based on the maximum entropy principle. When a query Web page is submitted, the approach first determines if this Web page is similar to the training Web pages based on their Kullback-Leibler distance. If the Web page is similar, the corresponding MESMM model is applied to find the sequences embedding structured data and their optimal segmentations with the inference algorithm. If the query Web page is not similar to any model's training Web pages, users are required to highlight structured data and a new model is generated accordingly.

The maximum entropy segmental Markov model (MESMM) has a segment of observations instead of one observation for each state. Fig. 1 illustrates the graphical structure of the MESMM, where K is the length of a state sequence except s_0, which denotes a "start" state, and $1 \leq K \leq T$.

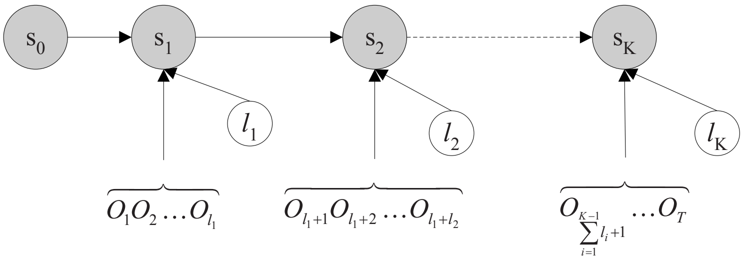
s_0
s_1
s_2
s_K
l_1
l_2
l_K
$O_1 O_2 \ldots O_{l_1}$
$O_{l_1+1} O_{l_1+2} \ldots O_{l_1+l_2}$
$O_{\sum_{i=1}^{K-1} l_i+1} \ldots O_T$

Fig. 1. The graphical structure of the MESMM

The pseudocode of the inference algorithm is shown in Figure 2 and is similar to the Viterbi algorithm [13], but differs in that it also stores the length of the current observation segment. The algorithm first creates two matrices, δ and φ, with the size $T \times |M|$ (lines 2-3), where T is the length of an observation sequence and $|M|$ is the number of distinct states in the MESMM. Note that each state is indexed with a number between zero and $|M|-1$. Furthermore, the "start" state is assigned with the index zero. bGiven a position t in an observation sequence, the algorithm finds the maximal value of δ_t for each possible current state by enumerating segments with the length from one to t (lines 13-29). This step runs iteratively for each position in the sequence. The time complexity of this algorithm is $O(|M|^2T^2)$ and the space complexity is $O(|M|T)$. Further details of the operation of the MESMM may be found in [14].

```
Algorithm OptimalSegmentation (In: $O$, $Q$; Out: $\varphi$)
// $O$ is an observation sequence; $Q$ is the MESMM. $\varphi$ is the matrix storing related values.
{
1.      T=O's length;
// $|M|$ the number of distinct states in $Q$.
2.      Create the matrix $\delta$ with the size $T \times |M|$;
3.      Create the matrix $\varphi$ with the size $T \times |M|$;
// A "start" state is indexed with '0'.
4-6.    $\delta[0,0] = 1$; $\varphi[0,0].Length = 0$; $\varphi[0,0].Prev = 0$;
7.      For $i = 0$ To $|M| + 1$
8-10.       $\delta[0,i] = 0$; $\varphi[0,i].Length = 0$; $\varphi[0,i].Prev = 0$;
11.     EndFor
12.     For $t = 1$ To $T$
13.             For $i = 1$ To $|M|$
14.             $v = 0$; //store the maximum probability;
15.             $s = 0$; //store the previous state;
16.             $n = 0$; //store the length of a segment;
17.             For $n = i$ To 1
18.                 For $j = 1$ To $|M|$
19.                     If $\delta[t-l,j] \times p(s=i|s'=j, O_{t-l+1}O_{t-2+2}\dots O_t) > v$ then
20-22.                      $v = \delta[t-l,j] \times p(s=i|s'=j, O_{t-l+1}O_{t-2+2}\dots O_t)$; $s = j$; $n = l$;
23.                     EndIf
24.                 EndFor
25.             EndFor
26-28.          $\delta[t,i] = v$; $\varphi[t,i].Length = n$; $\varphi[t,i].Prev = s$;
29.         EndFor
30.     EndFor
}
```

Fig. 2. The inference algorithm for the MESMM

4 Experiments

One critical issue in designing experiments is to choose an appropriate approach to compare with the MESMM approach. The semi-automatic approaches definitely have better performance than the manual or automatic approaches, so it is meaningless to select a manual or automatic approach to be compared with the MESMM approach. Among all semi-automatic approaches, Stalker has the best performance and its idea has been adopted by the commercial product, Fetch [15]. Hence, Stalker is selected to be compared with the MESMM approach on performance.

Another issue is the collection of experimental data. As no standard experimental data is available in this field, each approach constructs its data by directly selecting Web pages on the Internet. To avoid the bias of some special Web pages on performance, experimental data is constructed by selecting Web pages from those Web sites used in published papers. There are two requirements on selecting Web sites. One is that a Web site still exists on the Internet; another is that the MESMM approach requires at least two similar Web pages, one for learning and another for extracting, to evaluate its performance.

The principal goal of the MESMM approach is to keep the high accuracy with fewer training web pages. To achieve this goal, the experiment is performed in the following way: for each web site, one web page is selected as the training example, which is used to generate the MESMM model and to learn the corresponding state transition distributions. Then, the model is used to extract structured data from the remaining similar web pages.

4.1 Evaluation Metric

Three types of errors exist for an approach to extract structured data from a web page. The first type, denoted as m, is missed expected data items; the second type, denoted as w, is wrong expected data items; the third type, denoted as e, is extra (or unexpected) data items (note that errors are represented with data items instead of structured data). Based on three types of errors, a metric, error rate, is proposed to measure the performance of an approach extracting structured data from a Web page.

$$r = \frac{n_u}{N_E} \tag{2}$$

where N_e is the number of expected data items to be extracted from a Web page; n_u is the number of erroneous items extracted from a Web page and $n_u = m + w + e$. Besides the error rate, another metric is the precision of an approach which may be derived as one minus the error rate.

4.2 Results

The experimental results are listed in Tab. 1. While more pages were used (30 Web sites each with 4 or 5 similar pages), due to space limitations, the data shown includes all Web pages where the MESMM approach had errors or where Stalker had an average error of .5 or above. Most of the Web pages had four to 20 pieces of structured data each with 2 or 3 items (www.ubids.com had upwards of 662 pieces of structured data). The MH column indicates the approach utilized to extract structured data from Web pages, where the Stalker and the MESMM approaches are denoted with S and M, respectively. The error rate is calculated according to Eq. 2. The average error rate of a Web site is calculated based on the error rates of its similar Web pages. From the table below, the overall average error rate for Stalker is 0.64 (0.37 with all 30 Websites) and for the MESMM, 0.14 (0.08 with all 30 Websites).

The performance difference between Stalker and the MESMM approach can be justified by investigating their underlying mechanisms. Stalker uses context symbols to discover the boundaries of sequences embedding structured data or data items. The MESMM approach, however, takes sequences themselves to determine if they embed expected structured data. This way is consistent with the basic assumption on extraction systems that similar structured data are embedded on similar sequences. Secondly, Stalker chooses common symbols occurring in the contexts of embedding sequences to locate structured data. However, it is very difficult to find those common symbols with fewer training examples. In this situation, Stalker would consider the symbols specific to the training web page as common symbols. As a result, the generated extraction rules are too specific to extract structured data from similar web pages. In the MESMM approach,

Table 1. Experimental Result Error

Website	**MH**	**Page 1**	**Page 2**	**Page 3**	**Page 4**	**Avg**
www.asiatravel.com	S	0.0	0.0	0.0	0.0	0.0
	M	1.0	1.0	1.0	1.0	1.0
www.barnesnoble.com	S	1.0	1.0	1.0	1.0	1.0
	M	0.0	0.0	0.0	0.0	0.0
www.bestbuy.com	S	0.11	0.38	0.89	0.78	0.54
	M	0.01	0.02	0.11	0.0	0.04
www.borders.com	S	0.5	0.67	0.57	0.57	0.58
	M	0.0	0.0	0.0	0.0	0.0
www.compusa.com	S	0.97	0.93	0.93	0.88	0.93
	M	0.0	0.0	0.0	0.0	0.0
www.coolhits.com	S	0.0	0.0	1.0	1.0	0.5
	M	0.0	0.0	0.0	0.0	0.0
www.epicurious.com	S	1.0	1.0	0.25	1.0	0.81
	M	0.0	1.0	0.0	0.5	0.38
www.etoys.com	S	1.0	1.0	1.0	1.0	1.0
	M	0.0	0.0	0.0	0.0	0.0
www.flipdog.com	S	0.0	1.0	0.0	0.05	0.26
	M	0.0	0.0	0.0	0.0	0.0
www.grijins.com	S	1.0	1.0	1.0	1.0	1.0
	M	0.0	0.0	0.0	0.0	0.0
www.newegg.com	S	1.0	1.0	1.0	1.0	1.0
	M	0.0	0.0	0.0	0.0	0.0
www.overstock.com	S	0.07	0.0	0.08	1.0	0.29
	M	0.0	0.0	0.0	1.0	0.25
www.qualityinks.com	S	0.0	0.0	0.0	0.0	0.0
	M	0.0	1.0	0.0	0.0	0.25
www.radioshark.com	S	0.33	0.38	0.38	1.0	0.52
	M	0.0	0.0	0.0	1.0	0.25
www.scistore.cambridgesoft.com	S	0.75	0.74	0.74	74.0	0.74
	M	0.0	0.0	0.0	0.0	0.0
www.ubids.com	S	1.0	1.0	1.0	1.0	1.0
	M	0.02	0.12	0.07	0.06	0.07

the model takes all features of an embedding sequence into account. Since most of features of similar sequences are approximately identical, specific features of the training web page have minor impact on determining the similarity of a sequence.

One weakness of the MESMM approach compared with Stalker is the time complexity of its inference algorithm. Given a sequence with the length n, Stalker takes almost linear time $O(n)$ to locate a subsequence with extraction rules. However, the MESMM approach takes $O(Cn^2)$ time to determine if a sequence is similar, where C is the number of states of the model.

5 Conclusions and Future Work

The primary contribution of this paper is a new semi-automatic approach for extracting structured data from Web pages, which maintains good performance with fewer training Web pages. The experimental results demonstrate that this approach has far better performance than Stalker when there is only one training Web page. However, there still exists room to improve the MESMM approach and extend its application. The improvement to the inference algorithm becomes very crucial when the approach is applied to process very long sequences. One solution is that the inference algorithm takes a chunk of contiguous symbols with the same state instead of taking each single symbol one time at a time. For example, if a tag node is assigned with a state, all its sub-nodes are assigned with the same state. The current implementation of the MESMM approach needs to be extended to process Web pages where data items from structured data are interleaved.

References

1. Feldman, R., Aumann, Y., Finkelstein-Landau, M., Hurvitz, E., Regev, Y., Yaroshevich, A.: A Comparative Study of Information Extraction Strategies. In: Gelbukh, A. (ed.) CICLing 2002. LNCS, vol. 2276, pp. 349–359. Springer, Heidelberg (2002)
2. Chang, C.-H., Kuo, S.-C.: OLERA: Semisupervised Web Data Extracion with Visual Support. IEEE Intelligent Systems 4(6), 56–64 (2004)
3. Zhai, Y., Liu, B.: Extracting Web Data Using Instance-Based Learning. In: Proceedings of 6th International Conference on Web Information System Engineering (2005)
4. Hogue, A., Karger, D.: Thresher: Automating the Unwrapping of Semantic Content from the World Wide Web (2005)
5. Lemma, K., Getoor, L., Minton, S., Knoblock, C.: Using the Structure of Web Sites for Automatic Segmentation of Tables. In: Proceedings of the 2004 ACM SIGMOD International Conference on Management of data, pp. 119–130 (2004)
6. Zhai, Y., Liu, B.: Web Data Extraction Based on Partial Tree alignment. In: Proceedings of the 14th International World Wide Web in Chiba, Japan (2005)
7. Liu, B., Zhai, Y.: NET- A System for Extracting Web Data from Flat and Nested Data Records. In: Proceedings of 6th International Conference on Web Information Systems Engineering (2005)

8. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE 77(2), 257–285 (1989)
9. McCallum, A., Freitag, D., Pereira, F.: Maximum Entropy Markov Models for Information Extraction and Segmentation. In: Proceedings ICML 2000, pp. 591–598 (2000)
10. Ge, X.: Segemental Semi-Markov Models and Applications to Sequence Analysis. PhD. Thesis, University of California, Irvine (2002)
11. Good, I.J.: Maximum Entropy for Hypothesis Formulation, Especially for Multidimensional Contingency Tables. The Annals of Mathematical Statistics 34, 911–934 (1963)
12. Darroch, J.N., Ratcliff, D.: Generalized Iterative Scaling for Log-Linear Models. The Annals of Mathematical Statistics 43(5), 1470–1480 (1972)
13. Viterbi, A.J.: Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm. IEEE Transactions on Information Theory IT-13, 260–269 (1967)
14. Jing, Y.: Extracting Structured Data from Web Pages with Maximum Entropy Segmental Markov Models. Texas Tech University, Computer Science, Doctoral Dissertation (2007)
15. Fetch Technologies, Inc. (2009), `http://www.fetch.com`