A Bottom Up Discovery of Generalized Web Browsing Patterns

WEI-SHUO LO
Institute of Information
Engineering I-Shou University
Kaohsiung, Taiwan 840, ROC
Dept. of Business Administration
Mei-Ho Institute of Technology
Pingtun, Taiwan 912, ROC

SHYUE-LIANG WANG
Dept. of Computer Science
New York Institute of Technology
New York 10023, USA

TZUNG-PEI HONG
Dept. of Electrical Engineering
National University of Kaohsiung
Taiwan 811, ROC

Abstract: - Web mining of browsing patterns including simple sequential patterns and sequential patterns with browsing times has been studied recently. However, most of these works focus on mining browsing patterns of web pages directly. In this work, we introduce the problem of mining generalized browsing patterns on cross-levels of a taxonomy comprised of web pages. An algorithm called based on bottom up approach is proposed to discover these cross level generalized browsing patterns. Example demonstrating the proposed approach is given. Comparison with the Generalized Sequential Patterns (GSP) [16] and Apriori-Level algorithms show that our approach generates fewer candidate sequences. The proposed algorithm thus promotes the efficiency of discovering coarser granularity of web browsing patterns.

Keywords- Web Mining, Browsing Patterns, Generalized Sequential Patterns, Cross-Level Sequential Patterns

1. Introduction

Web mining can be viewed as the use of data mining techniques to automatically retrieve, extract and evaluate information for knowledge discovery from web documents and services [11]. It has been studied extensively in recent years due to practical applications of extracting useful knowledge from inhomogeneous data sources in the World Wide Web. Web mining can be divided into three classes: web content mining, web structure mining and web usage mining [7].

Web content mining focuses on the discovery of useful information from the web contents, data and documents. Web structure mining deals with mining the structure of hyperlinks within the web itself. Web usage mining emphasizes on the discovery of user access patterns from secondary data generated by users' interaction with the web. In the past, several web mining approaches for finding user access patterns and user interesting information from the World Wide Web were proposed [3-6,15].

Web browsing pattern is a kind of user access pattern that considers users' browsing sequences of web pages. In fact, it is similar to the discovery of sequential patterns from transaction databases. The problem of mining sequential patterns was introduced in [1]. Let $I=\{i_1, i_2, \ldots, i_m\}$ be a set of literals, called *items*. An *itemset* is a non-empty unordered set of items. A *sequence* is an ordered list of itemsets. An itemset *i* is denoted as $(i_1, i_2, \ldots, i_m)$, where i_j is an item. A sequence *s* is denoted as $(s_1 \rightarrow s_2 \rightarrow \ldots s_q)$, where s_j is an itemset. Given a database *D* of customer transactions, each transaction *T* consists of fields: customer-id, transaction-time, and the items purchased in the transaction. All the transactions of a customer can together be viewed as a sequence. This sequence is called a *customer-sequence*. A customer *supports* a sequence *s* if *s* is contained in the customer-sequence for this customer. The *support* for a sequence is the fraction of total customers who support this sequence. A sequence with support greater than a user-specified minimum support is called a large sequence. In a set of sequences, a sequence is maximal if it is not contained in any other sequences. The problem of finding sequential patterns is to find the maximal sequences among all sequences that have supports greater than a certain user-specified minimum support.

Many efficient algorithms for discovering maximal sequential patterns have been proposed

[1,2,12,14,16,17,18,19,20]. In application to web browsing patterns, techniques for mining simple sequential browsing patterns and sequential patterns with browsing times have been proposed [4,5,6,7,10,11,15]. However, most of these works focus on mining browsing patterns of web pages directly. In this work, we introduce the problem of mining generalized browsing patterns on cross-levels of a taxonomy comprised of web pages. An algorithm called Apriori-Level is proposed to discover these cross-level browsing patterns.. The proposed algorithm can discover cross-level relevant browsing behavior from log data and promote the discovery of coarser granularity of web browsing patterns.

The rest of our paper is organized as follows. Section 2 presents the mining algorithm of cross-level generalized browsing patterns. Section 3 gives an example to illustrate the feasibility of the proposed algorithm and shows the difference between our algorithm, Apriori-Leve, and the GSP algorithm. A conclusion is given at the end of the paper.

2. Mining of the cross-level browsing patterns

In this section, we describe the proposed bottom up Apriori-Level data mining algorithm to discover cross-level generalized browsing patterns from log data.

2.1 Notation

The following notation is used in our proposed algorithm:

n: the total number of log data;
m: the total number of files in the log data;
c: the total number of clients in the log data;
n_i: the number of log data from the i-th client, $1 \le i \le c$;
D_i: the browsing sequence of the i-th client, $1 \le i \le c$;
D_{id}: the d-th log data in D_i, $1 \le d \le n_i$;
I^g : the g-th file, $1 \le g \le m$;
α: the predefined minimum support value;
C_r: the set of candidate sequences with r files;
L_r : the set of large sequences with r files.

2.2 The Bottom up Algorithm

The proposed bottom up algorithm, first finds all large one itemsets on each level of the concept hierarchy, from bottom to top. These itemsets are actually the large sequences of size one, which are also called large 1-sequences. Based on these large 1-sequences, it generates the candidate 2-sequences and calculates their supports in order to determine the large 2-sequences. This step is repeated until no candidate sequences can be generated. The maximal sequences can be obtained by eliminating the subsequences in the set of large sequences found. Steps of the proposed mining algorithm are described below.

INPUT: A server log, a predefined taxonomy of web pages, a predefined minimum support value α.
OUTPUT: A set of cross-level maximal browsing patterns

STEP 1: Select the web pages with file names including .asp, .htm, .html, .jva .cgi and closing connection from the log data; keep only the fields *date, time, client-ip* and *file-name*. Denote the resulting log data as *D*.

STEP 2: Encode each web page file name using a sequence of number and the symbol "*", with the *t*-th number representing the branch number of a certain web page on level *t*.

STEP 3: Form a browsing sequence D_j for each client c_j by sequentially listing his/her n_j tuples (web page), where n_j is the number of web page browsed by client c_j. Denote the *d-th* tuple in D_j *as* D_{jd}.

STEP 4: Find the set of large itemsets with respect to at the bottom level of concept hierarchies, then each items of the bottom can generate their ancestors from bottom to top. The large itemsets in each level are in fact the large 1-sequences for that level. The set of all large 1-sequences from all level is denoted as LS_1.

STEP 5: Generate candidate k-sequences by joining LS_{k-1} and LS_{k-1}. Those k-sequences with support counts greater than the pre-specified minimum support threshold will be the large k-sequences. This step ends when no more candidate sequence can be generated.

STEP 6: Find the maximal sequences by deleting those sequences that are subsequences of others.

3. An example

In this section, we describe an example to demonstrate the proposed mining algorithm of cross-level generalized web browsing patterns. The example shows how the proposed algorithm can be used to discover the cross-level sequential patterns from the web browsing log data shown in Table 1. In addition, the predefined taxonomy for web pages is shown in Figure 1. The predefined minimum support α is set at 50 %. The proposed data mining algorithm proceeds as follows.

Table 1: A part of log data used in the example

Date	Time	Client-IP	File-name
2002-06-01	05:39:56	140.117.72.1	News.htm
2002-06-01	05:40:08	140.117.72.1	Exective.htm
2002-06-01	05:40:10	140.117.72.1	Stock.htm
……..	……..	……..	……..
2002-06-01	05:40:26	140.117.72.1	University.htm
……..	……..	……..	……..
2002-06-01	05:40:52	140.117.72.2	Golf.htm
2002-06-01	05:40:53	140.117.72.2	Stock.htm
……..	……..	……..	……..
2002-06-01	05:41:08	140.117.72.3	Golf.htm
……..	……..	……..	……..
2002-06-01	05:48:38	140.117.72.4	Closing connection
……..	……..	……..	……..
2002-06-01	05:48:53	140.117.72.5	Golf.htm
……..	……..	……..	……..
2002-06-01	05:50:13	140.117.72.6	Stock.htm
……..	……..	……..	……..
2002-06-01	05:53:33	140.117.72.6	Closing connection

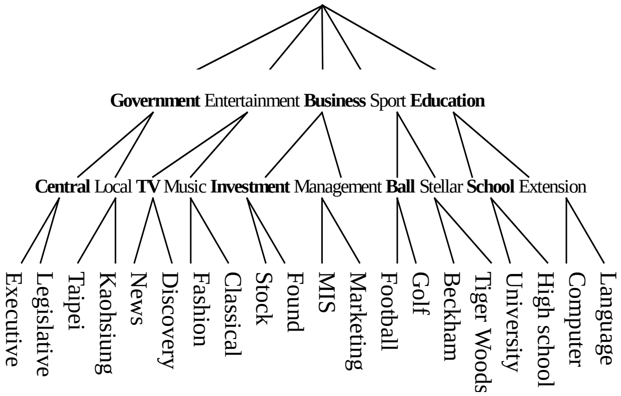

Figure 1: The predefined taxonomy used in this example

STEP 1: Select the web pages with file names including .asp, .htm, .html, .jva .cgi and closing connection from Table 1. Keep only the fields *date*, *time*, *client-ip* and *file-name*. Denote the resulting log data as Table 2.

Table 2: The resulting log data for web mining

Client-IP	File-name
140.117.72.1	News.htm
140.117.72.1	Exective.htm
140.117.72.1	Stock.htm
……..	……..
140.117.72.1	University.htm
140.117.72.2	Golf.htm
140.117.72.2	Stock.htm
140.117.72.3	Golf.htm
140.117.72.4	Closing connection
140.117.72.5	Golf.htm
……..	……..
140.117.72.6	Stock.htm
……..	……..
140.117.72.6	Closing connection

STEP 2: Each file name is encoded using the predefined taxonomy shown in Figure 1. Results are show in Table 3.

Table 3: Codes of file names

Code	File name	Code	File name
111	Executive.htm	1**	Government.htm
211	News.htm	2**	Entertainment.htm
212	Discovery.htm	3**	Business.htm
313	Stock.htm	4**	Sport.htm
321	Found.htm	5**	Education.htm
411	Football.htm	11*	Central.htm
412	Golf.htm	12*	Local.htm
421	Tiger Woods.htm	21*	TV.htm
511	University.htm	22*	Music.htm
512	Highshool.htm	31*	Investment.htm
		32*	Management.htm
		41*	Ball.htm
		42*	Stellar.htm
		51*	School.htm
		52*	Extension.htm

STEP 3: The web pages browsed by each client are listed as a browsing sequence. Each tuple is represented as (web page), as shown in Table 4. The resulting browsing sequences from Table 4 are shown in Table 5

Table 4: The web pages browsed with their duration

Client-ID	(web page)
1	(111)
1	(211)
1	(231)
2	(112)
2	(111)
2	(231)

3	(111)
3	(211)
4	(211)
4	(313)
4	(323)
4	(421)
4	(534)

Table 5: The browsing sequences formed from Table4

Client ID	Browsing sequences
1	(111), (211), (231), (222)
2	(112), (111), (231)
3	(111), (211)
4	(211), (313), (323), (421), (534)

STEP 4: Find the set of large itemsets at the bottom level of concept hierarchies, then each items of the bottom can generate their ancestors from bottom to top. The large itemsets in each level are in fact the large 1-sequences for that level. The set of all large 1-sequences from all level is denoted as LS_1.

Table 6: Candidate and Large 1-sequence at bottom level

Itemsets	Count
111	3 ▲
112	1
211	3 ▲
222	1
231	2 ▲
313	1
323	1
421	1
534	1

▲ Large

Table 7: Generation all Large 1-sequence at each level

	Level 3	Level 2	Level 1
Itemsets	111	11*	1**
	211	21*	2**
	231	23*	

STEP 5: Generate candidate k-sequences by joining LS_{k-1} and LS_{k-1}. Those k-sequences with support counts greater than the pre-specified minimum support threshold will be the large k-sequences. This step ends when no more candidate sequence can be generated.

Table 8: Candidates and Large 2-sequence at cross-level

	1**	2**	11*	21*	23*	111	211	231
1**	0	**3▲**	0	**2▲**	**2▲**	0	**2▲**	**2▲**
2**	0	1	0	0	1	0	0	1
11*	0	**3▲**	0	**2▲**	**2▲**	0	**2▲**	**2▲**
21*	0	1	0	0	1	0	0	1
23*	0	0	0	0	0	0	0	0
111	0	**3▲**	0	**2▲**	**2▲**	0	**2▲**	**2▲**
211	0	1	0	0	1	0	0	1
231	0	0	0	0	0	0	0	0

STEP 6: Find the maximal sequences by deleting those sequences that are subsequences of others.

Large-1-sequence：
<(1**)>, <(2**)>, <(11*)>,
<(21*)>,<(23*)>, <(111)>,
<(211)>, <(231)>

Large-2-sequence：
<(1**), (2**)>, <(1**), (21*)>,
<(1**), (23*)>, <(1**), (211)>,
<(1**), (231)>, <(11*), (2**)>,
<(11*), (21*)>, <(11*), (23*)>,
<(11*), (211)>, <(11*), (231)>,
<(111), (2**)>, <(111), (21*)>,
<(111), (23*)>, <(111), (211)>,
<(111), (231)>

In this following, we compare our algorithm with the GSP approach and our previous Apriori-Leve approaches. Table 9 shows the candidate and large sequences generated by the three approaches, using the example in this section. It can be seen that the bottom up Apriori-Level approach generates fewer candidate itemsets at the 1-sequence phase.

Table 9: Comparison of generated sequences

K-sequence		GSP	Apriori-Level	Bottom up
1	Candidate itemset	22	16	9
	Large itemset	8	8	8
2	Candidate sequence	64	64	64
	Large sequence	15	15	15
3	Candidate sequence	60	60	60
	Large sequence	0	0	0

4. Conclusion

In this work, we have proposed a novel web-mining algorithm that can process web server logs to discover cross-level generalized web browsing patterns. The inclusion of concept hierarchy (taxonomy) of web pages produces browsing patterns of different granularity. This allows the views of users' browsing behavior from various levels of perspectives. In addition, we show that the proposed bottom up approach generates fewer candidate itemsets compared to the GSP and Apriori-Leve approaches. However, more numerical simulations need to be carried out in order to justify the efficiency of the proposed approach.

5. Acknowledgement

This work was partially supported by National Science Council of the Republic of China, under grant number NSC-91-2213-E-214-019.

References:

[1] R. Agrawal, and R. Srikant, "Mining Sequential Patterns", Proc. of the 11th International Conference on Data Engineering, 1995, pp. 3-14.

[2] N. Chen, A, Chen, "Discovery of Multiple-Level Sequential Patterns from Large Database", Proc. of the International Symposium on Future Software Technology, Nanjing, China, 1999, pp. 169-174.

[3] M.S. Chen, J.S. Park and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns", IEEE Transactions on Knowledge and Data Engineering, Vol. 10, 1998, pp. 209-221.

[4] L. Chen, K. Sycara, "WebMate: A Personal Agent for Browsing and Searching," The Second International Conference on Autonomous Agents, ACM, 1998.

[5] E. Cohen, B. Krishnamurthy and J. Rexford, "Efficient Algorithms for Predicting Requests to Web Servers, The Eighteenth IEEE Annual Joint Conference on Computer and Communications Societies, Vol. 1, 1999, pp. 284-293.

[6] R. Cooley, B. Mobasher and J. Srivastava, "Grouping Web Page References into Transactions for Mining World Wide Web Browsing Patterns ", Knowledge and Data Engineering Exchange Workshop, 1997, pp. 2-9.

[7] R. Cosala, H. Blockleel, "Web Mining Research: A Survey", ACM SIGKDD, Vol. 2, Issue 1, 2000, pp. 1-15.

[8] H. Mannila and H. Toivonen, "Discovering Generalized Episodes Using Minimal Occurrences", Proc. of the 2nd International Conference on Knowledge Discovery and Data Mining, 1996, pp. 146-151.

[9] T. Oates, et al, "A Family of Algorithms for Finding Temporal Structure in Data", Proc. of the 6th International Workshop on AI and Statistics, Mar 1997, pp. 371-378.

[10] T.P. Hong, K.Y. Lin, and S.L. Wang, "Web Mining for Browsing Patterns", Proc. of the 5th International Conference on Knowledge-based Intelligent Information Engineering Systems, Osaka, Japan, September 2001, pp. 495-499.

[11] S.K. Pal, V. Talwar, and P. Mitra, "Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Directions", to appear in IEEE Transactions Neural Network, 2002.

[12] J. Pei, J.W. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M.C. Hsu, "Prefixspan: Mining Sequential Patterns by Prefix-Projected Growth", Proc. of the 17th IEEE International Conference on Data Engineering, Heidelberg, Germany, April 2001.

[13] G. Piategsky-Shapiro, "Discovery, Analysis and Presentation of Strong Rules", Knowledge Discovery in Databases, AAAI/MIT press, 1991, pp. 229-248.

[14] M. Spliliopoulou, L.C. Faulstich, "WUM: A Web Utilization miner", Workshop on the Web and Data Base (WEBKDD), 1998, pp. 109-115.

[15] H. Pinto, "Multiple-Dimensional Sequential Patterns Mining", University of Lethebridge, Alberta, Canada, Master Thesis, April, 2001.

[16] R. Srikant and R. Agrawal, "Mining Sequential Patterns: Generalizations and Performance Improvements", Proc. of the 5th International Conference on Extending Database Technology, March 1996, pp. 3-17.

[17] S.L. Wang, C.Y. Kuo, T.P. Hong, "Mining Similar Sequential Patterns from Transaction Databases", Proc. of the 9th National Conference on Fuzzy Theory and Its Application, Taiwan, November 2001, pp. 624-628.

[18] S.L. Wang, W.S. Lo, T.P. Hong, "Discovery of Cross-Level Sequential Patterns from Transaction Databases", Proceedings of the 6th International Conference on Knowledge-based Intelligent Information Engineering Systems, September 2002, pp. 683-687.

[19] M.J. Zaki, "Efficient Enumeration of Frequent Sequences", Proc. of the 7th International Conference on Information and knowledge Management, Washington DC, Nov 1998, pp. 68-75.

[20] M. Zhang, B. Kao, C.L. Yip, and D. Cheung, "A GSP-Based Efficient Algorithm for Mining Frequent Sequences", Proc. of the IC-AI'2001, Las Vegas, Nevada, June 2001.