ABSTRACT

Web mining is one of the mining technologies, which applies data mining techniques in large amounts of Web data to improve the Web services. Web traversal pattern mining discovers most of the users’ access patterns from Web logs. This information can provide the navigation suggestions for Web users such that appropriate actions can be adopted. However, the Web data will grow rapidly in the short time, and some of the Web data may be antiquated. The user behaviors may be changed when the new Web data is inserted into and the old Web data is deleted from Web logs. Besides, it is considerably difficult to select a perfect minimum support threshold during the mining process to find the interesting rules. Even the experienced experts also cannot determine the appropriate minimum support. Thus, we must constantly adjust the minimum support until the satisfactory mining results can be found. The essences of incremental or interactive data mining are that we can use the previous mining results to reduce the unnecessary processes when the minimum support is changed or Web logs are updated. In this chapter, we propose efficient incremental and interactive data mining algorithms to discover Web traversal patterns and make the mining results to satisfy the users’ requirements. The experimental results show that our algorithms are more efficient than the other approaches.
INTRODUCTION

With the trend of the information technology, huge amounts of data would be easily produced and collected from the electronic commerce environment every day. It causes the Web data in the database to grow up at an amazing speed. Hence, how should we obtain the useful information and knowledge efficiently based on the huge amounts of Web data has already been the important issue at present.

Web mining (Chen, Park, & Yu, 1998; Chen, Huang, & Lin, 1999; Cooley, Mobasher, & Srivastava, 1997; EL-Sayed, Ruiz, & Rundensteiner, 2004; Lee, Yen, Tu, & Hsieh, 2003, 2004; Pei, Han, Mortazavi-Asl, & Zhu, 2000; Yen, 2003; Yen & Lee, 2006) refers to extracting useful information and knowledge from Web data, which applies data mining techniques (Chen, 2005; Ngan, 2005; Xiao, 2005) in large amount of Web data to improve the Web services. Mining Web traversal patterns (Lee et al., 2003, 2004; Yen, 2003) is to discover most of users’ access patterns from Web logs. These patterns can not only be used to improve the Web site design (e.g., provide efficient access between highly correlated objects, and better authoring design for Web pages, etc.), but also be able to lead to better marketing decisions (e.g., putting advertisements in proper places, better customer classification, and behavior analysis, etc.)

In the following, we describe the definitions about Web traversal patterns: Let \( I = \{x_1, x_2, ..., x_l\} \) be a set of all Web pages in a Web site. A traversal sequence \( S = <w_1, w_2, ..., w_m> \) \((w_i \in I, 1 \leq i \leq m)\) is a list of Web pages, which is ordered by traversal time, and each Web page can repeatedly appear in a traversal sequence, that is, backward references are also included in a traversal sequence. For example, if there is a path which visits Web page \(<B>\), and then go to Web page \(<G>\) and \(<A>\) sequentially, and come back to Web page \(<B>\), and then visit Web page \(<C>\). The sequence \(<B>GABC>\) is a traversal sequence. The length of a traversal sequence \( S \) is the total number of Web pages in \( S \). A traversal sequence with length \( l \) is called an \( l \)-traversal sequence. For example, if there is a traversal sequence \( \alpha = <AGDFAB> \), the length of \( \alpha \) is 6 and we call \( \alpha \) a 6-traversal sequence. Suppose that there are two traversal sequences \( \alpha = <a_1, a_2, ..., a_m> \) and \( \beta = <b_1, b_2, ..., b_n> \) \((m \leq n)\), if there exists \( i_1 < i_2 < ... < i_m \), such that \( b_{i_1} = a_{i_2} = a_{i_3} = ... = a_{i_m} \), then \( \beta \) contains \( \alpha \). \( \alpha \) is a sub-sequence of \( \beta \), and \( \beta \) is a super-sequence of \( \alpha \). For instance, if there are two traversal sequences \( \alpha = <BEA> \) and \( \beta = <ABCEA> \), then \( \alpha \) is a sub-sequence of \( \beta \) and \( \beta \) is a super-sequence of \( \alpha \).

A traversal sequence database \( D \), as shown in Table 1, contains a set of records. Each record includes traversal identifier (TID) and a user sequence. A user sequence is a traversal sequence, which stands for a complete browsing behavior by a user. The support for a traversal sequence \( \alpha \) is the ratio of user sequences, which contains \( \alpha \) to the total number of user sequences in \( D \). It is usually denoted as Support (\( \alpha \)). The support count of \( \alpha \) is the number of user sequences which contain \( \alpha \). For a traversal sequence \( <x_1, x_2, ..., x_l> \), if there is a link from \( x_i \) to \( x_{i+1} \) \((for all i, 1 \leq i \leq l-1)\) in the Web site structure, then the traversal sequence is a qualified traversal sequence. A traversal sequence \( \alpha \) is a Web traversal pattern if \( \alpha \) is a qualified traversal sequence and Support (\( \alpha \)) \( \geq \) min_sup, in which the min_sup is the user specified minimum support threshold. For instance, in Table 1, if we set min_sup to 80%, then Support (\( <AB> \)) = 4/5 = 80% \( \geq \) min_sup = 80%, and there is a link from “A” to “B” in the Web site structure shown in Figure 1. Hence, \( <AB> \) is a Web traversal pattern. If the length of a Web traversal pattern is \( l \), then it can be called an \( l \)-Web traversal pattern.

However, the user sequences will grow rapidly and some of the user sequences may be antiquated. The Web traversal patterns may be changed when the new user sequences are inserted into and the old user sequences are deleted from traversal sequence database. Therefore, we must re-discover the Web traversal patterns from the updated database. For example, if a new movie “Star Wars” is coming, in