REVIEW ON FREQUENT SEQUENTIAL PATTERNS FOR WEB RECOMMENDATION SYSTEM

Author: Sampada A.Khorgade[1]; Praful B.Sambhare[2]

Affiliation: M.E IInd year (CSE), P.R.Pote COET, Amravati, Maharashtra, India[1];

Assistant Professor, P.R.Pote COET, Amravati, Maharashtra, India[2].

E-mail: sampadakhorgade111@gmail.com[1];sambharepraful832@gmail.com[2]

ABSTRACT

About millions of visitors are interacting daily with web sites around the world, for this several kinds of data have to be organized in such a manner that they can be accessed by several users effectively and efficiently. Web mining is the process of extraction of exciting and useful facts of information. The objective behind Web Usage Mining (WUM) is to analyze web log files for extracting usage patterns. Several algorithms are proposed for finding sequential patterns. The very first is the Apriori algorithm, which was put forward by the founders themselves. Later more scalable algorithms for complex applications were developed as GSP, Prefix Span algorithm etc. In this paper, a systematic review of the algorithms is proposed by classifying pattern-mining process for finding the patterns the algorithms are designed to increase efficiency of patterns by mining it. To evaluate the effectiveness and efficiency of sequential pattern mining algorithms, an extensive performance study is done on these algorithms: Prefix Span, Free Span, GSP and Apriori. The Prefix span algorithm is best suited both in terms of scalability, time complexity, memory usage etc. Various process of web usage mining such as preprocessing, pattern discovery, pattern analysis been discussed.

Keywords: Web Usage Mining, Preprocessing, Pattern Discovery, Pattern Analysis, Weblog.

1. INTRODUCTION

With the growth of information technology, the Internet has penetrated almost every field of our world. Also there is rapid increase in number of web sites, web pages for discovering and understanding the web users and its surfing behavior becomes essential for the development of successful web monitoring and recommendation systems. Analyzing the web logs from server for extracting the users navigating pattern has become necessary for website administrator to make sure that the site serves the users needs. The surfing behavior of web user gets recorded and stored in a text file which is basically known as web log file, but it consists of huge amount of information also some kind of certain undesirable rather useless data which has nothing to do with the mining procedure. Preprocessing of log file an important role for discovering the patterns, and its aim is to transform the raw stream of dataset for building user profiles. Data preprocessing presents a number unique algorithms and techniques for preprocessing tasks such as cleaning, user and session identification etc [4]. Data mining techniques are applied on the web log files to remove out unnecessary data and then finding the patterns from that file. The process of applying the data mining techniques on web data to discover out the particular patterns is known as web mining [1].

1.1 Preprocessing

Web log contains lots of incomplete, noisy and inconsistent record it is necessary to perform preprocessing on log file to improve efficiency and scalability. Web log data preprocessing can be done in several ways as data cleaning, page view identification, user identification, session identification etc [5].

1.2 Pattern Discovery

Pattern discovery means finding the patterns from the processed log data. For this reason the data must have passed through pre-processing phase. The discovered pattern helps the website administrator to find out the ongoing search the users follow up [4].

1.3 Pattern Analysis

It is the process after pattern discovery and it checks that patterns on the web is correct or not and guides the process of extraction of the information from the web. After completing the above steps obtained usage patterns are analyzed to filter uninteresting information and extract the useful information usage mining process. In this phase, uninteresting pattern are removed from the patterns identified during pattern discovery phase [4].

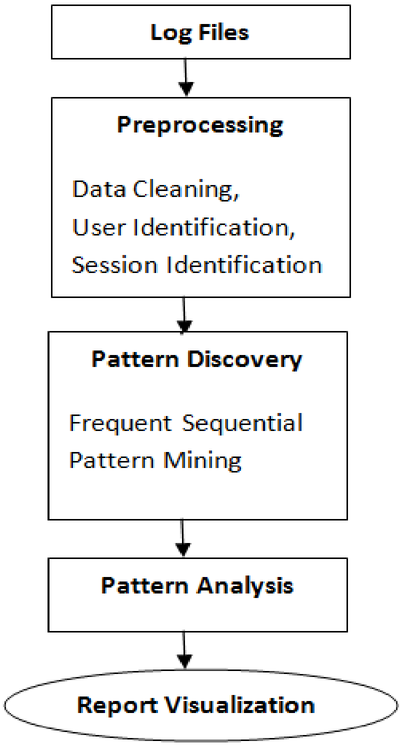

Fig 1: Process of Web Usage Mining

2. BACKGROUND

For discovering the frequent sequential web access patterns various algorithms were proposed by various Scientist and experiments were formed, following described algorithms were used for finding the relevant patterns from the web access log.

The GSP(Generalized Sequential Patterns) algorithm described by Agrawal and Shrikant [2] which makes multiple passes over the data if the candidate patterns do not fit in the memory, the algorithm makes only as many candidates as will fit in memory and the data is scanned to count the support of these candidates. Frequent sequences resulting from these candidates are written to disk, while those candidates without minimum support get deleted and this procedure is repeated until all the candidates have been counted. This means the GSP algorithm finds all the length-1 candidates and orders them with respect to their support and ignoring ones for which support < minsup, finally for each level the algorithm scans database to collect support count for each candidate sequence and generates candidate length (k+1) sequences from length-k frequent sequences using Apriori. This procedure is repeated until no frequent sequence or no candidate is found also the candidate generation-and-test methods GSP algorithm, requires a substantial amount of memory and not efficient.

The Apriori [Agrawal and Shrikant 1994] and AprioriAll set the algorithm that depends largely on the Apriori property and use the Apriori-generate join procedure to generate candidate sequences. The Apriori method states that all nonempty subsets of a frequent itemset must also be frequent [3]. Key features of Apriori-based algorithm are described as level-wise search algorithms as they construct all the k-sequences, in kth iteration of the algorithm, and they traverse the search space. Also the Generate-and-test is used by the very early algorithms in sequential pattern mining. Algorithms that depend on this feature only display an inefficient pruning method and generate a large number of candidate sequences and then test each one by one for satisfying some user specified constraints but consume lot of memory in the early stages of mining. It scans the original database whether a long list of generated candidate sequences is frequent or not. It is a very undesirable characteristic of most Apriori based algorithms and requires a lot of processing time and I/O cost [7].

Soon after the Apriori-based methods of the mid-1990s, the pattern growth method emerged in the early 2000s, which is a solution to the problem of generate-and-test. The key idea behind is to avoid the candidate generation step altogether, and to focus the search on a restricted portion of the initial database. The search space partitioning feature plays an important role in pattern growth. Almost every pattern growth algorithm starts with building a representation of the database to be mined, it then proposes a way to partition the search space, which then generates as few candidate sequences as possible by growing on the already mined frequent sequences, and that applying the Apriori property as the search space is being traversed recursively looking for frequent sequences [3]. The main features of pattern growth based algorithm are search space partitioning which allows it to partition the generated search space of large candidate sequences for efficient memory management. And the Tree projection algorithm it then implements a physical tree data structure representation of the search space, which is then traversed breadth-first or depth-first in search of frequent sequences and pruning is based on the Apriori property for finding patterns. Next is Depth first traversal improves the performance, and also helps in the early pruning method of candidate sequences. The main reason for this performance is that it utilizes far less memory and more directed search space, and thus less candidate sequence gets generated than breadth-first. A pattern-growth algorithm utilizes a data structure that allows them to prune candidate sequences early in the mining process. This result in early display of smaller search space and maintain a more directed and narrower search procedure.

The Prefix-projected Sequential pattern mining (Prefix Span) algorithm presented by Jian Pei, Jiawei Han represent the pattern-growth which finds the frequent items after scanning the sequence database once. The database is then prepared according to the frequent items, into several smaller databases. Finally, the complete set of sequential patterns is found by recursively growing subsequence fragments in each projected database. Prefix Span algorithm uses depth first search based approach, top down search which are efficient techniques to find frequent subsequences as sequential patterns form the large database, the Prefix Span does not generate any useless candidate and it only counts the frequency of local 1-itemsets, Since it generates no candidates and explores the divide-and-conquer methodology, it consumes stable memory space throughout the mining process [6].

Table 1. Comparative study of Techniques

Technique	Author	Advantage	Disadvantage
GSP Algorithm	Shrikant and Agrawal	Execution time, minimum threshold for scanning	inefficient in mining long sequential patterns
Apriori algorithm	Agrawal and Shrikant	Scans only original database	inefficient pruning generates number of candidate sequence, more processing time and I/O cost
Pattern Growth algorithm	J.Han.et.al	Efficient & improves performance	Consumes much memory
Prefix span algorithm	Jian Pei, Jiawei Han and Helen Pint	Execution time, memory usage, Scalability	Extra scanning time in database, big storage for projected database

3. CONCLUSION

Web log files are the best source to predict users behavior. Along with the useful information the raw log files also contains entries for unnecessary information image access, failed entries etc. which are of no use from the perspective of the Web Usage Mining. Therefore, it becomes necessary to get rid of this irrelevant information this is done by process of web mining. And from the comparative analysis of various mining algorithms, it is clear that Prefix-span based on pattern growth algorithms are more efficient with respect to running time, space utilization and scalability for finding the appropriate patterns. Using the Prefix Span algorithm the relevant patterns get identified and becomes easy for finding the users navigation behavior. So doing the above discuss process the information is obtained along with the relevant patterns.

4. REFERENCES

[1]. Navin Kumar Tyagi, A.K. Solanki and Sanjay Tyagi," An Algorithmic approach to data preprocessing in web usage mining" International Journal of Information Technology and Knowledge Management Volume 2, No. 2, pp. 279-283, July-December 2010

[2]. Shrikant R. and Agrawal R.,"Mining sequential patterns: Generalizations and performance improvements", Proceedings of the 5th International Conference Extending Database Technology, 1996, 1057, 3-17.

[3]. Chetna Chand, Amit Thakkar, Amit Ganatra "Sequential Pattern Mining: Survey and Current Research Challenges" International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012

[4]. Surbhi Anand, Rinkle Rani Aggarwal," An Efficient Algorithm for Data Cleaning of Log File using File Extensions", International Journal of Computer Applications (0975 – 888) Volume 48– No.8, June 2012

[5]. Manisha Valera, Uttam Chauhan GTU, "An Efficient Web Recommender System based on Approach of Mining Frequent Sequential Pattern from Customized Web Log Preprocessing" *IEEE – 31661*

[6]. J. Pei, J. Han, B. Mortazavi-Asi, H. Pino, "Prefix Span: Mining Sequential Patterns Efficiently by Prefix- Projected Pattern Growth", ICDE'01, 2001.

[7]. NIZAR R. MABROUKEH and C. I. EZEIFE, "A Taxonomy of Sequential Pattern Mining Algorithms" ACM Computing Surveys, Vol. 43, No. 1, Article 3, Publication date: November 2010.