Implementation Of Sequential Pattern Algorithms In Web Usage Mining

V Aruna¹, Dr. Harsh Pratap Singh², Dr. D. Sujatha³

¹Research Scholar, Dept. of Computer Science & Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal-Indore Road, MadhyaPradesh, India
²Research Guide, Dept. of Computer Science & Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal-Indore Road, MadhyaPradesh, India

ABSTRACT

Sequential pattern mining is a data-mining technique used for determining time-related behaviour in sequential databases. The information extracted from sequential pattern mining can be used in marketing fields, medical records, sales analysis, and so on. Several algorithms for maintaining sequential patterns have been developed. Although there have been many recent updates on the sequential patterns in static database. The complexity of sequential pattern mining increases in the dynamic database because of the insertion of new data sets. This paper characterizing sequential pattern-mining algorithms based on significant key highlights upheld by the strategies. This paper targets comprehension of sequential pattern mining issues.

KEYWORDS: Sequential pattern mining, frequent pattern mining, Item set mining, Data Mining, Sequential Patterns.

I. INTRODUCTION

Data mining is a process of extracting valuable patterns from huge amount of data. Data is present in databases, data distribution centres, or some other data repositories. This huge collection of data represents no useful information. Various data mining techniques are applied on this data to handle it and discover valuable knowledge and patterns from it. The approaches used to mine the data are Statistical examination, Association rules, Classification and Clustering. Data mining is effectively used in different fields like science, wellbeing, marketing, money and so forth. Web mining is a part of Data mining in which different mining procedures are applied on the data present on the web to discover previously unknown fascinating and valuable patterns. Applications of web mining are in the areas of market division, business insight, estimating returns of online missions, E-trade. The classification of web mining is as demonstrated in the figure 1.1. Web mining is basically categorized into three types. Web structure mining deals with extracting interesting patterns and data based on the structural design of the web. The structure of web is generally uses links which focuses to a page or an asset from the referrer in which it lives. By having such links on each page on the web, the web can be seen as a coordinated diagram having hubs as pages and assets and links as the edges among them. Web structure mining basically deals with these hyperlinks on the web pages. Different techniques used for web structure mining are PageRank, HITS, weighted PageRank and Topic Sensitive PageRank.
Web content mining: Web content mining is the process of extracting information from Web documents. The contents of a web document can be text, image, video, sound or records like lists and tables which are used to convey information to the users. Most of the data which is available on web is unstructured. Database approach and agent based approach are used in web content mining. This unstructured data from web documents is retrieved by making use of database approach. The agent based approach is used for searching the relevant information and organizing the collected information. The information retrieval view and database view are two different views of web content mining. The content mining from information retrieval view helps the users in filtering and finding the information from the web, whereas as the database view manages the web data.

Web usage mining basically deals with finding out what users are looking for on the internet. Web Usage Mining (WUM) is the process of extracting useful information from web log based on users' needs. Web data preprocessing should be done on the huge data present in the web in order to get the user needed information. The different phases in web usage mining include data cleaning process, data preparation process, user identification, session identification, data integration, data transformation, pattern discovery, and pattern analysis. The data preprocessing is most critical phase in the WUM. The data pre-processing step is applied by taking original data or on the data that is integrated from multiple sources. The purpose of web usage mining is to retrieve the raw data from web and analyze the pattern after it is discovered. Log files provide information about the activity of user, viz., which web site he/she using, whom you send/receive e-mail etc.

Sequential Pattern Mining Algorithms

Sequential pattern mining is a data-mining technique for determining time-related behavior in sequential databases. The information extracted from sequential pattern mining can be used in marketing fields, medical records, sales analysis, and so on. Several algorithms for maintaining sequential patterns have been developed. Although there have been many recent updates on the sequential patterns in static database, the complexity of sequential pattern mining increases in the dynamic database because of the insertion of new data sets.
Sequential pattern mining algorithms are used to discover the existing frequent sequences in a given database. Sequential pattern mining is related to association rule mining, except that the events are linked by time. Sequential patterns represent the inter transaction relationships whereas associations are used to represent intra transaction relationships. In association rule mining, the results contain the items which are bought together frequently in the same transaction. While the results of sequential pattern mining contain the items that are bought in a certain order by the same customer in different transactions.

In this paper the various parts of grouping the succession pattern mining algorithms are depicted. The primary highlights of the most popular algorithms are illustrated. An essential level-wise algorithm, the GSP is presented in detail. It is a significant strategy, since it can deal with the straightforward succession mining issue, yet additionally the issue of characterizing time imperatives, sliding time windows and scientific categorizations in sequential patterns too. We have examined a set of mining ways to deal with comprehend the adequacy of pattern disclosure in data mining field. Some of them are depicted sequentially in this segment.

**GSP Algorithm**

GSP (Generalized Sequential Patterns) is a sequential pattern mining algorithm which was proposed by Srikant and Agrawal in 1996. GSP is an Apriori based algorithm. It creates heaps of up-and-comer sets and it tests them by numerous passes. The algorithm to locate the sequential patterns is plot as follows: First, it scans the database to locate the regular items, that is, those with equivalent or more noteworthy than least help. Those continuous items are length-1 incessant groupings. Second, every one of them begins with a seed set of sequential patterns to create new possibly sequential patterns, called up-and-comer groupings. Every applicant grouping contains more than one item from which pattern it is created. The length of each arrangement is the quantity of occasions of items in a succession. The entirety of the applicant successions has a similar length in a given pass. To locate the successive arrangement, the algorithm at that point scans the database and disposes of those competitors which are rare. At long last, subsequent to getting the successive groupings it makes those arrangements as the seed for the following pass. The algorithm ends, when there are no incessant successions toward the finish of a pass, or when there are no competitor groupings created. At the point when new successions come, GSP begins its mining cycle from the earliest starting point. That implies, GSP scans the entire refreshed database (old + new) during incremental mining so various scanning of database and competitor successions age increment as the size of the database increments.

**Prefix Span Algorithm**

Prefix Span is a projection-based, sequential pattern-development approach for proficient and versatile mining of sequential patterns, which is an augmentation of FP-development. Unlike apriori-based algorithms it doesn't make huge number of futile applicant sets and creates total set of sequential patterns from enormous databases productively. The significant expense of Prefix Span is database projection, i.e., framing projected databases recursively. To locate the sequential patterns, Prefix Span recursively projects an arrangement database into a set of little projected databases and sequential patterns are filled in each extended database by investigating just locally incessant pieces. In this methodology, sequential patterns from arrangement database can be mined by a prefix projection strategy in the accompanying advances: Find length-1 sequential patterns. Scan database once to locate all the incessant items in groupings. Every one of these incessant items is a length-1 sequential pattern. Separation search space. The total set of sequential patterns can be apportioned by the quantity of length-1
sequential patterns (prefixes) found in sync 1. Discover subsets of sequential patterns. The subsets of sequential patterns can be mined by developing the relating set of projected databases and mining each recursively. During incremental mining, Prefix Span mines the refreshed database from the scratch by scanning the entire refreshed database (old + new). So that, projected database age and scanning of each projected database increment simultaneously as the size of the database raises.

FUSP – Tree Algorithm
To productively mine the sequential patterns, proposed the FUSP-tree design and its support algorithm. FUSP-tree comprises of one root hub marked as „root” and a set of prefix sub trees as the offspring of the root. Every hub in the prefix sub trees contains item-name; which represents the hub contains that regular item, check; the quantity of successions represented by the part of the way arriving at the hub, and hub link; links to the following hub of that item in the following part of the FUSP-tree. The FUSP-tree contains a Header-Table which store successive item, their check and the link of first event hub in the tree of that item. This table assists with finding suitable items or arrangements in the tree. The development cycle is like FP-tree the development cycle is executed tuple by tuple from first succession to last. To make this tree, it requires two scans of huge database which expands the tree development time. Mining cycle of FUSP-Tree is practically like Prefix Span and FP-development algorithms. After the FUSP-Tree is kept up, the last continuous arrangements would then be able to be found by a recursive technique from the tree. This strategy finds the sequential patterns from the FUSP-Tree structure by creating set of little extended trees from the enormous tree recursively. It creates no up-and-comer sets except for it produces many extended trees for each prefix grouping which require more memory. During incremental mining, when any rare item in the old database become successive in the refreshed database, in the present circumstance, this algorithm rescans the old database to discover the help of the rare item from the old database, i.e., it scans the entire refreshed database (old + new) to refresh the old FUSP-Tree structure because of capacity of regular items as it were. So that scans of database increments simultaneously as the size of the database raises.

EXPERIMENTAL METHOD
Sequential patterns generated can be used by the managers to determine which items are bought one after another in a sequence, or to analyze browsing orders of homepages in a Web site. Sequential pattern mining is usually defined as finding the entire set of frequent subsequences in a given set of sequences. Sequential pattern is a sequence of item sets that occur frequently in a specific order; all items in the same item set are supposed to have the same transaction time value or within a time gap. Each sequence corresponds to a temporally ordered list of events, where each event may be a collection of things (item set) occurring simultaneously. The temporal ordering among the events is induced by absolutely the timestamps related to the events. Usually, all the transactions of a customer are together viewed as a sequence, called customer-sequence, where each transaction is represented as an item set in a sequence and each one the transactions are listed in a certain order with regard to the transaction-time. The approach for mining sequential patterns from a client exchange database is portrayed as follows:
Let D be a set of customer transactions where each transaction T consists of a customer-id, a transaction time and a set of items involved in the transaction. Let \( \{ i_1, i_2, \ldots, i_n \} \) be a set of literals called items. An item set \( s \) is a set of item sets ordered according to their time stamp. It is denoted by \( n, s, i_1, i_2, \ldots, i_n \) where \( s \) is a set of items and \( n \) is an item set. A k-sequence is a sequence of k items (or of length k). A sequence \( s \) is a sub-sequence of another sequence \( s' \) if \( i_1, i_2, \ldots, i_n \) such that \( \forall s', i_1, i_2, \ldots, i_n \in s' \subseteq s \). The problem of mining sequential patterns is to find all sequences \( s \) such that \( \text{support}(s) \geq \minsup \) for a database \( D \), given a support threshold \( \minsup \). The task of finding all incessant arrangements in enormous databases is very testing, since the inquiry space is amazingly huge. For example, with \( m \) ascribes there are \( O(mk) \) possibly successive groupings of length \( k \). The elements, which make sequential pattern mining an extremely troublesome and tedious one, are as per the following: first, the development of a pattern isn't restricted to single items in given item sets.

Proposed Sequential Pattern Algorithm
In Sequential Pattern strategy for mining, there are two modules in which the first is to assign the weightage to the each regular item and then finding sequential pattern. In the proposed Sequential Pattern algorithm, in first part, the events of client gets to webpage is utilized as to relegate the weightage to every item from an entrance grouping. In the second piece of the proposed algorithm, pattern finding algorithm thinks about the heaviness of arrangement and utilized it for discovering the help decided for client meeting and relying on this the sequential patterns are found.

www.turkjphysiotherrehabil.org
Table 1: Data Structures used in the proposed algorithm

<table>
<thead>
<tr>
<th>Modified Data Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item_List (ItLt)</td>
<td>Linked list modified for containing the items and their occurrence in the web access log data.</td>
</tr>
<tr>
<td>Curr_Item_List (CrItLt)</td>
<td>Linked list modified so that it can be used for storing the items from the current user access webpages.</td>
</tr>
<tr>
<td>First_Occ_List (FsOcLt)</td>
<td>Linked list modified for storing the first occurrences of a given item in the web access log data.</td>
</tr>
<tr>
<td>Ace_List (AcLt)</td>
<td>Linked list modified for storing each web access sequence.</td>
</tr>
<tr>
<td>Header_List[]</td>
<td>Array structure containing start address of list containing item with their weight represented as</td>
</tr>
</tbody>
</table>
initial seed point. Accurate clustering requires a clear definition of the closeness between a pair of objects, in terms of either the pair wise similarity or distance. A variety of similarity or distance measures have been proposed and are widely used, such as cosine similarity, Jaccard coefficient, Euclidean distance, Pearson Correlation Coefficient and City Block etc., So an algorithm K-means Track execution is designed to enhance k-means in which there are two steps. In the first step dataset is divided into subsets and then the initial cluster points are calculated. The proposed K-means Track Execution (K-means TE) is designed for optimizing the following parameters.

- Clustering accuracy
- Error rate
- Time complexity
- Space complexity

CONCLUSION

This paper has given a detailed study of sequential pattern mining. The paper has presented the principle sorts of algorithms for finding sequential patterns. In addition, the paper has presented significant enhancements of the sequential pattern mining issues that address a few inadequacies of sequential pattern mining. Also, the paper has addressed some issues identified with sequential pattern mining, for example, item set mining, association rule mining, sequential guideline mining and intermittent pattern mining. At last, a framework was devised for web usage data clustering for users’ sessions using variants of k-Means clustering algorithms.

REFERENCES

5. Gupta M. and Han J., “Approaches for Pattern Discovery Using Sequential Data mining”, Pattern Discovery using Sequential Data Mining, 2011