© 2016 IJSRSET | Volume 2 | Issue 3 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Pure Incremental Approach for Sequential Pattern Mining

Bhargav Shroff, Prof. Bakul Panchal

Information Technology, L. D. Engineering College, Ahmedabad, Gujarat, India

ABSTRACT

In data mining, mining sequential pattern from a very huge amount of database is very useful in many applications. Most of sequential pattern mining algorithms works on static data means the database should not change. But the databases in today's real world application do not have static data, rather they are incremental databases. New transactions are added at some intervals of time in database. For updated database, the algorithm actually needs to be executed again for whole sequence database. So those approaches are not appropriate to use, for that the algorithm with incremental approach should be modelled and used. In this paper analysis of existing approaches for finding sequential pattern mining, and the survey is helpful in forming a new model or improving some existing approach to handle incremented database & obtain sequential patterns out of them. In this a proposed a model that is totally incremental approach, which we call pure incremental approach. This proposed pure incremental mining is used for mining the frequent sequences for sequence database.

Keywords: BLSPM, Incremental approach, IncSpan, PrefixSpan, Sequential Pattern mining.

I. INTRODUCTION

Data Mining has been considered as a very important area of research since many decades. Mining useful and unknown knowledge from vast amount of databases has remained its goal. There are various techniques in data mining like association rules mining, classification, clustering, sequential pattern mining, etc. Sequential pattern mining was first of all introduced by R. Agrawal and R. Srikant. Sequential pattern mining is basically obtaining frequently occurring ordered events or some subsequence from database.

A sequential pattern is a relatively common sub-sequence of transactions, where each transaction is a set of items (Itemset).[5] For example, each customer record in the transactional database is an Itemset associated with the transaction time and a customer-id.[5] Data having the same customer-id are sorted by ascending transaction time into a data sequence before mining. If a sufficient number of customers in the transactional database have the purchasing sequence of PC, printer, and printing software, then such a sequence is called a sequential pattern.[5]
Mostly sequence mining algorithms works on static databases, i.e. the data should not change in the database. If the database is updated then the database needs to be rescanned and again that particular algorithm should be applied on them. But the database is not static in practice; new transactions would be added to the databases. So the sequential patterns should be mined in incremental sequential database in such a way that whole database need not be rescanned in the process.

In real world applications, the database changes to little extent. For example, a retail sales database is updated each month, sales data for the new month often represent only a small percentage of the previous ten year's sales data.[5] Sequential patterns also change very little, so applying the algorithm on whole updated database is waste of time and cost. So for that some incremental approach needs to be modelled that would consider only the incremented fragment of database. For that various existing approaches are surveyed in this paper.

II. METHODS AND MATERIAL

A. Related Work

While literature survey, in [1] a new method for mining sequential patterns called SPAM (Sequential Pattern Mining) that integrates variety of old and new algorithmic contributions and add into a practical algorithm. The algorithm is very efficient when the sequential patterns are very long in the database. In this paper they introduce novel depth-first search strategy that combine a depth-first traversal of the search space with effective pruning mechanisms.

In [3], PrefixSpan algorithm uses the method of Divide and Conquer for generating frequent suffix items complete frequent prefix projection. Then generate new patterns by connecting them. However, Some shortcomings are also of this algorithm. Firstly, it needs a large resource to build projected database recursively. Then second, the algorithm requires again and again scan projected database, which will cut the efficiency of the algorithm. Thus, author presents an improved sequential pattern mining algorithm BLSPM.

This algorithm is reducing the size and number of projected databases. The experimental data prove that it is very efficient to mining sequential patterns in large data. For future work in this paper, it will use BLSPM algorithm for solving practical problems and ultimately to find the most realistic needs of sequential patterns.

In [4], author develops an efficient algorithm that is IncSpan. It is develop for incremental mining of sequential patterns, by exploring some interesting properties. In this method, they buffer semi-frequent patterns that can be considered as a statistics-based approach.

INCSpan is the fastest then non-incremental PrefixSpan and incremental ISM. Balance efficiency and reusability. This algorithm uses projected database that’s why the approach is non-incremental PrefixSpan. Main drawback is that not able to mine complete set of frequent sequential pattern completely. Though it’s based on PrefixSpan execution time is also somehow high.

B. Problem with Existing Approaches

Sequential mining algorithms can mine a static database. But, nowadays, almost all databases are dynamic in nature and they grow incrementally. To mine whole database every time it is updated and it is highly inefficient. For this Incremental Approach is utilized.
It utilizes the mined information to get new set of frequent sequential patterns instead of mining the whole database from scratch.

The ultimate aim of using an incremental mining algorithm instead of non-incremental one is to gain efficiency with respect to time.

III. RESULTS AND DISCUSSION

A. Proposed Approach

Sequential pattern mining is an important data mining task, different algorithms have been proposed to perform this task efficiently. The problem with this, is to find all sequential patterns with higher or equal support to some predefined minimum support threshold in a data sequence database.

Objectives of the proposed solution

The problems or the limitations defined in the above section of this chapter are proposed to be solved by:

- To observe the effect of various existing algorithms for mining the frequent sequences on various datasets.
- To propose a pure incremental approach for mining the frequent sequences for sequence database i.e. for the above problem.
- To validate the incremental approach on different datasets used in various ways.
- Mining of only newly generated sequence items.
- Represent the output by the pattern.
- Compare the result with the existing algorithms result

Work-flow of Proposed Approach

Step 1: Based on the two threshold value, original database sequences are divided into small, pre-large and large sequences. The original large and pre-large sequences with their counts from preceding runs are retained for later use in maintenance.

Step 2: As new transactions are added, the proposed approach scans these transactions to obtain candidate 1 Itemset, and then these candidate sequences are divided into three parts according to whether they are large, pre-large or small in the original database.

Step 3: They are compared to the large and pre-large 1-sequences which were previously retained.

CASE 1: If a candidate 1-sequence is also among the previously retained large or pre large 1-sequences, its new total count for the entire updated database can easily be calculated from its current count and previous count, since all previous large and pre-large sequences with their counts have been retained.

If an original large or pre-large sequence is still large or pre-large after new transactions are added is then determined from its new support ratio, which is then derived from its total count over the total number of customer sequences.

CASE 2: If a candidate 1-sequence does not exist among previously retained large or pre-large 1-sequences, then the sequence is absolutely not large for entire updated database when the number of newly merged customer sequences is within the safety bound.

In this situation, no action is needed. When new transaction data are incrementally added and the total number of newly added customer sequences exceeds the safety bound, the original database must be re-scanned to find new large and pre-large sequences.

The proposed approach can thus find all large 1-sequences for the entire updated database.

Step 4: After that, candidate 2-sequences from the newly merged customer sequences are formed, and the same procedure is used to find all large 2-sequences. This procedure is repeated until all large sequences have been found.

B. Implementation And Results

The methodology proposed in this paper is implemented and analysed. The platform details and mining tool details are:

- **Hardware Specification**
  - Processor: - Intel(R) Core(TM) i3 CPU M370 @2.40GHz
  - RAM: - 4 GB RAM
- **Software Specification**
  - OS: - Microsoft Windows 7
  - System type: - 64 bit Operating System

For the Implementation of Sequential Pattern Mining Algorithm, SPMF (Sequential pattern Mining Framework) tool is used. The results of implemented algorithm is obtained and compared with existing algorithm.

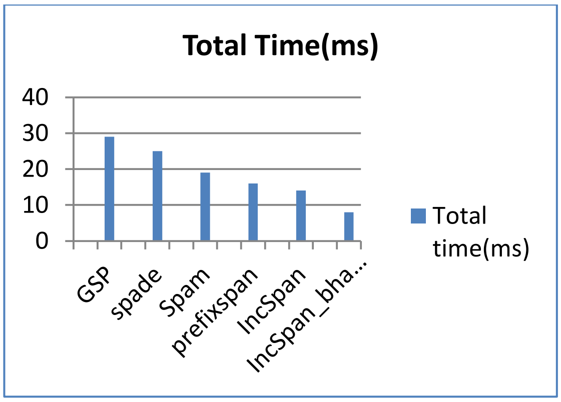

Figure 1: Comparison of Time-taken

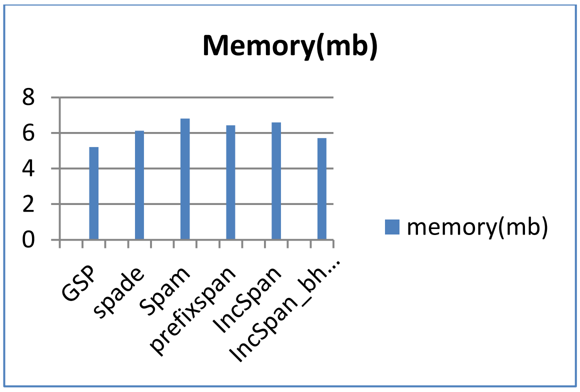

Figure 2 : Comparison of Max-Memory Used

IV. CONCLUSION AND FUTURE WORK

In this work we have reviewed many different methods and techniques that are used for finding sequential pattern mining. Many past techniques are available, but they are suitable only for static databases. Some of techniques with incremental approaches are also considered. All these techniques and algorithms have their own advantages and disadvantages. A methodology is proposed in this work, in which works on pure incremental approach. It uses two threshold values to define the small, large and pre-large item sets from the existing database. So when comparison of sequential patterns needs to be done, than it reduces the rescanning of the original database and to maintain the original cost.

In Future Work, a new concept of Data streaming sequential patterns has been proposed recently in which not only the order of events but the time between them is also considered. This work can be further extended to develop an incremental mining algorithm for streaming of data sequential patterns.

V. REFERENCES

[1] "Sequential PAttern Mining using A Bitmap Representation", Jay Ayres, Johannes Gehrke, Tomi Yiu, and Jason Flannick, in ACM.

[2] "Prediction of Students Performance Using Frequent Pattern Tree", Priyanka Anandrao Patil, R. V. Mane, in 2014 Sixth International Conference on Computational Intelligence and Communication Networks, IEEE.

[3] "A Improved PrefixSpan Algorithm For Sequential Pattern Mining", Liang Dong, Wang hong, in 2014 IEEE

[4] "IncSpan: Incremental Mining of Sequential Patterns in Large Database", Hong Cheng, Xifeng Yan, in ACM

[5] "Incremental Discovery of Sequential Patterns Using a Backward Mining Approach", Ming-Yen Lin,Sue-Chen Hsueh,Chih-Chen Chan, in IEEE

[6] Endu Duneja, A.K. Sachan," A Proficient Approach of Incremental Algorithm for Frequent Pattern Mining" IJSR.

[7] Bagrudeen Bazeer Ahamed and Shanmugasundaram Hariharan, "A Survey On Distributed Data Mining Process Via Grid".

[8] Bhargav Shroff, Prof. Bakul B. Panchal, ―A Survey On Different Approaches For Sequential Pattern Mining‖, International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET)

[9] "data mining" http://en.wikibooks.org/wiki /Data_Mining_Algorithms_In_R/Sequence_Mining/SDE

[10] By Jiawei Han And Micheline Kamber,Data Mining Concept and Techniques, Copyright 2006, Second Edition.