A STUDY ON CONSTRAINED FREQUENT PATTERN MINING OF STREAMING DATA

Dr.B.Lavanya
lavanmu@gmail.com
Assistant Professor
Department of Computer Science
University Of Madras, Chepauk, Chennai-600005

Abstract: In most of the real time applications data may arrive as continuous ordered sequence of items, called data streams. Constraints are used to specify rules for the data. This study deals with different data stream mining and sequence pattern mining techniques and found that most of the techniques recover a most number of results. Data stream is voluminous, complex and dynamically arriving stream of data. There are certain techniques to deal with data streams, in particular, finding the frequent or sequential patterns that occur repeatedly. These results retrieve huge number of patterns, which are hard to analyse and use them, also difficult to store these results and its intermediate results. The traditional pattern mining techniques fail to give the relevant details to the use, which is hard and get the material information for the user in order to filter results/patterns receive from those techniques, constraint based mining approach, in which user can give the constraints and get the required and right information from the data stream.

Keywords: Data stream mining, Frequent mining, Constraint, Sequential data mining, pattern mining.

1. INTRODUCTION

Frequent pattern mining focuses on searching frequently occurring patterns from different types of data sets, including structured, unstructured, text datasets, semi-structured, XML datasets and graph datasets[3]. The patterns can be item-sets, sequences, sub-trees, or sub-graph, etc., depending on the mining tasks and targeting data sets [11]. The unique challenges in discovering frequent pattern mining need to search for space with more number of patterns. The numbers of the answering set itself which contains all frequent patterns can be very large too. In particular, it can cost much more space to generate an approximate answering set for frequent patterns in a streaming environment. Therefore, the mining algorithm needs to be very memory-efficient. Second, frequent pattern mining relies on the down-closure property to prune infrequent patterns and generate the frequent ones [4]. This process (even without the streaming constraint) is very compute-intensive. Consequently, safety up the high-speed data streams can be very hard for a frequent pattern-mining task[4]. Given these challenges, a more important issue is the quality of the approximate mining results. The more right results usually require more memory and computations.

1.1 Importance of constraint based Mining

Frequent pattern mining usually produces too many solution patterns. This situation is harmful for two reasons:

a. Performance: mining is usually inefficient or often simply unfeasible

b. Identification of fragments of interesting knowledge: which is blurred within a huge quantity of small, mostly useless patterns.

1.2 Constraints are the solution to these problems

They can be pushed in the frequent pattern computation exploiting them in pruning the search space, thus reducing time and resources requirements. They provide to the user guidance over the mining process and a way of focusing on interesting knowledge. With constraints we can obtain less patterns which are more interesting[14][15].

General Framework of Data Stream

Association rule mining finds frequent itemsets[2] which are satisfying minimum support threshold value, based on that strong association rules is generated. The association rule generate set of rule which satisfy user defined theshold value and Based on that one can develop marketing strategies. Not only in sales marketing, there are many areas such as inventory management, sales management and strategy management etc. in which this kind of strong rule become very helpful.

Table 1. DBMS (Database Management System Vs DSMS (Data Stream Management System)[4][5]
There are many key challenges in data streaming mining that need to be overcome like storage, high speed processing, immediate response etc. As shown in figure data stream generated from many data sources, enters at high speed in Data Stream Management System (DSMS) [11]. In DSMS, algorithm may use different types of model based on user interest.

### 2 APPROACH OF CONSTRAINT BASED DATA AND SEQUENCE PATTERN MINING

#### 2.1 Constraint based mining

Frequent Item set mining or Frequent Pattern Mining algorithms retrieve patterns that may appear more than a threshold value. [6]

**2.1 (a) Counter based algorithm**

In this method the clustering is performed by incorporation of user or application oriented constraints. Data stream can be classified as offline stream in which regular bulk arrivals and online stream in which real time data is updated one by one as time progresses. Frequent item set mining or frequent pattern mining algorithms retrieve patterns that may appear more than a threshold value. The constraint refer to the user expectation or the properties of desired clustering results. The constraint given us the clustering process. The constraint can be specified by the user or the application requirement.

- **(a) Majority Algorithm**: It selects each item and retrieves the majority vote, if any, and then the algorithm stores that item. The algorithm verifies by taking the number of occurrences of items stored
- **(b) Frequent Algorithm**: It stores k-1 (item, counter) pairs ensure that the count associated with each item on termination is at most below the true value. It finds the items which occur more than the given fraction of time,
- **(c) Lossy Algorithm**: It stores an item and its counts. A “delta” value is calculated such that the difference between the upper bound and the lower bound. If the delta value of an item is increased, all its count will be decreased by 1, and items with zero count will be deleted.
- **(d) Space Saving Algorithm**: It is a combination of frequent algorithm and lossy algorithm. It stores k items as a pair of (item-name, count). Algorithm initializes first distinct k items and their counts. If the next item matches with the k items stored then the count is incremented else the smallest count has its item value replaced with the new item, and the count is incremented.
- **(b) Quantile Algorithm**

**2.1 (b) Quantile Algorithm**

- **(a) Gk Algorithm**: It stores tuples containing an item from the input, a frequency count g and a delta value.
- **(b) Q Digest**: A dyadic range and a count. A dyadic range is a range whose length is a power of two, and which ends at a multiple of its own length.
- **(c) Count Sketch**: This algorithm uses a hash function and hash table used for estimating frequent number of occurrences and storing the estimated values. The count is estimated for each element and it maintains a heap of top count k elements. The efficiency of the algorithm is improved by using the same technique is used for every update, which affects only a small subset instead of the whole data stored.

#### 2.2 Sequential Pattern Mining

Sequential pattern mining refers to the mining of frequently occurring ordered events or subsequence as pattern.

- **(a) GSP (Generalize Sequential Pattern)**: GSP mining which based on Apriori algorithm to mine frequent item sets. Initially it finds all the frequent of length one item with minimum support.
- **(b) Temporal Association Rule**: This method is using Apriori approach to help improve delivery specific pathologies. This result constraint of antecedents and consequence signifying that if antecedent occurs then the consequent would also occur with a certain probability.
- **(c) Fast Time Interval Method**: Exploits transitivity inherent in temporal relation. The mining algorithm based on pattern growth technique. When mining classification rules for document a user may be interested in only frequent patterns with at least 5 keywords.

### 3. Classification of Constraints

#### 3.1 Based on data stream semantics

- **(a) Item Constraint**: An item constraint specifies what are the particular individual or group of items that should not be present in pattern. For example a soap company may be interested in patterns containing only soap products, when it mines transactions in a grocery store.
- **(b) Length Constraint**: A length constraint specifies the requirement on the length of patterns, the number of items in the patterns. For example when mining classification rules for document a user maybe interested in only frequent patterns with at least 5 keywords.
- **(c) Model-based constraint**: A model-based constraint looks for patterns which are sub or super patterns of some given patterns (models). For example: A car dealer may be interested in knowing what are all the other accessory items a purchaser would buy when he buys a car.
- **(d) Aggregate Constraint**: An Aggregate constraint is on an aggregate of items in a pattern, where the aggregate function can be SUM, AVG, MAX, MIN, etc. For example a marketing analyst may like to find pattern where the average price is over $150.
- **(e) User Constraint**: User constraints are those in which user can use a rich set of SQL-style constraints to guide the mining process to find only those frequent patterns—containing market basket items—that satisfy the user constraints.

### 3.2 BASED ON PROPERTIES

<table>
<thead>
<tr>
<th>Data type</th>
<th>Static-data</th>
<th>Stream Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship</td>
<td>Persistent data</td>
<td>Volatile data stream</td>
</tr>
<tr>
<td>Access</td>
<td>Random</td>
<td>Sequential</td>
</tr>
<tr>
<td>Query</td>
<td>One time</td>
<td>Continuous</td>
</tr>
<tr>
<td>Storage</td>
<td>Passive repository</td>
<td>Active repository</td>
</tr>
<tr>
<td>Available memory</td>
<td>Flexible</td>
<td>Limited</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Processing time is not a constraint</th>
<th>Processing time is most important as data may skip</th>
</tr>
</thead>
<tbody>
<tr>
<td>Results</td>
<td>Accurate</td>
<td>Approximate</td>
</tr>
<tr>
<td>Response speed</td>
<td>No time requirements</td>
<td>Real-time requirements</td>
</tr>
<tr>
<td>Data scan</td>
<td>Flexible</td>
<td>One time scan only</td>
</tr>
<tr>
<td>Data Schema</td>
<td>Static</td>
<td>Dynamic</td>
</tr>
</tbody>
</table>

---

(a) **Monotonicity**: Is the most commonly used constraint property in many applications. With the help of monotonicity generating classification trees that satisfies the constraint property, both when training data which satisfies monotone and when it is not.

(b) **Constraint frequent pattern mining** with a patter growth view finds all frequent itemset that satisfy the constraint and then the pattern growth mining method generates and test only a few among them.

(c) **Another classical framework for mining** sequential pattern with prefix monotone constraints. This framework is extendible to aggregate constraints also.

(d) **SQL style constraints can be used for mining** process to find the user specified patterns. UFPS algorithm retrieves frequent patterns that satisfy succinct constraints from uncertain data.

(e) **Convertible constraints** cannot be pushed into algorithms like Apriori. Algorithms like FLCA can be used for pushing convertible anti-monotone constraints in frequent itemset mining and FLCM can used to push convertible monotone constraint in frequent itemset mining.

### 3.3 CONSTRAINTS IN SEQUENTIAL PATTERN MINING PROCESS \[287\]

(a) **Gap Constraint**: A gap constraint is defined in sequence databases where each transaction in every sequence has a timestamp. It requires that the sequential patterns in the sequence database must have the property such that the timestamp difference (difference of days) between every two adjacent transactions in a discovered sequential pattern must not be greater than given gap.

(b) **Compactness Constraint**: A compactness constraint requires that the sequential patterns in the sequence database must have the property such that the time-stamp difference (difference of days) between the first and the last transactions in a discovered sequential pattern must not be greater than given period.

(c) **Recency Constraint**: Recency constraint is specified by giving a recency minimum support \((r\_minsup)\), which is the number of days away from the starting date of the sequence database.

(d) **Monetary Constraint**: Monetary constraint is specified by giving monetary minimum support \((m\_minsup)\). It ensures that the total value of the discovered pattern must be greater than \(m\_minsup\). Suppose the pattern is \(<(a), (bc)>\). Then can say that a sequence satisfies this pattern with respect to the monetary constraint, if \(i\) can find an occurrence of pattern \(<(a), (bc)>\) in this data sequence whose total value must be greater than \(m\_minsup\).

(e) **Frequency Constraint**: The frequency constraint is defined as each discovered pattern must satisfy minimum support. Frequency constraint is specified by giving frequency minimum support \((f\_minsup)\). The frequency of a pattern is the percentage of sequences in database that satisfy the recency constraint.

<table>
<thead>
<tr>
<th>S.NO</th>
<th>ALGORITHM</th>
<th>DESCRIPTION</th>
<th>PERFORMANCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>MAIDS</td>
<td>It is a single scan, multidimensional stream alarming system based on the window concept.</td>
<td>40 MB memory used and 400s execution time.</td>
</tr>
<tr>
<td>2.</td>
<td>CVFDT-2001</td>
<td>It is a decision tree mining algorithm based on the MB concept drift learning technique.</td>
<td>It takes 1k sec and 70 memory to process 1000k transaction.</td>
</tr>
<tr>
<td>3.</td>
<td>OLIN:Last-2002..</td>
<td>Last-2002 It is a fuzzy network based online data stream mining algorithm.</td>
<td>Its time complexity is (O(n)); (n) is no of transaction.</td>
</tr>
<tr>
<td>4.</td>
<td>BTS-2002</td>
<td>It is a landmark based stream mining technique based on minimum support and average error rate.</td>
<td>It takes approx 1k sec and 40MB to process 100k data block.</td>
</tr>
<tr>
<td>5.</td>
<td>INSTANT-2007</td>
<td>It is a stream mining algorithm based on landmark method, it is a simple and efficient.</td>
<td>Its time complexity is (O(xy)); (x) is average size of (y) window</td>
</tr>
<tr>
<td>6.</td>
<td>AWSOM</td>
<td>Arbitrary Window Stream mOdeling Method is a pattern discovery method for sensor data. Single pass algorithm.</td>
<td>It requires only (O(\log n)), it is a memory; (n) is length of window.</td>
</tr>
<tr>
<td>7.</td>
<td>FW-SM-2008,</td>
<td>FW-SM is an one pass algorithm based on window sliding it uses a new type of structure CP-Tree to. store the frequent itemsets</td>
<td>150s is average execution time for data block size of 20k.</td>
</tr>
<tr>
<td>8.</td>
<td>DSM-FI-2007</td>
<td>Data Stream Mining for Frequent Itemsets is used frequent item discovery scheme, it uses landmark window model.</td>
<td>Average execution time and memory required for data block of size 2000k is: 400s and 30 KB.</td>
</tr>
<tr>
<td>9.</td>
<td>Top-k-FCI-2011</td>
<td>It is a real time single pass algorithm, it uses a new can(T) algorithm for candidate generation</td>
<td>It is efficient for small window of data.</td>
</tr>
</tbody>
</table>
10. INSTANT+ - 2012. + It is a improved type of INSTANT, based on stream landmark model. It uses FP-FOREST to store frequent item set 

11. WIS - 2014 Windowing Itemset shift is based on concept of Apriori and siding window model. 

<table>
<thead>
<tr>
<th>4.4 CONCLUSION</th>
</tr>
</thead>
</table>

The constrained frequent pattern mining has gained importance in recent years because of the necessity to reduce unwanted patterns in many applications. While the field will continue to expand over time, it is hoped that this survey will provide an understanding of the data streams and sequence pattern. In nowadays more high-speed data streams are generated in different application domains, like millions of transactions generated from retail chains, millions of calls from telecommunication companies, millions of ATM and credit card operations processed by large banks, and millions of hits logged by popular Web sites. As most of these problems are solved and more efficient and user friendly mining techniques are developed for end users, it is quite likely that in near future data stream association rule mining play key role in business world.

<table>
<thead>
<tr>
<th>REFERENCES</th>
</tr>
</thead>
</table>


[12] A Mala, F Ramesh Dhanaseelan, Data Stream Mining algorithms- A Review of issues and existing approaches, ICSE.

