ABSTRACT

“Informational Society” is the most proper and fitting term to describe the present day world. An astounding and ever expanding amount of information is pouring in every day which is used for organizing the economy and the society itself. This has paved the way for the collection of such information with the aid of sophisticated technologies and tools such as computers, satellites, remote sensors, and many more. At the outset, much more importance was given to the collection and storage of information as it is an established fact that information leads to power and power in turn leads to success. Unfortunately, these massive collections of data from heterogeneous sources, stored on disparate structures very rapidly became overwhelming. Nowadays even simple transactions such as using a credit card, a phone or browsing the web have become ubiquitous and mammoth. In such scenarios, analysis of the collected information becomes more essential than collection and storage. Thus automatically summarizing the data, extracting the core information and discovering interesting patterns from the vast and ever growing raw data streams has become the need of the hour.

The extraction of hidden predictive information from such large streams of data is commonly referred to as data stream mining. It is a new kind of technology that blends traditional data analysis methods with sophisticated algorithms for processing large volumes of dynamic data. Traditional data analysis tools and techniques cannot be used in such cases because of the massive size of data streams. Also the transactions arrive on the data streams continuously and so they cannot be stored permanently even though memory devices are cheap and readily available. Hence there is an urgent need to develop new schemes for storing and mining these data streams.

The objective of this thesis is to apply data mining techniques to discover frequent patterns from data streams in order to understand and serve better the needs of the society. In this thesis, more efficient pattern mining algorithms that operate on data streams is proposed. Analysis can then be performed to understand the results obtained by the mining algorithms and draw conclusions from them which may help in making better decisions in future.
In this thesis, four algorithms have been proposed to mine the data streams and discover useful information or knowledge from them. The first two algorithms, FIM-CQTransSWin and FIM-CQTimeSWin, mine Frequent Itemsets (FIs) and the other two algorithms, HATCI and FOCIT, mine Closed Frequent Itemsets (CFIs) which is a compressed but lossless form of FIs. Sliding window model is used in three of these algorithms, FIM-CQTransSWin, FIM-CQTimeSWin, and HATCI, since many applications need to concentrate on only the recent transactions without the burden of maintaining all the transactions seen so far. Applications like network analysis help firewalls identify packet fragmentation attacks. Such applications need to concentrate on all data instead of only recent data for which landmark window model is preferred. FOCIT algorithm uses landmark window model and mines CFIs.

Frequent Itemset Mining within a Circular Queue based Transaction-sensitive Sliding Window (FIM-CQTransSWin) algorithm is proposed for mining frequent itemsets from data streams using a transaction-sensitive sliding window model. The sliding window is implemented using a Circular Queue to improve the sliding process and reutilize the same space for incoming transactions. This algorithm uses an effective sliding technique and reduces the space consumption for storing the transactions. But after the window becomes full, every transaction that arrives causes a sliding. As new transactions keep arriving on the data stream, this technique requires a large number of sliding operations and this proves to be less efficient and unnecessary.

To overcome this drawback, Frequent Itemset Mining within a Circular Queue based Time-sensitive Sliding Window (FIM-CQTimeSWin) algorithm is proposed. FIM-CQTimeSWin algorithm uses a time-sensitive sliding window, in which the basic unit of processing is a time unit. This makes the sliding more efficient. The new incoming transactions are accumulated till a complete time unit has arrived. The representation of the time unit is then constructed and only then the sliding operation is performed. This process effectively reduces the number of sliding operations which is the main drawback of the FIM-CQTransSWin algorithm. But still the number of FIs is very large and the time needed to generate and store them is significantly high.
HAsh Table of Closed Itemsets (HATCI) algorithm is proposed to overcome this drawback. It mines only the CFIs which are less in number but still provide all details about the FIs. HATCI algorithm builds a table of Closed Itemsets (CI-Table) to store the Closed Itemsets (CIs) present in the transactions within the current sliding window. Along with this table, three additional tables are maintained for storing the supersets of the CIs (Superset-Table), subsets (Subset-Table) and list of CIs affected by each transaction (Transaction-Table). CI-Table, Superset-Table, and Subset-Table are implemented as hash tables which enable fast searching and retrieval operations. For building and maintaining these tables, a sliding window model is used to enable the user concentrate only on recent transactions and the algorithm also performs only one scan over the data stream.

Sometimes the decisions need to be taken by analyzing the entire data stream from a specified time till the current time. To do this, sliding window model which captures only the recent transactions is not sufficient. In such cases, a landmark window is essential. The proposed Forest of Closed Itemset Trees (FOCIT) algorithm that uses a landmark window model can be used for such applications. FOCIT algorithm performs a single scan over the data stream to build a forest of CI trees. It utilizes a tree organization for efficient representation of CIs that helps make the search and update operations on CIs more efficient. These operations are performed whenever a new transaction arrives on the data stream and the trees are updated incrementally. Incremental updates are very essential in handling data streams because reconstructing entire structure from the scratch for every transaction that arrives on the data stream is infeasible and time consuming as the transactions keep arriving indefinitely and at a fast rate.

On the arrival of a user’s request for CFIs, the CIs with support greater than or equal to the specified support threshold are retrieved and returned to the user. The number of CFIs and the storage needed for them keeps increasing as the window size increases. The time needed is also on the increase as it takes a prohibitively long time to extract CFIs by traversing the FOCIT. Due to the above said drawbacks, landmark windows are preferred only when the application demands it. On the whole, the aim of this thesis is to improve the time and space efficiency of frequent pattern mining over data streams using the proposed algorithms.