ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

Frequent Itemset Mining on Streaming data

Prajakta K. Sarolkar

PG Student, Dept. of Computer Science and Engineering, Government College of Engineering Aurangabad, India

ABSTRACT: Finding frequent itemsets from data streams is one of important tasks of stream data mining. In a time-varying data stream, when the significant change on the frequent itemsets is detected, it is ideal to compute the new frequent itemsets as quickly as possible. Traditional stream data mining algorithms mainly focus on finding the frequent itemsets in one scan, which saves the disk access time. However, the computation cost of processing data is also an important factor in the stream management. In this paper, we propose a new approach that can avoid disk access and meanwhile simplify the computation of data processing. The basic idea is: during the mining of the frequent itemsets, we make every data pass on the significant amount of new coming data rather than scan the old data multiple times. Preliminary experiment results are reported to demonstrate the performance of our approach.

KEYWORDS: FIM, Support, Confidence, Streaming data, Apriori algorithm, Association rule mining.

I. INTRODUCTION

Nowadays, a growing number of applications generate data streams [2, 6], such as telecom call records, web click streams, data collected in sensor networks and etc. Generally, the data streams have the following characteristics: 1) data are coming continuously and at a very rapid rate, 2) the size of data is unbounded, and 3) data (including the patterns hidden in the data) are time-varying. Because the traditional database management system is designed for finite and persistent datasets, the tasks of managing data streams provide researchers with many new challenges and opportunities. In general, the stream data mining research targets on extracting approximate knowledge/patterns by only scanning the data stream once so that the mining results can be obtained as quickly as possible.

Finding frequent itemsets is one of the most important data mining tasks. In some applications, one may be only interested in the frequent itemsets in the recent period of time. For example, in the telecommunication network fault analysis, it is important to know what are the frequent itemsets happening before the faults on the network during the recent period of time. In a time-varying data stream, the frequency of itemsets may change with time. This will invalid some old frequent itemsets and also introduce new ones. When changes are detected (e.g., a large fraction of old frequent itemsets are found no longer valid during the test against new arriving data), the frequent itemsets need to be re-computed to reflect the features of new data and the result should be available as quickly as possible.

In this paper, we investigate the problem of finding frequent itemsets in data streams from a different angle. That is, besides avoiding disk access, we also try to reduce the data processing time. It is well-known that finding accurate frequent itemsets needs multiple scans on the data. Our basic idea is: in a data stream, when enough amount of data are observed, we compute the part of mining result from one scan on these data, then we make the subsequent scans on the new coming data until the complete set of frequent itemsets is discovered. Especially, we use Chernoff bound to find how many transactions are significant enough to show the statistical features of the data stream. Then, using this number, we can partition the streaming data into fragments and the k-th fragment corresponds to a data scan required for finding the k-lengthed frequent itemsets. Thus we only need to scan the whole data stream once and the computation cost is dramatically reduced because in each fragment we only compute a set of fixed-lengthed frequent itemsets.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

II. LITERATURE SURVEY

According to Vanteru Kusuma kumari et al. [1] the need for knowledge discovery from real-time stream data is continuously increasing nowadays and processing of transactions for mining patterns needs efficient data structures and algorithms. They propose a time-efficient Hadoop CanTree- GTree algorithm, using Apache Hadoop. This algorithm mines the complete frequent item sets (patterns) from real time transactions, by utilizing the sliding window technique. These are used to mine for closed frequent item sets and then, association rules are derived. It makes use of two data structures – Can Tree and GTree. The results show that the Hadoop implementation of the algorithm performs 5 times better than in Java.

According to Manisha Thool et al. [2] they propose an integrated online streaming algorithm for solving both problems of finding the top-k elements, and finding frequent elements in a data stream. Their Space-Saving algorithm reports both frequent and top-k elements with tight guarantees on errors. We also develop the notion of association rules in streams of elements. The Streaming-Rules algorithm is integrated with Space-Saving algorithm to report 1-1 association rules with tight guarantees on errors, using minimal space, and limited processing per element and they also implement the Apriori algorithm for static datasets and generated association rules and implement Streaming-Rules algorithm for pair, triplet association rules. They compare the top- rules of static datasets with output of stream datasets and find percentage of error.

According to Danilo S. da Cunha et al. [3] They describes how bacterial colony networks and their skills to explore resources can be used as tools for mining association rules in static and stream data. A new algorithm is designed to maintain diverse solutions to the problems at hand, and its performance is compared to that of other well-known bacteria, genetic, and immune-inspired algorithms: Bacterial Foraging Optimization (BFO), a Genetic Algorithm (GA), and the Clonal Selection Algorithm (CLONALG). Taking into account the superior performance of their approach in static data, they applied the algorithms to dynamic environments by converting static into flow data via a stream data model named sliding-window. They also provide some notes on the running time of the proposed algorithm using different hardware and software architectures.

According to JiaLuo et al. [4] They proposes inter-transaction association rules mining method based on dynamic support threshold. Inter-transaction association rules refer to the association rules between different time periods. Firstly uses the sliding window to limit stream data, then do pre-processing on stream data. In the process of pre-treatment using linearization method fitting to raw data and it reduce the amount of data at the same time, and finally at the end of the pre-processing, generating large transaction grouping method of inter transaction association rules is proposed. They use conceptual data attenuation, thereby reducing the influence of old data to the mining result. Due to artificial setting minimum support threshold may bring many problems, so they presents a method for searching minimum support threshold.
According to AmrAlyAbdElaty et al. [5]A technique to mine an association rules from streaming data efficiently is proposed. The proposed technique develops a tree structure called Fast Update Frequent Pattern Tree (FUFP-Tree) that reduce the number of traversing between tree nodes in both inserting a new transaction and extracting an association rules between items. Also, to avoid congestion during inserting incoming streaming data to FUFP-Tree, a sliding window approach is used to divide incoming data equally to all available windows. The complexity and the performance of this technique are investigated, and a dataset of storehouse is used to test the proposed technique and measure itsefficiency. The efficiency of the proposed technique is compared with other most related algorithms.

According to SamadGahderiMojaveri et al. [6] They proposed an effective approach to mining frequent itemsets used for association rule mining in DS named GRM1. Unlike other semi-graph methods, our method is based on graph structure and has the ability to maintain and update the graph in one pass of transactions. In this method data storing is optimized by memory usage criteria and mining the rules is done in a linear processing time. Efficiency of our implemented method is compared with other proposed method and the result is presented.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

According to Nan Jiang et al. [7] There exist emerging applications of data streams that require association rule mining, such as network traffic monitoring and web click streams analysis. Different from data in traditional static databases, data streams typically arrive continuously in high speed with huge amount and changing data distribution. This raises new issues that need to be considered when developing association rule mining techniques for stream data. They discuss those issues and how they are addressed in the existing literature.

According to Juryon Paik et al. [8] The data mining community has attempted to extract knowledge from the huge amount of data that they generate. However, previous mining work in WSNs has focused on supporting simple relational data structures, like one table per network, while there is a need for more complex data structures. This deficiency motivates XML, which is the current de facto format for the data exchange and modeling of a wide variety of data sources over the web, to be used in WSNs in order to encourage the interchangeability of heterogeneous types of sensors and systems. However, mining XML data for WSNs has two challenging issues: one is the endless data flow; and the other is the complex tree structure. they present several new definitions and techniques related to association rule mining over XML data streams in WSNs. To the best of our knowledge, this work provides the first approach to mining XML stream data that generates frequent tree items without any redundancy.

According to SunithaVanamala et al. [9] Theyproposed an idea to analyze the data stream to identify interesting rare association rules. Rare association rule mining is the process of identifying associations that are having low support but occurs with high confidence. The rare association rules are useful for many applications such as fraudulent credit card usage, anomaly detection in networks,detection of network failures, educational data, medical diagnosis etc. The proposed rare association rule mining algorithm for data stream is implemented using sliding window technique over a stream of data, data is represented in vertical bit sequence format. The advantage of proposed algorithm is that it requires single scan to discover all rare associations. The proposed algorithm outperforms both in terms of memory and time.

According to Ruoming Jin et al.[10] they present a one pass algorithm for frequent itemset mining, which has deterministic bounds on the accuracy, and does not require any out-of-core summary structure. Second, because our one pass algorithm does not produce any false negatives, it can be easily extended to a two pass accurate algorithm. Our two pass algorithm is very memory efficient, and allows mining of datasets with large number of distinct items and/or very low support levels. Our detailed experimental evaluation on synthetic and real datasets shows the following. First, our one pass algorithm is very accurate in practice. Second, our algorithm requires significantly lower memory than Manku and Motwani's one pass algorithm and the multi-pass apriori algorithm. Our two pass algorithm outperforms apriori and FP-tree when the number of distinct items is large and/or support levels are very low. In other cases, it is quite competitive, with possible exception of cases where the average length of frequent itemsets is quite high.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

III. PROPOSED SYSTEM DESIGN

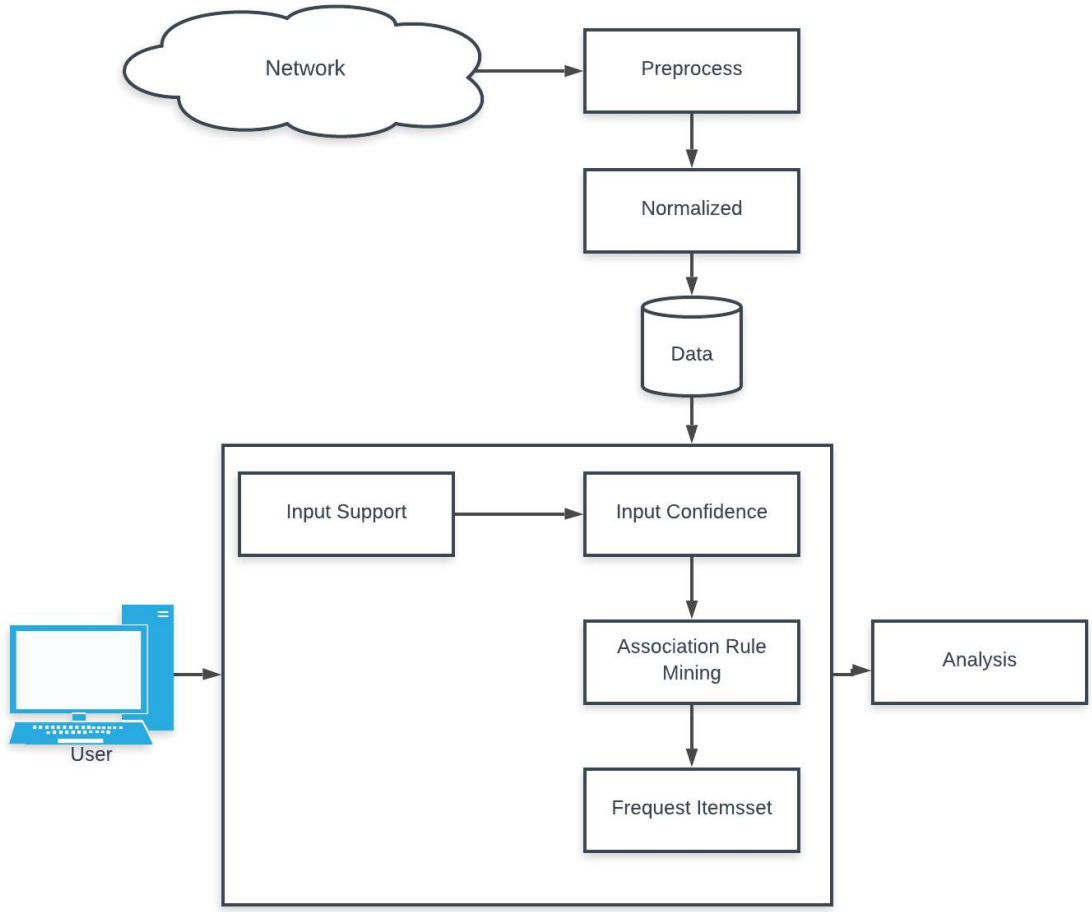

Figure 1: Proposed System Architecture

The propose system deals with frequent itemset mining on on streaming data, basically the main existing system has done the frequent itemset mining on various kind of transactional data sheets. In proposed research below figure 1 illustrates system execution flow in a systematic manner. The Twitter API has used extract runtime data with the help of live streaming. The Java API supports to extract different attributes in Twitter data set. System first deals with data pre-processing as well as data normalisation. Once data has normalised it has stored into buffer dreams, it should be a temporary data tables. The proposed frequent item set mining approach works according to $a,b \rightarrow c$ it means a and b both are the different attributes and c is the class item, finally system generate the frequent rules according to given support and confidence.

Dataset Used

For this research we have download twitter data from runtime serializable environment, around 12 attributes has used for proposed research, below tables show the attributes extracted from twitter data.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

Table 1 : dataset description

No.	Attribute Name	Data type
1	Screen Name	Text
2	Gender	Text
3	Age	Numeric
4	Followers count	Numeric
5	Longitude and latitude	Text
6	Country	Text
7	Zip code	Numeric
8	Friendscount	Numeric
9	Tweetcount	Numeric
10	statuscount	Numeric

The purposed of using this dataset, it is applicable for frequent itemset mining because it we can convert into relational form. Once a set of frequent itemsets has been found, association rules can be generated. Association rules are of the form A→B, and could be read as "A implies B". Each association rule has support (how common the precondition is in the dataset), confidence (how often the precondition leads to the consequence in the dataset) and lift (how much more common the consequence is in instances covered by the rule compared to the whole dataset). For whole scenario our dataset is applicable for find frequent itemset.

Algorithm Design
Algorithm 1: Modified Apriori
Input: Dataset D, Support generation denominator De, min_req_itemsmk;
Output: Generate T item set
Step 1: for all (T in DBi) do
Step 2: items [] ←split (T)
Step 3: for all (item in items []) do
Step 4: if the item is available in FLits
Step 5: Add a [] ← item
Step 6: end for
Step 7: add all a toArrayList<items name, count>All items.
Step 8: Generate the support base on support= (T.count/De)
Step 9: for (k in Array List)
Step 10: if (k.count>=support)
Step 11: FreqItems (k);
Step 12: end for
Step 13: apply step 9 to 12 when reach mk
In the proposed modifies Apriori we have generate minimum we have written new function for generate the association rule mining, the below function has used for generate the rules.

$$A,B \rightarrow C \quad \ldots\ldots\ldots(1)$$

Basically A is the set of first two item set A={age, gender}, and b is the set of another attributes, C consider as a class item of specific record. Around 5000 input records has given to system, with 5-fold, 10-fold and 15-fold cross validation. The various support confidence has used for generate the frequent itemset from given test dataset.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

IV. RESULTS AND DISCUSSION

The experiment is carried out in a machine with Intel core i3-M480 with 2.70 GHz and RAM of 4GB. VMware workstation is installed and two VMs are setup. One VM is for Hadoop 2.6.5 setup with configuration of 25GB hard disk and 512 MB main memory. The other VM is for Eclipse Juno, the virtual configuration is 20GB hard disk and 2 GB main memory.

Below table indicates the time taken in seconds by Proposed Algorithm for the three different datasets of Grocery, Electronic & Sports for different support counts.

Table 1: Performance of Time required in seconds with different support denominator with different dataset

Support Count	Time in Seconds	Time in Seconds	Time in Seconds
Support 5	31	29	30
Support 8	26	24	25
Support 10	22	21	20
support 15	15	16	14

Figure 2 shows the result presented in Table 1 for three different item set of Grocery, Electronic & twitter for various support values of 5,8,10 and 15% respectively. It shows the time required to extract the frequent item set in seconds with different dataset. Utilization is very good in applying the algorithm FIM for frequent item set mining with retail item set for various support values.

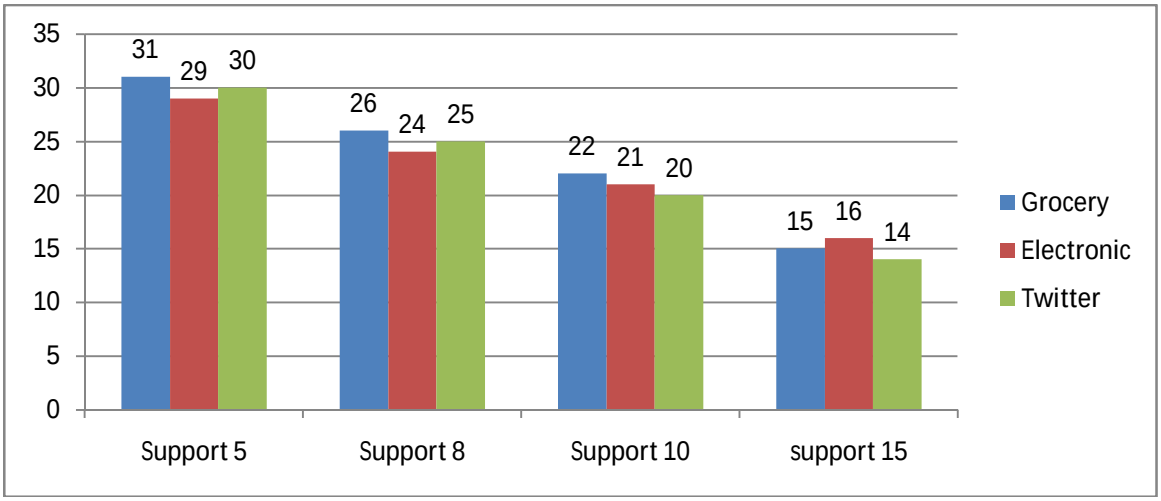

Figure 2: Time required in seconds with different support denominator with different dataset.

The experiment results showed using graphs exhibits that the time used for extraction of frequent item set using FIM decreases as the support increases. Following table indicates the Memory Utilization by frequent item sets in MB by Proposed Algorithm FIM for the three different datasets of Grocery, Electronic & twitter for different support counts.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 6, June 2019

Table 2: Memory Utilization by frequent item sets in MB with different support denominator with different dataset

	Grocery	Electronic	Sport
Support 5	12	11	9
Support 8	9	9	8
Support 10	6	5	5.6
support 15	3	3.2	3.5

V. CONCLUSION

In this paper, we proposed finding frequent itemsets in data streams. To meet the real time requirement, previous research targets on avoiding disk access to reduce the response time. Due to the fact that the response time is related to disk access time and the data process time, we made our attempt from a different angle, i.e., to lower the computation cost of data processing. Based on this consideration, we have proposed a new approach which can avoid disk access and meanwhile simplify the computation of data processing. The basic idea is that data stream is partitioned into fragments (with certain size), from each of which part of mining result is generated. Finally, we report and analyze the preliminary experiment results of our proposed algorithm. One weakness of the proposed approach is that the level of approximation between computed result and accurate result can be affected by the data distribution. In the future work, we will investigate how to guarantee the good approximation of mining result when the data distribution changes.

REFERENCES

[1] Kusumakumari V, Sherigar D, Chandran R, Patil N. Frequent pattern mining on stream data using Hadoop CanTree-GTree. Procedia computer science. 2017 Jan 1;115:266-73.
[2] Thool M, Voditel P. Association rule generation in streams. International Journal of Advanced Research in Computer and Communication Engineering. 2013 May;2(5).
[3] Cunha DS, Xavier RS, Ferrari DG, Vilasbôas FG, de Castro LN. Bacterial Colony Algorithms for Association Rule Mining in Static and Stream Data. Mathematical Problems in Engineering. 2018;2018.
[4] Luo J, Chen S, Pan F, Zhu Y, Wu L, Sun Y, Zhang C. Mining Association Rules from Stream Data Based on the Dynamic Support. In2016 International Conference on Artificial Intelligence: Technologies and Applications 2016 Jan 24. Atlantis Press.
[5] Elaty AA, Salem R, Elkader HA. Efficient Association Rules Mining from Streaming Data with a Fault Tolerance. In2018 13th International Conference on Computer Engineering and Systems (ICCES) 2018 Dec 18 (pp. 627-632). IEEE.
[6] Mojaveri SG, Mirzaeian E, Bornaee Z, Ayat S. New approach in data stream association rule mining based on graph structure. InIndustrial Conference on Data Mining 2010 Jul 12 (pp. 158-164). Springer, Berlin, Heidelberg.
[7] Jiang N, Gruenwald L. Research issues in data stream association rule mining. ACM Sigmod Record. 2006 Mar 1;35(1):14-9.
[8] Paik J, Nam J, Kim UM, Won D. Association Rule Extraction from XML Stream Data forWireless Sensor Networks. Sensors (14248220). 2014 Jul 1;14(7).
[9] Vanamala S, Sree LP, Bhavani SD. Rare association rule mining for data stream. InInternational Conference on Computing and Communication Technologies 2014 Dec 11 (pp. 1-6). IEEE.
[10] Jin R, Agrawal G. An algorithm for in-core frequent itemset mining on streaming data. InFifth IEEE International Conference on Data Mining (ICDM'05) 2005 Nov 27 (pp. 8-pp). IEEE.