ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Sliding Window Based Weighted Maximal Frequent Pattern Mining

Sachin Kathuria[1], Ankit Rai[2], Aniket Rai[2], Nritant Singh[2]

Asst. Professor, Dept. of CSE, GCET, Greater Noida, India[1]

Scholar, Dept. of CSE, GCET, Greater Noida, India[2]

ABSTRACT: The Frequent example mining is one of the essential errands utilized as a part of information mining area and regular information mining methodologies are broadly connected onto static database as well as information stream yet the information have been gathered more rapidly as of late and the relating databases have additionally get to be huger, and subsequently, general regular example mining strategies have been confronted with confinements that don't properly react to the monstrous information, so it is important to lead more proficient and prompt mining errands by examining databases just once. Thus, techniques for productively compacting created examples are required keeping in mind the end goal to tackle that issue. we propose a novel algorithm, weighted maximal frequent pattern mining over data streams based on sliding window model (WMFP-SW) to obtain weighted maximal frequent patterns reflecting recent information over data streams.

KEYWORDS: Data mining, Data stream, Sliding window, weighted maximal frequent pattern mining.

I. Introduction

Data Mining is a procedure of separating concealed example from colossal measure of information, where the information is fundamentally static in nature. It will investigate the verifiable information and concentrate attractive learning. Be that as it may, now a day the information are evolving progressively and becoming quickly. That is the motivation behind why some additional exertion must be given to track the approaching information and dissect legitimately to create fascinating valuable examples. A percentage of the surely understood crucial continuous example mining calculations are as of now proposed in light of BFS and after that FP-Growth taking into account DFS.The surely understood successive example mining (FPM) calculations are apriori in view of broadness first pursuit finds successive designs over static databases and to acquire complete results of successive examples, the calculation ought to output databases over and again and FP-development on the premise of profundity first inquiry conduct mining work with two settled database filters and does not produce applicant designs in contrast with Apriori, but rather it is important to apply incessant example mining in element information streams.

Information streams imply that exchange information are included continually, and subsequently, they have nonstop and boundless features. Data stream mining needs to fulfill the accompanying necessities. (1) Each information component required for information stream examination must be analyzed just once. (2) In spite of the fact that information streams turn out to be always vast as information components are consistently included, memory use for mining operations should be restricted to a worthy and consistent territory. (3) All of the entered information components must be handled at the earliest opportunity. (4) Results of information stream examination ought to be accessible right away as well as their quality ought to likewise be worthy at whatever point clients need the results.

Static Datasets: We have contemplated basic and organized information sets, for example, information in social information bases, value-based information bases and information distribution centers. Those information sources can be considered as organized or semi organized a few times and all are static in nature, where information mining operation can be performed effectively for example assessment. A percentage of the current calculations like Apriori, FP-Growth is adequate to discover the regular example and relationship among information.

Dynamic Datasets: These information sets are fundamentally in complex structure for instance: semi-organized, unstructured, spatial and fleeting, hypertext and mixed media and so on. A portion of the wellsprings of these sorts of datasets are:

- Time-series data relating to stock market.
- Sales forecasting. Utility studies.
- Observation of natural phenomena.
- Banking and Credit card data.
- Social media like Face book, Twitter, Link ,etc World Wide Web.

Every one of these information are monstrous e.g. terabytes in volume, transiently requested, quick changing, and boundless. These information are dynamic in nature furthermore alluded as stream information. Conventional information mining techniques are not proficient to break down element stream datasets. Since it requires various sweeps of the information and hence not pertinent for stream information examination. Not at all like static datasets, stream information streams all through a PC framework ceaselessly. So it is difficult to store the information and output it different times. In this paper another information structure is examined called sliding window model, which is exceptionally helpful where just late occasions might be essential. It additionally lessens memory necessities on the grounds that just a little window of information is put away for examination.

II. FREQUENT PATTERN MINING

A collection of one or more items in a transaction is known as item set. Consider the example T= {beer, bread, chips, diaper} is an item set. An item set whose threshold value is greater than equal to minimum support and confidence is known as frequent item set.

Table 1 Transaction Database

ID	ITEMSET
1	A,B,D
2	A,C,D
3	A,D,E
4	B,E,F
5	B,C,D,E,F

In the above table there are five transactions occurred namely A,B,C,D,E,F. In that A occurred in 3 times. B occurred in 3 times. C occurred in 2 times and D, E, F occurred 3, 4, 2 times respectively. In that example we set 3 as threshold value. So according to the threshold value A, B, D, F are called as frequent items because their occurrence values are greater than the threshold value. C and F are called as infrequent item sets. So they are omitted. Thus this is called as frequent pattern mining.

III. RELATED WORKS

As an early continuous example mining calculation, Apriori finds successive examples over static databases. The calculation performs mining operations in Breadth First Search (BFS) way and needs to create various hopeful examples in the procedure of real incessant examples. In addition, to get complete after effects of incessant examples, the calculation ought to output databases over and over, and particularly in the most pessimistic scenario, the examining errand must be executed the same number of as the quantity of things of the longest exchange in a database. From that point, FP-Growth calculation in view of Depth First Search (DFS) was proposed so as to defeat that issue, and a large portion of the various calculations recommended so far are on the premise of the structure and strategies of FP-development. The calculation can all the more productively direct mining work with two settled database filters and does not create competitor designs in contrast with Apriori. As an early continuous example mining calculation, Apriori finds successive examples over static databases. The calculation performs mining operations in Breadth First Search (BFS) way and needs to create various hopeful examples in the procedure of real incessant examples. In addition, to get complete after effects of incessant examples, the calculation ought to output databases over and over, and particularly in the most pessimistic scenario, the examining errand must be executed the same number of as the quantity of things of the longest exchange in a database. From that point, FP-Growth calculation in view of Depth First Search (DFS) was proposed so as to defeat that issue, and a large portion of the various calculations recommended so far are on the premise of the structure and strategies of FP-development.

The calculation can all the more productively direct mining work with two settled database filters and does not create competitor designs in contrast with Apriori. Finding regular examples in a persistent stream of exchanges is basic for some applications, for example, retail market information examination, system checking, web utilization mining, furthermore, securities exchange forecast. Despite the fact that various continuous .design mining calculations have been produced over the past decade, new answers for taking care of stream information are still required because of the nonstop, unbounded, and requested arrangement of information components created at a quick rate in an information stream. The complete arrangement of late continuous examples is gotten from the tree of the present window utilizing a FP-development mining strategy.

The fundamental commitments of this work are compressed as:

1. We present a novel calculation, WMFP-SW which can proficiently mine WMFPs with one and only look over sliding window-based information stream environment and a tree structure, WMFP-SW-tree utilized for the WMFP mining work. We additionally depict another tree structure, WMFP-tree overseeing WMFP data and performing subset-checking assignments adequately and an exhibit structure, WMFP-SW-cluster for enhancing effectiveness of mining operations. We comprehend mining procedures of the proposed calculation by giving different illustrations.
2. Pruning techniques for diminishing unnecessary mining operations effectively are depicted. Since WMFP-SW considers not just examples' backings additionally their weights when it chooses whether extricated examples are substantial or not, the relating pruning range gets to be ale than that of general incessant example mining. Also, components aside from the most recent ones are rejected in the mining strategy by the sliding window model, and in this way WMFP-SW conducts mining operations with speedier runtime and less memory utilization. We additionally give a methodology which can prune pointless operations bringing about unimportant example era in single ways.
3. To assess execution of the proposed calculation, we contrast our own and past best in class calculations, furthermore, different genuine and engineered datasets applying weight conditions are utilized as a part of execution examinations. These test results demonstrate that WMFP-SW displays more remarkable execution contrasted with the past ones.

IV. ANALYSIS SCHEME

Sliding window-based frequent pattern mining over data streams

The mining techniques taking into account FP-Growth affect static databases and they are not suitable for information streams aggregating information ceaselessly. Since these techniques perform more than two database checks, they don't manage information streams in a flash. Besides, since they build trees with things stayed after rare things are erased, they need to toss beforehand created trees and construct new trees again if new exchange information are included into information streams. In information streams, despite the fact that a specific thing is right now occasional, it can get to be regular one as indicated by expansion of new exchange information. Be that as it may, those two output based strategies must read databases from the primary again since they as of now dispensed with occasional things in the past step.

To understand this, digging strategies suitable for information streams have been proposed, and they can perform mining errands with one and only database examine, along these lines reacting to changes of information streams promptly. After that, sliding window-based incessant example mining approaches have been proposed, which can mine successive examples considering the most recent exchange information of vast information streams. Particularly in those paper, a proficient tree-rebuilding technique, BSM was proposed. Among collected information streams, the most critical components are as of late included information as a rule. In different words, significance of already included information can be brought down or insignificant, while that of recently aggregated ones can be generally higher. In this way, to mirror these attributes, the sliding window model can be connected into mining process. The technique partitions information streams into windows made out of an arrangement of steady measured exchanges and finds successive examples from as of late produced windows, where the measure of windows and the quantity of them can be allotted as different qualities by clients. Through the sliding window-based methodology, we can simply acquire incessant designs reflecting late data.

Maximal frequent pattern mining over data streams

Mining every single incessant example over information streams and additionally static databases can bring about various computational overheads all in all if information sizes are vast. In sliding window-based information stream mining, subsequent to the remaining parts aside from the most recent windows are not viewed as, the overheads can be diminished, be that as it may, we can't at present abstain from bringing on them if the measure of windows then again the quantity of them turns out to be substantial. Thus, the MFP (Maximal Frequent Pattern) documentation, which can pack created incessant designs into a little number of compacted structures, can be used in the mining process, and an assortment of MFP mining vertical bitmap representation was proposed to mine MFPs all the more productively.

The calculation utilizes an extra information structure with a bitmap structure to diminish the quantity of tree traversals. After the Universal Journal of Advanced Research in Computer Engineering and Technology bitmap is built, MAFIA can know example's recurrence through AND operation of the bitmap despite the fact that it doesn't attempt to navigate trees really. FPmax* is a cutting edge MFP mining calculation, where FP-exhibit, an extra information structure for mining MFPs all the more rapidly, was proposed, in this manner diminishing tree traversal times impressively. Since FP-exhibit has data of examples' underpins, the calculation can ascertain them ahead of time some time recently trees are really navigated when development procedures are performed.

Thus, this system not just can lessen tree traversal operations successfully additionally can improve pruning productivity by forestalling era of unnecessary restrictive trees. Notwithstanding, subsequent to the above calculations have two sweep based procedures, they are not suitable for the information stream mining.

Applying weight conditions into frequent pattern mining over data streams

Every thing existing in information streams has interesting significance (on the other hand weight). Weights of things in information streams are utilized as a part of the mining process after they are changed over into standardized values inside of a specific extent. The reason is that if a weight of any thing is too vast, it is difficult to indicate its weighted backing as a limited number of digits. The principle test of applying weights is to keep up the counter monotone property. Be that as it may, the application for the most part pulverizes that property since weighted occasional examples can get to be weighted successive ones as example development operations are led. For this reason, specialists have attempted endeavors to keep up the hostile to monotone property, and an assortment of techniques mines weighted successive examples over information stream environment taking into account the sliding window model. The calculation conducts tree rebuilding work with the BSM method and gives the latest mining results from the sliding window at whatever point clients demand them. In this study, the structure of the proposed calculation, WMFP-SW(Weighted Maximal Frequent Pattern-Sliding Window) is in light of the best in class MFP mining calculation, FPmax* also, the remarkable tree rebuilding method, BSM.

Calculation–1

Sliding window-based frequent pattern mining over data streams

The FP-Growth methodology is productive for static databases, yet it is not suitable for information streams with persistent information stream. The FP-Growth technique checks the dataset more than two times subsequently they don't manage information streams immediately. The calculation develops tree with things stayed after rare things are erased, they need to dispose of already produced trees and fabricate new trees again if new exchange information are included the information stream. The two output based technique must read databases from the primary again since they as of now dispensed with rare things in the past step. To tackle this digging strategy suitable for information streams have been proposed and they can perform mining assignment with stand out database examine, in this way reacting to changes of information stream instantly.

After that, sliding window-based regular example mining approaches have been proposed, which can mine incessant examples considering the most recent exchange information of substantial information streams. Among gathered information streams, the most imperative components are as of late included information as a rule. As it were, significance of beforehand included information can be brought down or aimless, while that of recently collected ones can be moderately higher. In this manner, to mirror these qualities, the sliding window model can be connected into mining process. The strategy separates information streams into windows made out of an arrangement of consistent estimated exchanges and finds incessant examples from as of late produced windows, where the measure of windows and the quantity of them can be allotted as different qualities by clients. Through the sliding window-based methodology, we can continuously acquire regular examples reflecting late data.

Calculation – 2

Maximal frequent pattern mining over data streams

Mining every continuous example over information streams may prompts various computational overheads as a rule if information sizes are gigantic. In sliding window-based information stream mining, subsequent to the remaining parts aside from the most recent windows are definitely not considered, the overheads can be decreased, however we can't in any case abstain from bringing about them if the span of windows or the quantity of them turns out to be vast. FP max is a best in class MFP mining calculation, where FP-cluster, an extra information structure for mining MFPs all the more rapidly, was proposed, subsequently diminishing tree traversal times impressively. Since FP-exhibit has data of examples' backings, the calculation can ascertain them ahead of time before trees are really crossed when development procedures are performed. Thus, this procedure can lessen tree traversal operations successfully as well as can upgrade pruning proficiency by counteracting era of unnecessary restrictive trees. Be that as it may, subsequent to the above calculations have two sweep based procedures, they are not suitable for the information stream mining.

Calculation – 3

Applying weight conditions into frequent pattern mining over data streams

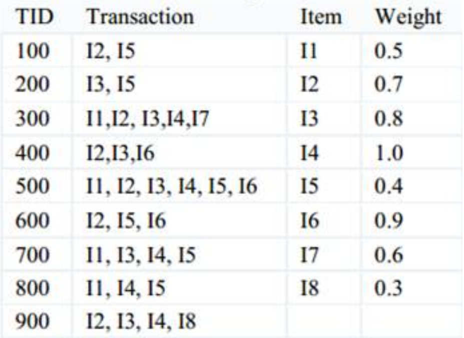
Every thing existing in information streams has certain weight. Case in point, given things over retail information streams, support data of them mean their business volume, and their weight data speaks to costs or benefits for every thing.

Hence, when both of those two components are considered, we can pick up mining results reflecting complex variables in the true. Weights of things in information streams are utilized as a part of the mining process after they are changed over into standardized values inside of a specific reach. The reason is that if a weight of any thing is too huge, it is difficult to mean its weighted support as a limited number of digits. The primary test of applying weights is to keep up the counter monotone property.

In any case, the application by and large wrecks that property since weighted rare examples can get to be weighted regular ones as example development operations are directed. Thus, specialists have tried endeavors to keep up the against monotone property, and an assortment of techniques. WFPMDS mines weighted incessant examples over information stream environment in view of the sliding window model. The calculation conducts tree rebuilding work with the BSM strategy and gives the latest mining results from the sliding window at whatever point clients demand them. In this study, the system of the proposed calculation, WMFP-SW depends on the cutting edge MFP mining calculation.

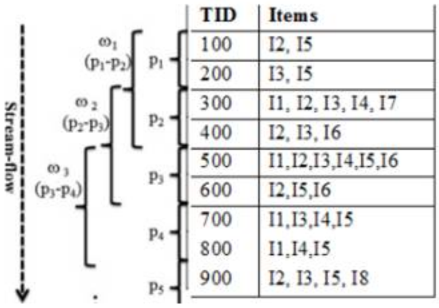
Calculation-4

Weighted maximal frequent pattern mining over data streams based on sliding window model



TID	Transaction	Item	Weight
100	I2, I5	I1	0.5
200	I3, I5	I2	0.7
300	I1,I2, I3,I4,I7	I3	0.8
400	I2,I3,I6	I4	1.0
500	I1, I2, I3, I4, I5, I6	I5	0.4
600	I2, I5, I6	I6	0.9
700	I1, I3, I4, I5	I7	0.6
800	I1, I4, I5	I8	0.3
900	I2, I3, I4, I8		

Table 2: Data Stream with weighted information of data


Stream-flow
ω_1 $(p_1\text{-}p_2)$
ω_2 $(p_2\text{-}p_3)$
ω_3 $(p_3\text{-}p_4)$
p_1
p_2
p_3
p_4
p_5

TID	Items
100	I2, I5
200	I3, I5
300	I1, I2, I3, I4, I7
400	I2, I3, I6
500	I1,I2,I3,I4,I5,I6
600	I2,I5,I6
700	I1,I3,I4,I5
800	I1,I4,I5
900	I2, I3, I5, I8

Fig 1: Sliding window based data stream derived from table 2

In the above scenario both the window size (i.e. the number of panes) and pane size (i.e. the number of items) are set as 2. In the data stream, sliding window process is performed as follows. We first read p1 and p2 to fill ω1. After that, when reading the next pane, p3, we remove the old pane, p1 . Then, p2 and p3 belong to the current window, ω2. In the same manner, the old pane, p2 is deleted and the new pane p4 is entered into the current window, ω3 in the next step. As a result, the current window always has the latest data stream information, and thus, the sliding window-based mining methods can instantly provide users with frequent pattern results considering the most recent data whenever they request mining results.

V. CONCLUSION

In this paper we have contemplated the idea of information streams what's more, how the continuous examples are mined over information streams. In expansion to this, we have dissected the diverse existing research works of incessant example mining over information streams. The benefits, bad marks and future upgrades of the current works are additionally talked about. In future, we will grow new strategies and calculations for finding continuous examples over information streams which beats the disadvantages of the existing procedures furthermore we overview the strategies and calculations for mining weighted maximal continuous examples over information streams in view of the sliding window idea, which can perform mining operations concentrating on as of late amassed parts over information streams and these information stream mining techniques can separate regular examples over information streams adequately.

REFERENCES

1. Gangin Lee , Unil Yun , Keun Ho Ryu," Sliding window based weighted maximal frequent pattern mining over streams" Expert Systems with Applications vol. 41, pp. 694–708, 2014.
2. Chen, Y., Bie, R., & Xu, C. "A new approach for maximal frequent sequential patterns mining over data streams". International Journal of Digital Content Technology & its Application.Vol. 5(6), pp. 104–112, 2011.
3. Agrawal, R., & Srikant, R. "Fast algorithms for mining association rules. In Proceedings of the 20th international conference on very large databases" pp. 487–499, September 1994.
4. Ahmed, C. F., Tanbeer, S. K., Jeong, B. S., & Lee, Y. K. "An efficient algorithm for sliding window-based weighted frequent pattern mining over data streams".IEICE Transactions, Vol. 92-D(7),pp. 1369–1381, 2009.
5. Chen, Y., Bie, R., & Xu, C." A new approach for maximal frequent sequential patterns mining over data streams". International Journal of Digital Content Technology and its Applications, Vol.5 (6), pp.104–112, 2011.
6. Chen, H., Shu, L., Xia, J., & Deng, Q. "Mining frequent patterns in a varying-size sliding window of online transactional data streams". Information Sciences Vol.215, pp.15–36, 2012.
7. Han, Jiawei, Jian Pei, and Yiwen Yin. "Mining frequent patterns without candidate generation." ACM SIGMOD Record. Vol. 29. No. 2. ACM, 2000.
8. Lee, Gangin, Unil Yun, and Keun Ho Ryu. "Sliding window based weighted maximal frequent pattern mining over data streams." Expert Systems with Applications 41.2 (2014): 694-708.
9. Farzanyar, Zahra, Mohammadreza Kangavari, and Nick Cercone. "Max-FISM: Mining (recently) maximal frequent itemsets over data streams using the sliding window model." Computers & Mathematics with Applications 64.6 (2012): 1706-1718.
10. Ahmed, Chowdhury Farhan, et al. "An efficient algorithm for sliding window-based weighted frequent pattern mining over data streams." IEICE TRANSACTIONS on Information and Systems 92.7 (2009): 1369-1381.
11. Ahmed, Chowdhury Farhan, et al. "Single-pass incremental and interactive mining for weighted frequent patterns." Expert Systems with Applications 39.9 (2012): 7976-7994.

12. Chen, Hui, et al. "Mining frequent patterns in a varying-size sliding window of online transactional data streams." Information Sciences 215 (2012): 15-36.
13. Deypir, Mahmood, Mohammad Hadi Sadreddini, and Sattar Hashemi. "Towards a variable size sliding window model for frequent.
14. Farzanyar, Zahra, Mohammadreza Kangavari, and Nick Cercone. "Max-FISM: Mining (recently) maximal frequent itemsets over data streams using the sliding window model." Computers & Mathematics with Applications 64.6 (2012): 1706-1718.
15. Gouda, Zaki, M. J., "GenMax: An efficient algorithm for mining maximal frequent item sets", Data Mining and Knowledge Discovery, vol. 11(3), pp. 223–242, 2005.
16. Luo, C., & Chung, S. M., "A scalable algorithm for mining maximal frequent sequences using a sample", Knowledge and Information Systems, vol. 15(2), pp. 149–179, 2008.
17. Tanbeer, S. K., Ahmed, C. F., Jeong, B. S., & Lee, Y. K., "Efficient single-pass frequent pattern mining using a prefix-tree", Information Sciences, vol. 179(5), pp. 559–583, 2009.
18. Yang, C., Li, Y., Zhang, C., & Hu, Y., "A novel algorithm of mining maximal frequent pattern based on projection sum tree", Fuzzy Systems and Knowledge Discovery, vol. 1, pp. 458–462, 2007.
19. Zhi-Hong Deng, "Fast mining Top-Rank-k frequent patterns by using Node-lists", Expert Systems with Applications, Vol. 4, pp. 1763–1768, 2014.
20. Mhmood Deypir, Mohammad Hadi Sadreddini, "An Efficient Sliding Window Based Algorithm for Adaptive Frequent Item set Mining over Data Streams", Journal of Information Science and Engineering, Vol. 29, pp. 1001-1020, 2013.
21. Yunyue Zhu, Dennis Shasha, "StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time", VLDB, Vol. 15, 2002.
22. Caiyan Dai, Ling Chen, "An Algorithm for Mining Frequent Closed Item sets in Data Stream", Physics Procedia, Vol. 24, PP. 1722 – 1728, 2012.

BIOGRAPHY

Ankit Rai, Aniket Rai, Nritant Singh are students (scholars) in the Computer Science Department, Galgotias College Of Engineering And Technology. They are pursuing Bachelor degree in Computer Science from Galgotias College of Engineering And Technology. Their research interests are Data Mining and Pattern Mining.