JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-6, June 2018 E-ISSN: 2347-2693

# High Utility Sequential Pattern Mining over Data Streams with Sliding Window Control

**S. M. V. Sirisha[1], V. MNSSVKR Gupta[2]**

[1] Department of Computer science and Engineering, SRKR Engineering College, Bhimavaram, India
[2] Department of Computer science and Engineering, SRKR Engineering College, Bhimavaram, India

**Available online at: www.ijcseonline.org**

Accepted: 23/Jun/2018, Published: 30/Jun/2018

***Abstract***- Mining valuable examples from successive information are a testing subject in information mining. An essential undertaking for mining successive information is consecutive example mining, which finds arrangements of thing sets that as often as possible show up in a grouping database. In consecutive example mining, the determination of arrangements is by and large in light of the recurrence/bolster structure. Nonetheless, the vast majority of the examples returned by successive example mining may not be sufficiently educational to representatives and are not especially identified with a business objective. In perspective of this, high utility sequential pattern (HUSP) mining has risen as a novel research point in information mining as of late. The principle goal of HUSP mining is to separate important and valuable successive examples from information by considering the utility of an example that catches a business objective (e.g., benefit, client's advantage). In HUSP mining, the objective is to discover successions whose utility in the database is no not as much as a client indicated least utility edge.
Assembling arrange for which enables the company to expand its income, high utility example mining is an essential viewpoint. A lot of stream information identified with client buys conduct utilized for building up assembling design. Ongoing inclination of the clients likewise helps in producing fabricating plans. This review work contains a rundown structure and a novel calculation for producing high utility example over expansive information, based on Sliding Window Control Mode. This approach maintains a strategic distance from the age of hopeful example. Because of that calculation not required a lot of memory space and in addition computational assets for checking hopeful examples. Because of this current, it's exceptionally proficient approach.

***Keywords***- Sliding window based utility pattern mining, Manufacturing plan, Industrial systems

## I. Introduction

The fundamental piece of mining is one of the method for Frequent thing sets in information mining designs assume a huge part in different undertaking identified with information mining that find intriguing examples from databases to be specific affiliation rules, classifiers, relationships, bunches, arrangement, and so on. Visit design mining is the fundamental procedure in mining the affiliation govern, where in visit designs are thing sets that as often as possible exist in an informational collection with higher recurrence than a client determined limit.

[2] Information disclosure inside the database is the fundamental point of information mining. KDD works for finding fascinating data from a lot of huge information in a programmed way. To break down complex mechanical information and discover answers for learning related issues. Arrangement and example mining like methodologies has been produced for mechanical zone databases. To investigate complex information and discover significant example information, design mining has been mostly utilized. Basic example mining, visit design mining approaches is utilized when the accessibility of important data identified with recurrence and least help show. On the off chance that consider design is an imperative data than their recurrence is a help and it's more prominent than or equivalent to the given least help. [3] Inside FPM, item parameters data does not show separately on the grounds that it's expected that all things have parallel data. In this manner it's exceptionally mind boggling to examine mechanical information with item parameter like benefit, cost, and amount bought by the client.

Disclosure of thing set with a high utility like benefits of the value-based database required for mining high utility thing sets. A substantial number of calculations are available which utilized for creating high utility example yet it by implication produce an expansive number of applicant design likewise because of that mining execution of the framework is a reduction in term of execution time and space necessity. At the point when the database contains an extensive number of exchange and long high utility thing set than execution parameter turn out to be more awful.

 

For expanding income of the plant, it's imperative for a supervisor to perceiving producing plan as indicated by client require. For that reason, the director has to know which item produce high benefit inside the plant. To accomplishing this point, the chief needs to utilize high utility example mining approach. Example data identified with the arrangement of things which are beneficial, exhibit towards supervisor likewise helps in producing more gainful plans. Because of that generation of this item increment and income of the plant can be augmented.

There are three fundamental techniques display in the example mining territory, sliding window, damped window, historic point window. Inside sliding window strategy stream information isolates in to various pieces called groups. Ongoing groups are utilized for design mining approach. In the damped window, new approaching information regarded as more imperative than the past one. Because of that significance of information rots over the long haul. Inside historic point window strategy, its solitary utilizes the information in the middle of particular day and age like day and age between current time point and particular time point.

[4] Inside this approach novel, information structure named SHUP-List has proposed chiefly utilized for putting away late clumps data. Additionally, it's containing a proposed calculation for effectively refreshing outstanding utilities. For lessening look space its present pruning strategies. Additionally presenting proposed calculation named as SHUPM (Sliding Window based High Utility Pattern Mining).

## II. Related work

In this segment, we display the diverse approach and procedures given by various creators with respect to sliding windows control based example mining is as take after.

[1] Propose a rundown structure and novel calculation for discovering high utility example over information stream based on sliding window mode. In this approach, competitor designs are not produced. Because of that extensive memory space and more computational assets not required for competitor design check. The calculation utilized inside this approach work effectively. In this, different continuous databases are utilized for the exploratory reason for producing an outcome as far as runtime, memory ease of use, and scalability.

[2] Specifies two calculations for producing an example from the database. Its shows how the best highlights of the two proposed calculation can be consolidated into one cross breed calculation called Apriori Hybrid. For a substantial number of exchanges, it's essential to approach. Apriori Hybrid calculation having amazing scale-up property.

The novel tree structure for proficiently performs incremental and intuitive HUP mining. Incremental HUP lexicographic tree is the principal tree structure masterminded by a thing's lexicographic request. With no recreating activity, it can catch incremental information. IHUP exchange recurrence tree is the second tree structure. By orchestrating things as indicated by their exchange recurrence, it's getting a conservative size. It's likewise decreases mining time. IHUP-Transaction weighted use tree depends on the TWU estimations of things in plummeting request. It's extremely productive structure for adaptable, incremental mining [3].

TABLE I
EXAMPLE DATABASE

$W_1$ ($B_1$-$B_3$)
$W_2$ ($B_2$-$B_4$)

| | TID | Items | TU |
|---|---|---|---|
| $B_1$ | $T_1$ | A (2), B (3), D (2), E (2) | 19 |
| | $T_2$ | A (1), C (2), D (1) | 12 |
| $B_2$ | $T_3$ | C (1), E (4) | 11 |
| | $T_4$ | B (2), C (1), E (2) | 9 |
| $B_3$ | $T_5$ | B (1), D (2) | 9 |
| | $T_6$ | C (5), D (1), E (1) | 21 |
| $B_4$ | $T_7$ | A (2), B (2), E (1) | 8 |
| | $T_8$ | A (3), C (1), D (2), E(1) | 19 |

TABLE II
EXTERNAL UTILITIES OF ITEMS

| Item | A | B | C | D | E |
|---|---|---|---|---|---|
| External utility | 2 | 1 | 3 | 4 | 2 |

A novel strategy Temporal High Utility itemset mine propose for mining transient high utility itemset from extensive dataset [4]. The fundamental commitment of this approach is to decrease the age of hopeful itemset and viably distinguish the transient high utility thing sets. Because of that execution time can be decreased in mining. With this approach, thing sets can be accomplished adequately with less memory space and execution time.

An approach in view of the investigation of thing co-events to decrease the quantity of join tasks that should be performed [5]. This approach is called as Fast High Utility Miner (FHM). It's six times quicker than the best in class calculation and in addition it lessens the quantity of join activities.

[6] Propose a calculation called High Utility Itemset-Miner. It's utilized for High Utility Itemset Mining. It's having a novel structure called utility rundown, utilized for putting away utility data around an itemset and the heuristic data for pruning the inquiry space of HUI-Miner. Its effectively mine high utility itemset from the utility rundown created from the dataset. It's keeping away from exorbitant age of utility calculation. Paper likewise contains a correlation of HUI-Miner with Stat-of-the-Art calculations on different databases. The aftereffect of this calculation as far as running time and memory utilization are clarified.

 

In these paper [7] creator Present Two-Phase calculations for effectively prune down the quantity of applicants and correctly acquire the entire arrangement of high utility itemset. It's extremely effective as far as speed and memory cost on a genuine database. It's additionally proficient for expansive databases.

A novel calculation called FP-CDS that can catch all successive shut itemsets and another capacity structure called FP-CDS tree [8]. It's utilized for changing advancement of itemsets frequencies after some time. For refreshing units, a few fundamental windows of the point of interest window utilized. Itemsets in every essential window are mined and put away in FP-CDS tree based procedures.

An Explained two calculations, utility example development (UP-Growth) and UP-Growth+, for mining high utility itemset with compelling methodologies [9]. It's having an information structure called UP-Tree utilized for mining high utility itemsets data. Because of that competitor itemsets can be created proficiently with two output on database. In this paper correlation between UP-Growth and UP-Growth+ introduced. This approach chips away at hopeful example lessening as well as its expansion the general execution in term of runtime when database contains bunches of long exchanges.

[10] Explain technique for mining affiliation decides that mirror the conduct of past clients was proposed for a versatile web crawler. RDF demonstrates utilized for recovering clients practices. Utilizing affiliation rules client conduct removed effectively..

## III. Problem statement

This work is motivated by the accompanying issue in mechanical zones. In an assembling plant, an administrator needs to redesign an assembling design with the goal that the income of the plant can be boosted. Thusly, the supervisor needs to recognize what items make high benefits for the plant. Keeping in mind the end goal to accomplish this reason, HUPM is an appropriate arrangement. On the off chance that the supervisor gets design data demonstrating that an arrangement of specific items makes high benefit, the director can build up more gainful assembling designs, which increment the generation of these items so the income of the plant can be expanded.

## IV. Implementation Methodology

In this paper, we propose a novel calculation for effectively mining up and coming high utility examples based on the sliding window system. The fundamental commitments of this paper are as per the following. 1) Proposing a novel rundown structure named SHUP-List that can keep up the data of ongoing clusters and handle them group by bunch keeping unwavering quality. 2) Suggesting a novel methodology that permits the proposed calculation to effectively refresh remaining utilities in SHUP-Lists as indicated by a TWU rising request 3) Introducing another pruning strategy so as to decrease the inquiry space of HUPM in view of the sliding window display 4) Demonstrating that the proposed calculation named SHUPM (Sliding window based High Utility Pattern Mining) has much preferred execution over best in class calculations based on the broad trial comes about.

A sliding window strategy separates stream information into numerous lumps called groups and uses just ongoing bunches for mining designs. The principle system of given Proposal contain following:

**Procedure:**
**Step1:** Proposing a novel list structure named SHUP-List that can maintain the information of recent batches and handle them batch by batch keeping reliability.

- Capturing information of recent batches in list structures.
- A list of each pattern is composed of one or more entries representing transactions, which contain the Pattern as its sub-pattern.

**Step2:** Suggesting a novel strategy that allows the proposed algorithm to efficiently update remaining utilities in SHUP-Lists according to a TWU ascending order.
**Step3:** High utility pattern mining with sliding window technique. Introducing a new pruning technique by combining two lists in order to reduce the search space of HUPM based on the sliding window model.
**Step4:** Analysis of the given pattern mining algorithm.
**Step5:** Performance evaluations with respect to previous techniques such as SHU-Growth etc.

- Run Time Test
- Memory Usage Test
- Scalability Test

With the help of this tests, performance efficiency of given approach is find out.

## V. High Utility pattern Mining with Sliding Window

In this paper, we investigate the issue of proficiently mining high utility itemsets in worldly databases like information streams. We propose a calculation named THUI-Mine that can find fleeting high utility itemsets from information streams productively and successfully. The hidden thought of THUI-Mine calculation is to coordinate the benefits of Two-Phase calculation and SWF calculation and expand with the incremental digging procedures for mining worldly high utility itemsets proficiently.

 

The novel commitment of THUI-Mine is that it can effectively recognize the utility itemsets in information streams with the goal that the execution time for mining high utility itemsets can be considerably lessened. That is, THUI-Mine can find the transient high utility itemsets in current time window and furthermore find the worldly high utility itemsets in whenever window with constrained memory space and less calculation time by sliding window channel strategy. Along these lines, the way toward finding all fleeting high utility itemsets under unequaled windows of information streams can be accomplished viably under restricted memory space, less competitor itemsets and CPU I/O. This meets the basic necessities of time and space proficiency for mining information streams. Through test assessment, THUI-Mine is appeared to create less hopeful itemsets in finding the transient high utility itemsets, so it beats different techniques regarding execution efficiency.Moreover, it additionally accomplishes high adaptability in managing vast databases. To our best information, this is the main work on mining fleeting high utility itemsets from information streams.

THUI-Mine depends on the rule of Two-Phase calculation [14], and we expand it with the sliding-window sifting strategy and spotlight on using incremental strategies to enhance the reaction time with less applicant itemsets and CPU I/O. Basically, by dividing an exchange database into a few allotments from information streams, calculation THUI-Mine utilizes a sifting edge in each segment to manage the exchange weighted usage itemsets age. The 66ycumulative data in the earlier stages is specifically continued toward the age of exchange weighted usage itemsets in the ensuing stages by THUI-Mine.

In the preparing of a segment, a dynamic exchange weighted usage set of itemsets is created by THUI-Mine. Expressly, a dynamic exchange weighted use set of itemsets is made out of the accompanying two kinds of exchange weighted use itemsets, i.e., (1) the exchange weighted usage itemsets that were extended from the past dynamic applicant set in the past stage and stay as exchange weighted use itemsets after the present parcel is mulled over and (2) the exchange weighted use itemsets that were not in the dynamic hopeful set in the past stage however are recently chosen after just considering the present information segment As such, after the preparing of a stage, calculation THUI-Mine yields a combined channel, signified by CF, which comprises of a dynamic exchange weighted use set of itemsets, their event tallies and the comparing fractional help required. With these outline contemplations, calculation THUI-Mine is appeared to have great execution for mining transient high utility itemsets from information streams.

**Algorithm**

**Input:** a new transaction Si, window size w, minimum utility threshold δ, ItemUtilLists, HUSP-Tree

**Output:** ItemUtilLists, HUSP-Tree, HUSPs

1: if i ≤ w (when Si is a transaction in the first window) then

2: ∀ item ∈ Si, put(r, i, u(item, Si)) to ItemU tilLists(item)

r r

3: if i = w then

4: Construct HUSP-Tree using ItemUtilLists and δ

5: end if

6: else

7: Update ItemUtilLists and HUSP-Tree using Si, w and δ

8: end if

9: if the user requests to get HUSPs for the current window then

10: Find all the HUSPs from the potential HUSPs stored in HUSP-Tree using δ

11:end if

12: Return ItemU tilLists, HU SP − T ree, HU SP s if requested

The calculation incorporates three principle stages: (1) Initialization stage, (2) refresh stage and (3) HUSP mining stage. The introduction stage applies when the information exchange has a place with the principal sliding window. In the introduction stage (lines 1-5), the ItemUtilLists structure is built for putting away the utility data for each thing in the info exchange Sir. At the point when there are w exchanges in the principal window, HUSPTree is developed for the primary window. In the event that there are as of now w exchanges in the window when the new exchange Sir arrives, Sir is added to the window and the most seasoned exchange in the window is expelled. This is finished by incrementally refreshing the ItemUtilLists and HUSP-Tree structures on line 6, which is the refresh period of the calculation. After the refreshing stage, if the client solicitations to discover HUSPs from the new window, HUSPStream restores all the HUSPs to the client by navigating HUSPTree once.

## VI. Experimental evaluation

The proposed THUI-Mine algorithm can be best understood by the illustrative transaction database in Table 1 and Figure where a scenario of generating high utility itemsets from data streams for mining temporal high utility itemsets is given. We set the utility threshold as 120 with nine transactions. Without loss of generality, the temporal mining problem can be decomposed into two procedures:

1. Preprocessing procedure: This procedure deals with mining on the original transaction database.
2. Incremental procedure: The procedure deals with the update of the high utility itemsets form data streams.

| $P_1$ | | |
|---|---|---|
| $C_2$ | START | TRANSACTION WEIGHTED UTILTY |
| AB AD | 1 | 0 |
| AE BD | 1 | 42 |
| BE | 1 | 0 |
| DE | 1 | 71 |
| | 1 | 71 |
| | 1 | 71 |

| $P_2$ | | |
|---|---|---|
| $C_2$ | START | TRANSACTION WEIGHTED UTILTY |
| AB | 2 | 48 |
| AD | 1 | 42 |
| AE | 2 | 48 |
| BD | 1 | 123 |
| BE | 1 | 119 |
| DE | 1 | 71 |

| $P_3$ | | |
|---|---|---|
| $C_2$ | START | TRANSACTION WEIGHTED UTILTY |
| AB | 2 | 88 |
| AC | 3 | 67 |
| AD | 3 | 27 |
| AE | 2 | 75 |
| BC | 3 | 40 |
| BD | 1 | 123 |
| BE | 1 | 224 |
| CE | 3 | 27 |

| $db^{1,3\text{-}v=D}$ | | |
|---|---|---|
| $C_2$ | START | TRANSACTION WEIGHTED UTILTY |
| AB | 2 | 88 |
| AC | 3 | 67 |
| BC | 3 | 40 |
| BD | 2 | 52 |
| BE | 2 | 153 |

| $D+V^{+}=db^{2,4}$ | | |
|---|---|---|
| $C_2$ | START | TRANSACTION WEIGHTED UTILTY |
| AB | 2 | 88 |
| AC | 3 | 67 |
| BC | 3 | 123 |
| BD | 4 | 42 |
| BE | 2 | 153 |
| CD | 4 | 0 |

Figure: Temporal high utility item sets generated from data streams by THUI-Mine.

From above Figure, it is noticed that since there are additionally 3 exchanges in P2, the separating edge of those itemsets did from the past stage is 40 + 40 = 80 and that of recently recognized applicant itemsets is 40. It can be seen from Figure 2 that we have 4 transient high exchange weighted usage 2-itemsets in C2 after the preparing of parcel P2, and 2 of them are conveyed from P1 to P2 and 2 of them are recently recognized in P2.

At long last, segment P3 is handled by calculation THUI-Mine. The subsequent fleeting high exchange weighted use 2-itemsets are {AB, AC, BC, BD, BE} as appeared in Figure 2. Note that however showing up in the past stage P2, itemset {AE} is expelled from transient high exchange weighted usage 2-itemsets once P3 is considered since its exchange weighted utility does not meet the sifting limit at that point, i.e., 75< 120. In any case, we do have two new itemset, i.e., AC and BC, which join the C2 as worldly high exchange weighted usage 2-itemsets. Subsequently, we have 5 worldly high exchange weighted use 2-itemsets created by THUI-Mine, and 2 of them are conveyed from P1 to P3, 1 of them is conveyed from P2 to P3 and 2 of them are recently distinguished in P3. Note that rather than 10 hopeful itemsets that would be created if Two-Phase calculation were utilized, just 5 worldly high exchange weighted usage 2-itemsets are produced by THUI-Mine. In the wake of preparing P1 to P3, those transient high exchange weighted use itemsets in db1,3 are {A, B, C, D, E, AB, AC, BC, BD, BE}.

**Dataset Based Results**

| * | Product | Brand | Year | Price | Quantity | Rank |
|---|---|---|---|---|---|---|
| 1 | computers | Dell | 2016 | 32176 | 53 | Average |
| 2 | computers | Dell | 2014 | 74412 | 67 | Average |
| 3 | computers | Dell | 2004 | 61149 | 61 | Worst |
| 4 | computers | Dell | 2018 | 31317 | 47 | Worst |

Figure: Based on Product selection ranks

 

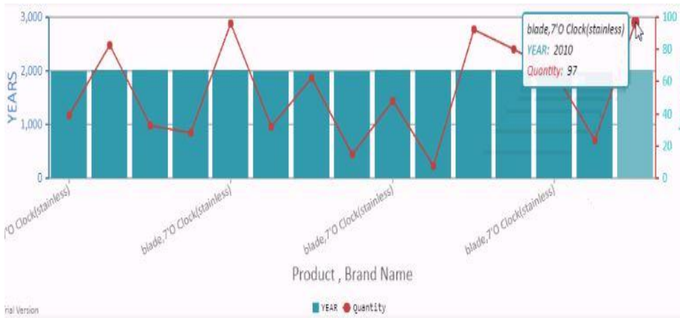

Figure: Analysis product sales in the years

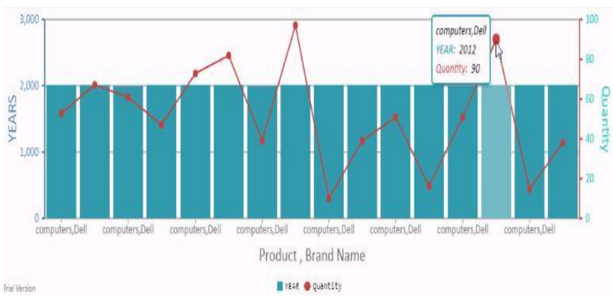

Figure: Analysis product sales in the year's different category

## VII. Conclusion

Here in this paper, we review another approach for design mining. Sliding Window Control Based High Utility Pattern Mining approach contains a novel rundown structure named SHUP-List for putting away mined information. Additionally, it's proficiently refreshing residual utilities. The approach guarantees to, comprehend the ongoing buy inclinations of clients. In this way, it is reasonable for setting up productive assembling designs in different modern zones and also Algorithm does not expend enormous computational assets for confirming competitor designs since it can maintain a strategic distance from the age of applicant designs. In this framework, proposed for removing and understanding the client designs from the gigantic information that produces the helpful examples utilizing Sliding Window techniques and the successive things calculation for powerful example acknowledgment, to discover progressive data.

## References

[1] Yun, Unil, Gangin Lee, and Eunchul Yoon. "Efficient High Utility Pattern Mining for Establishing Manufacturing Plans with Sliding Window Control." IEEE Transactions on Industrial Electronics, 2017.

[2] R. Agrawal and R. Srikant. "Fast Algorithms for Mining Association Rules." Proceedings of the 20th International Conference on Very Large Data Bases. [Online], 1994.

[3] C. Ahmed, S. Tanbeer, B.-S. Jeong and Y.-K. Lee. "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases." IEEE Transactions on Knowledge and Data Engineering. [Online], 2009.

[4] C.-J. Chu, V. Tseng, and T. Liang. "An efficient algorithm for mining temporal high utility itemsets from data streams." Journal of Systems and Software. [Online], 2008.

[5] P. Fournier-Viger, C.-W. Wu, S. Zida, and V. Tseng. "FHM: Faster High-Utility Item set Mining Using Estimated Utility Cooccurrence Pruning." Proceedings of 21st International Symposium ISMIS. [Online], 2014.

[6] M. Liu and J. Qu. "Mining high utility item sets without candidate generation." Proceedings of the 21st ACM international conference on Information and Knowledge management. [Online], 2012.

[7] Y. Liu, W. Liao, and A. Choudhary. "A Two-Phase Algorithm for Fast Discovery of High Utility Item sets." Proceedings of 9th Pacific-Asia Conference PAKDD. [Online], 2005.

[8] X. Liu, J. Guan, and P. Hu. "Mining frequently closed item sets from a landmark window over online data streams." Computer & Mathematics with Applications. [Online], 2009.

[9] V. Tseng, B.-E. Shie, C.-W. Wu, and P. Yu. "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases." IEEE Transactions on Knowledge and Data Engineering. [Online], 2013.

[10] Y. Takama and S. Hattori. "Mining Association Rules for Adaptive Search Engine Based on RDF Technology." IEEE Transactions on Industrial Electronics. [Online], 2007.

[11] C. Ahmed, S. Tanbeer, B.-S. Jeong and Y.-K. Lee. "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases" IEEE Transactions on Knowledge and Data Engineering. [Online], 2009.

[12] C. Ahmed, S. Tanbeer, B.-S. Jeong, H.-J. Choi. "Interactive mining of high utility patterns over data streams. Expert Systems with Application [Online]. 2012.