

Mining Frequent Itemsets Without Candidate Generation In Machine Learning

¹B. Satheesh, ²Ajay P., ³Alberto Clavería Navarrete, ⁴Dario E. Soto Duran, ⁵Gerber F. Incacari Sancho

¹AP/ Dept. of IT, Mailam Engineering College, Tamilnadu, India.

²Research Scholar, Anna University, Department of Electronics and Communications Karpagam College of engineering.

³Universidad de Sevilla, España

⁴Facultad de ingeniería, Tecnológico de Antioquia I.U

⁵Universidad Nacional del Callao, Lima, Perú

¹satheeshbssb@gmail.com, ²ajaynair707@gmail.com, ³claveria.alberto@gmail.com, ⁴dsoto@tdea.edu.co,

⁵gfincacaris@unac.edu.pe

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Mining of regular trends in group action databases, time series databases, and lots of different database types was popularly studied in data processing research. Most previous studies follow the generation-and-test method of associate degree Apriori-like candidate collection. In this study, we seem to propose a particular frequency tree like structure, which is associated degree of prefix-tree like structure that is extended to be used for compressed storage, crucial knowledge of the frequency pattern, associated degrees create an economic FP-tree mining methodology, FP growth, by the growth of pattern fragments for the mining of the entire set of frequent patterns. Three different mining techniques are used to outsize the information which is compressed into small structures such as FP-tree that avoids repetitive information scans, cost. The proposed FP-tree-based mining receives an example philosophy of section creation to stay away from the exorbitant age of several competitor sets, and an apportioning-based, separating and-overcoming technique is used to divide the mining task into a contingent knowledge base for restricted mining designs that effectively reduces the investigation field.

Keywords: Itemsets, FP-Tree, transactions, Conditional FP-Growth.

1. Introduction

Data mining may be a way of getting beneficial, previously unknown, and eventually understandable knowledge from the details. Association rules mining is one in every critical piece of information on information knowledge} mining and is used to look for interesting associations or connection relationships in mass data between item sets [1]. The discovery of frequent item sets could be a key technology and step within the application of the mining rules of the association. The primary illustrious rule is that Apriori implies within the algorithms of the discovery of frequent item sets by Agrawal. Apriori rule scans the details extracting solitary item sets by continuous association to search for all the frequent item sets in the information. However, the Apriori rule repeatedly scans the information in the mining system and generates an oversized variety of candidate itemsets affecting the mining running pace [2].

The FP-Growth (frequent-pattern growth) rule is an improved rule suggested by the Jiawei dynasty and then forward by the Apriori rule. It compresses information sets to an FP-tree, doubly scans the data, doesn't turn out the candidate, item sets in the mining process, and greatly enhances mining capacity. The FP-Growth rule must, however, generate an FP-tree containing all the information sets. The memory house is in high demand for this FP-tree. And scanning the information doubly together does not make the power of the FP-Growth rule strong. Then the compressed information is divided into a series of databases on condition (a special kind of prediction database) [3-5].

2. Working of FP-Growth Algorithm

Without generating candidate itemset, the FP-Growth the algorithm permits the frequent itemset to be found. The method mentioned below is a two-step one,

1) The primary step is to scan the information inside the information to look for the occurrences of the element sets. This process is the same as the beginning of Apriori. Inside the information the count of 1-itemsets is called 1-itemset help count or frequency [6].

2) Constructing the FP the tree is the second step. For that, produce the tree's base. The base is drawn by null [7].

3) The subsequent move is to search the data once again and review the transactions. Examine and assess the entity set in the primary transactions. The itemset with the grievous count of bodily harm is taken at the highest, consecutive itemset with a then lower count. It implies that the tree branch is constructed in a down order of count with dealings itemsets [8].

4) Consecutive transactions inside the information shall be reviewed. In order of count, the itemsets square measure ordered in down order. If any itemset of these deals is already a gift in another branch (for example, in the first transaction), then a traditional prefix to the foundation will be exchanged by this dealings branch. This implies that during these dealings, the popular itemset is joined to the new node of another itemset.

5) The itemset count is therefore increased since it occurs within the transactions. As they are generated and joined by transactions, each common node and new node count is inflated by one.

6) The consequent move is to mine the FP Tree which was formed. For this, the ties of very cheap nodes are examined 1st along with hand. The quite cheap node represents the duration one of the frequency patterns. Traverse the trail within the FP Tree from here. Conditional pattern base may be a sub database consisting of prefix ways with very cheap node (suffix) occurring within the FP tree [9].

7) Construct a Conditional FP Tree, which is formed by the number of items in the path. Inside the Conditional FP Tree, the itemsets meeting the brink help square measure thought-about [10].

8) The square measure developed from the Conditional FP Tree for Frequent Patterns [11].

Consider transaction data environment, as shown in Fig. 1 Includes 5 entries with a unique TID (Transaction ID) each.

Table 1: Input Dataset

T_ID	Itemset
T100	BS,PS,SM,SS,BB,NB
T200	LS,PS,SM,SS,BB,NB
T300	BS,KS,SS,BB
T400	BS,ME,MR,SS,NB
T500	MR,PS,PS,SS,RM,BB

A speculative dataset of exchanges with each letter speaking to an item is the information given previously. Every thing's recurrence is determined.

Table 2: frequency of item

T_ID	Itemset
KS	1
MR	2
LS	1
BB	4
RM	1
SS	5
BS	3

SM	2
PS	3
ME	1
NB	3

Leave the base support alone 3. An assortment of Regular Patterns is built containing all components with a recurrence more prominent than or equivalent to the base help. In plummeting request of their separate frequencies, these components are prepared. The set L seems as though this in the wake of adding the fitting things: $L = \{SS : 5, E : 4, BS : 3, PS : 3, NB : 3\}$

Presently the separate Ordered-Item assortment is intended for every exchange. It is accomplished by repeating the Frequent Pattern set and testing if the exchange is being referred to contain the current item. In the event that the current thing is contained, the thing is added for the current exchange in the Ordered-Item climate. For all exchanges, the accompanying table is built:

Table 3: Creating ordered-item set

Transaction ID	Items	Ordered-Item Set
T100	BS,PS,SM,SS,BB,NB	{SS,BB,BS,PS,NB}
T200	LS,PS,SM,SS,BB,NB	{SS,BB,PS,NB}
T300	BS,KS,SS,BB	{SS,BB,BS}
T400	BS,ME,MR,SS,NB	{SS,BS,NB}
T500	MR,PS,PS,SS,RM,BB	{SS,BB,PS}

Now, in a Tree Data Structure, all the Ordered-Item sets are added.

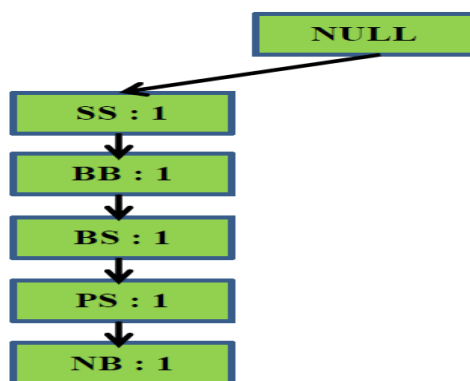


Fig A: inserting the set {SS, BB, BS, PS, NB}

Fig: A indicates the mapped TID 1 for the path. {BS, PS, SM, SS, BB, NB} Read transaction 1: Based on support count = 3, {SS, BB, BS, PS, NB}, build 5 nodes and the path numbers NULL->SS->BB->BS->PS->NB and Set 1.

The support count is simply increased by 1.

Before the SS and BB components are added. We can see that there is no immediate connection among BB and PS while adding PS, so another hub for the thing, PS is instated with a help tally of 1, and thing BB is appended to this new hub.

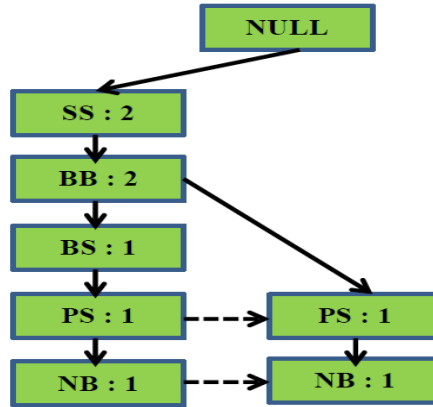


Fig B: Insert the {SS, BB, PS, NB} set

The TID 2 mapped to a path is shown by b. {LS, BB, SS, SM, PS, NB} Read transaction 2: Based on support count = 3, {SS, BB, PS, NB}, cross the current FP tree and create 2 new nodes. And the path NULL —>SS->BB->PS->NB and K&E = 2, and O & Y = 1.

Here, each element's support the count is simply increased by 1.

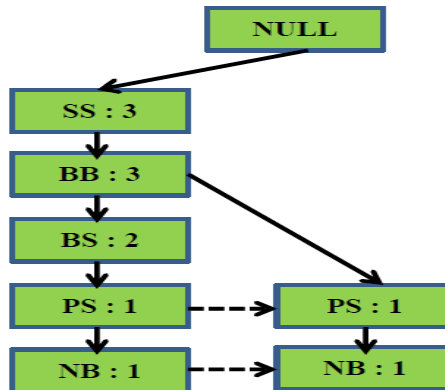


Fig C: Inserting the set {SS, BB, BS}

Fig. C displays the plotted TID 3 of the route. Transaction 3 read: {KS, BB, SS, BS} Based on support count = 3, {SS, BB, BS}, traverse the current FP tree and the NULL path->SS->BB->BS and Set counts SS&BB = 3, and BS = 2 Similar to stage b), the help tally of SS is first expanded, at that point new hubs are instated for BS and NB and associated as needs be.

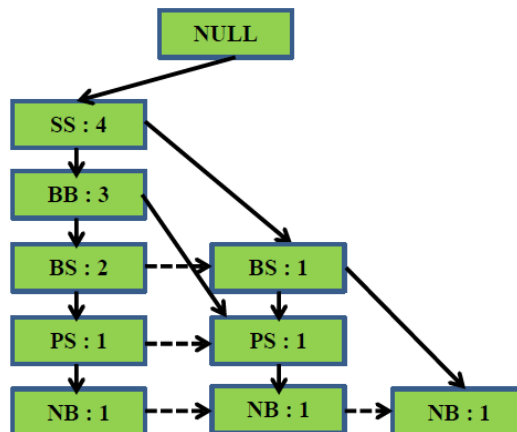


Fig D: Inserting the set {SS, BS, NB}

Fig D: indicates the route mapped to the TID 4. {MR, SS, BS, ME, NB} Read transaction 4: Cross the current FP tree with support count = 3, {SS, BS, NB} and the NULL path->SS->BS->NB and Set counts SS = 4, and BS = 1, NB=1.

Here, the help support counts of the comparing components are essentially expanded. Notice that the quantity of supports for the new thing hub PS is expanded.

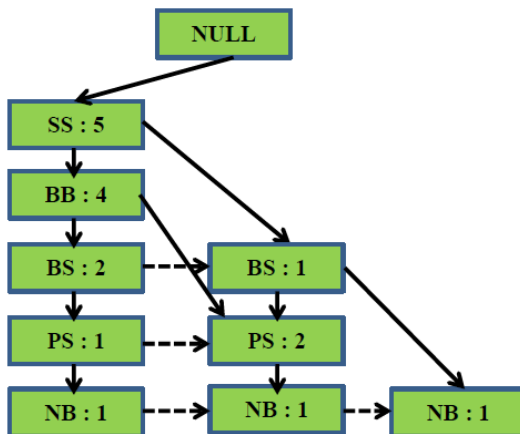


Fig E: Inserting the set {SS, BB, PS}

Fig E: indicates the path mapped to the TID 5. Transaction 4 read: {MR, BB, RM, SS, PS, PS}. Traverse the current FP tree by counting support = 3, {SS, BB, PS} and the path NULL->SS->BB->PS and Set counts SS = 5 and BB = 4, PS= 2 and Set counts SS = 5.

Now, the Conditional Pattern Base is computed for each item, which is the path labels of all paths in the frequent-pattern tree leading to any node of the given item. Note that the elements in the table below are grouped with their frequencies in ascending order.

Item	Conditional Pattern Base
NB	{{SS, BB, BS, PS : 1}, {SS, BB, PS : 1}, {SS, BS : 1}}
PS	{{SS, BB, BS : 1}, {SS, BB : 2}}
BS	{{SS, BB : 2}, {SS : 1}}
BB	{{SS, BB : 2}, {SS : 1}}
SS	

The Conditional Pattern Base is currently processed for everything, which are the way names of the apparent multitude of ways in the incessant example tree prompting any hub of the given thing. Note that the things are gathered in the climbing request of the frequencies in the table underneath.

Items	Conditional Pattern Base	Conditional Frequent Pattern Tree
NB	{{SS, BB, BS, PS : 1}, {SS, BB, PS : 1}, {SS, BS : 1}}	{SS : 3}
PS	{{SS, BB, BS : 1}}	{SS, BB : 3}

	1}, { SS,BB : 2}}	
BS	{{SS,BB : 2}, {SS : 1}}	{SS : 3}
BB	{{SS,BB:2},{SS : 1}}	{SS : 4}
SS		

The principles of the Frequent Patterns are produced from the Tree of Conditional Frequent Patterns by matching the things of the Conditional Frequent Pattern Tree set with the thing as appeared in the table underneath.

Item	Frequent Pattern Generated
NB	{SS ,NB : 3}
PS	{SS ,PS : 3},{BB, PS : 3}, {BB ,SS,PS : 3}
BS	{SS ,BS: 3}
BB	{BB,SS : 3}
SS	

Two sorts of associationrules can be derived for each line, for example for the principal line containing the variable, the guidelines SS->NB and NB-> SS. The certainty of the two laws is determined to assess the genuine law and to protect the certainty of the one with certainty that is more noteworthy than or equivalent to the base certainty esteem.

For each line, two kinds of affiliation rules can be surmised; for instance, the guidelines SS->NB and NB->SS can be induced for the primary line that contains the component. The certainty of the two guidelines is estimated to build up the genuine law and the certainty of one with certainty more prominent than or equivalent to the base certainty esteem is held.

3. Advantages of FP Growth Algorithm

1. Compared to Apriori, which scans transactions for each iteration, this algorithm has to search the database twice only.
2. This algorithm has the pairing of things isn't done and that makes it quicker.
3. The database is stored in memory in a portable edition.
4. It is powerful and scalable for regular patterns, both long and short, to mining.

4. Disadvantages of FP-Growth Algorithm

1. FP Tree is more cumbersome than Apriori and hard to build.
2. Maybe it's costly.
3. The algorithm cannot fit in the shared memory, if the database is huge.

5. Result and Discussion

It can additionally be improved by supplanting the electronic edge with an optical limit to keep up the spatial optical parallelism and to dodge optoelectronic between changes. Since optical neural organizations are not an extremely famous strategy, the expense and the accessibility of assets should be managed.

The plan is adaptable as it tends to be utilized for different applications. Be that as it may, the maximal number of interconnections is a requirement. The frameworks with around 70,000 interconnections have been

effectively executed which is empowering for planning such models. Some information pressure methods can additionally be received to diminish the size of the weight veils to adapt up to the restriction of interconnections. The model can be upgraded utilizing the maximal itemset approach, appropriating the information base, and applying other proper strategies to it.

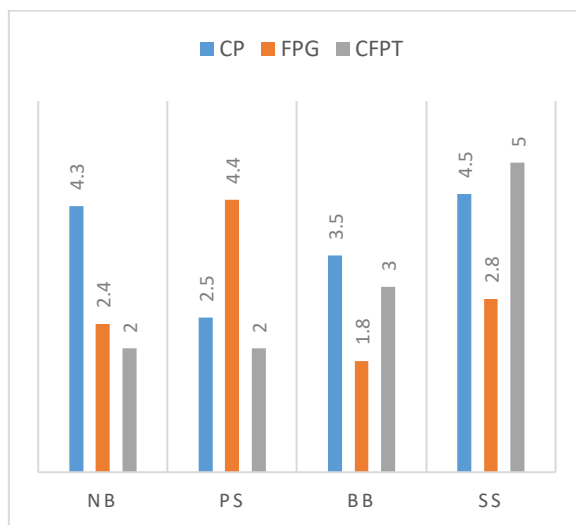


Fig.1 Comparison of FP growth Algorithm

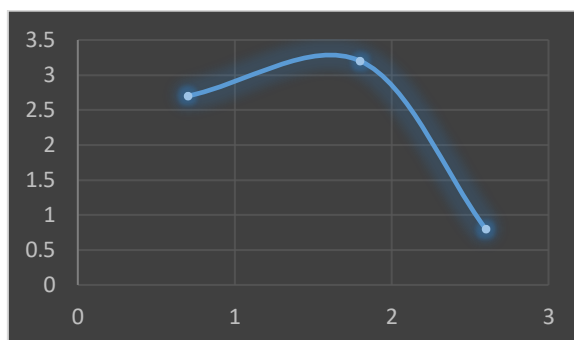


Fig.2 Conditional Frequent Pattern Tree

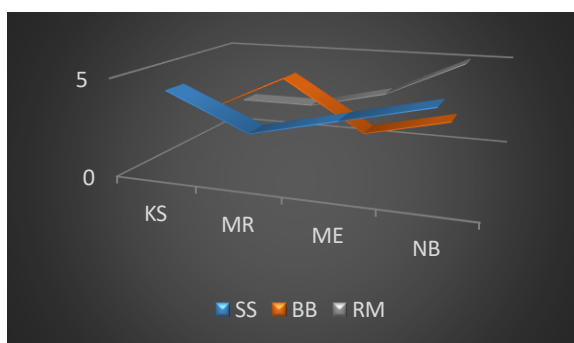


Fig.3 Comparison of Inserting the set

6. Conclusion

The algorithmic software Apriori is used for the rules on mining associations. It functions on the theory, "it can also, be frequent non-empty subsets of frequent itemsets." It shapes candidates for k-itemset from (k-1) object sets and scans the information to look for frequent itemsets.

The algorithmic software of frequent pattern growth is the technique of identifying frequent patterns while not producing candidates. Instead of victimization, it creates Associate an FP Tree to create and take a look at

Apriori's strategy. The FP Growth algorithmic program's main emphasis is on fragmenting stuff methods, and regular patterns of mining.

7. Acknowledgement

Our sincere gratitude to Dr. R. Mariappan , Professor & Head in Master of Computer Applications and Dr. M. Ramalingam, Professor & Head in Department of Information Technology for her valuable support and guidance.

References

1. Jia-Dong Ren, Hui-Ling He, Chang-Zhen Hu, Li-Na Xu, Li-Bo Wang Mining Frequent Pattern Based On Fading Factor In Data Streams, Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 July 2009.
2. http://120.105.184.250/lwcheng/data_mining/fp-growth/FPGrowth.pdf
3. Meera Narvekar, Shafaque Fatma Syed “An optimized algorithm for association mining using FP tree” International conference on advanced computing technologies and application (ICACTA-2015).
4. V. Ramya, M. Ramakrishnan “Mining Association Rules Using Modified FP-Growth Algorithm” international journal for research in emerging science and technology, E-ISSN: 2349-7610.
5. Rakesh Agrawal Ramakrishnan Shrikant, “Fast Algorithms for Mining Association Rules”, IBMAlmaden Research Center.
6. Chen Wenwei. Data warehouse and data mining tutorial [M]. Beijing: Tsinghua University Press. 2006.
7. Hand David, MannilaHeikki,SmythPadhraic. Principles of Data Mining[M].Beijing:China Machine Press,2002.
8. P. Saravanan (2019),” Improved Joint Selective Encryption and Matrix Embedding Technique Using Adaptive Block Size Coding in HEVC Standard”, Journal of Engineering and AppliedScience(JEAS) Vol: 14 Iss:11,Pp:3690-3697.
9. P. Saravanan (2018),” ComparativeStudy on different Video HidingTechniques”, International Journal of Computer Sciences and Engineering (IJCSE), Vol.6 (11), Nov 2018, E-ISSN: 2347-2693 Pp.497-502.
10. M. Ramalingam and R.M.S. Parvathi, “Policy-Based Semantic Access Control Framework for Fine-Grained Access in Semantic Web Services”, European Journal of Scientific Research, Vol.74, No.1, pp. 154-163, 2012.
11. Vinoth Kumar V, Ramamoorthy S, Dhilip Kumar V, Prabu M, Balajee J.M., “Design and Evaluation of Wi-Fi Offloading Mechanism in Heterogeneous Network”,
12. International Journal of e-Collaboration (IJeC), IGI Global, 17(1), 2020.