iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.53799

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Comparative Analysis of Algorithms for Mining Frequent Patterns

A. Selva Priya
PG Scholar, Department of Computer Science and Engineering, Government College of Technology, Coimbatore, India

Abstract: ***In the computerized world, everything is moving online, and data comes in different shapes and sizes and is collected in different ways. By using data mining, frequent pattern in the databases can be identified, and it can be used in numerous applications. Finding frequent patterns in huge databases is important because it reveals important information that cannot be found through simple data surfing. To find common patterns, a variety of methods are utilized, each of which performs differently. Apriori and FP Growth are the fundamental algorithms employed in frequent pattern mining. The functioning and experimental results of various algorithms are compared in this study, and their benefits and drawbacks are discussed.***
Keywords: Data mining, Frequent pattern mining, Apriori, FP-growth.

I. INTRODUCTION

Data mining has long been the subject of database research. The continually declining cost and growing compactness of storage devices have made it possible to store every transaction in a transactional database [2]. With this storage, individuals may access the data whenever they want, and they can also use it to discover relationships between different data points. Agarwal et al. [1] were the first to present the issue of determining relationships between various data components. The resolution to this issue can increase revenue and optimize storage. The researcher defines transactional databases, database design, frequent patterns, frequent item set, and candidate item set in this part. A database is a collection of material that has been organized systematically so that it may be conveniently retrieved and modified in the future. Although there are many various types of databases, including active databases, cloud databases, embedded databases, and transactional databases, the researcher solely addresses transactional databases in this work. A database that doesn't use auto commit is referred to be a transactional database. Transactional databases are the most common type of relational database today [4]. A database layout explains the representation of data. The horizontal layout and the vertical layout are the two layouts that are most frequently used. Two columns make up the horizontal arrangement. The first provides the transaction identifier and the second the things that were purchased during that transaction. In a vertical style, the first column displays the item ID, and the second displays the transaction ID for the transaction in which the specific item was purchased. A third layout, usually referred to be a predicted layout, exists. There is no actual layout here. In this configuration, the system just keeps track of the transaction identification and related item. It is a divide and conquers mechanism that recursively shrinks the size of the database by only taking into accounts the longest pattern. A pattern that appears in significantly more transactions is said to be common. A frequent item set is one whose support exceeds a minimum support determined by the user. The provided study is divided into seven sections: the introduction, a brief overview of the three common pattern mining algorithms— Apriori, Eclat, and FP Growth—in the second section. The experiment is described in the third section. The dataset is described in the fourth section. The algorithms utilized under various circumstances are compared in the fifth section. The conclusion is presented in the sixth part, and references are included there as well [33].

II.LITERATURE SURVEY

The performance of three frequent pattern mining algorithms is contrasted in the literature review in order to determine which one that performs the best.

A. Apriori

Agrawal and Srikant initially put forth the Apriori algorithm in 1994.The Apriori property, which asserts that "any sub (k-1)-Item set of frequent k-Item set must be frequent," is the foundation for this technique [5]. The apriori algorithm consists of two main processes: the first is candidate generation, which determines the support count of the corresponding sensor items by scanning transactional databases, and the second is large item set generation, which is produced by removing candidates from the candidate set whose support counts fall below a certain threshold. These processes are iteratively performed until candidate Item sets or big item sets are empty [29].

The candidate set, which consists of one sensor item, is once again scanned for in the initial database, and the support for those items is then tallied. The items with item counts below the user-specified threshold (in the example above, the threshold is 30%) are then simply removed from these 1-Itemset candidates. The database is once again scanned in the second pass to provide candidates for 2-Itemsets that each include two items. These candidates are then once more trimmed to make large 2-item sets using the apriori property. The apriori property states that every sub-item set of two frequent item sets must be frequent.The process is complete when the large item set is empty and the item set candidates are eliminated in the database's fourth scan.

This algorithm has two drawbacks:

1) The first is its complicated candidate item set formation procedure, which uses a lot of memory and takes a long time to execute;
2) The second is its excessive reliance on database scans for candidate generation.

There are typically two ways to get over these restrictions:

a) One is to experiment with various pruning and filtering procedures to reduce the size of candidate Item set.
b) The second strategy either replaces the original database with a subset of transactions based on a significant number of frequently occurring Item sets or reduces the frequency of database scans [17, 29].

B. Steps in Apriori Algorithm

There are 3 steps to mine the frequent patterns:

1) *Generate And Test:* By searching the database, first identify the 1-itemset frequent elements L1, and then eliminate all the elements from C that don't meet the minimal support requirements.
2) *Join Step:* Lk-1*Lk-1, also known as the Cartesian product of Lk-1, is used by Ck to join the prior frequent elements in order to reach the next level elements; in other words, this step creates new candidate k-itemsets based on joining Lk-1 with itself, which was discovered in the previous iteration. Let Lk be the common k-item set and Ck stand for candidate k-itemsets.
3) *Prune Test:* Pruning eliminates some of the candidate k item set using the Apriori's principle. A scan of the database is used to determine the count of each candidate in Ck would result in the determination of Lk (i.e. all the candidates having a count less than the minimum support count). Step 2 and 3 is repeated until no new candidate set is generated [22].

C. FP-Growth

The FP Growth algorithm is the most widely used algorithm for pattern finding in the field of data mining. A novel, compressed data structure called the FP tree a prefix- tree structure that stores quantifiable data about frequent patterns is created to address the two fundamental shortcomings of the Apriori method in [6]. A frequent pattern growth algorithm was created based on the FP tree [37].

It involves a two-step process.

1) A frequent pattern tree is built in the first stage by twice scanning the database. The first run of the database scans the data, calculates the support count for each item, and eliminates the list of uncommon patterns sorts the remaining patterns in descending order.
2) FP Tree is built during the database's second phase. Frequent patterns are extracted from the FP Tree in the second stage using the FP growth method.

Based on node link property and prefix path property, conditional FP tree base and conditional FP tree are constructed. Each element in the head table has a conditional pattern base. Construction of a conditional tree. For the frequently occurring items of the pattern base, a conditional FP tree is built [39]. Frequent patterns must be extracted after the Conditional FP tree has been created.

Three key goals can be accomplished with the help of the FP growth algorithm,

a) The first of which is that the computing cost is significantly reduced and the database is only scanned twice [35].
b) The generation of any candidate item sets is not the second key goal [35].
c) The third goal is to limit the search space by using a divide and conquer strategy.

On the other hand, there is one problem with the FP growth algorithm. Once new transactions are added to the database, the FP tree must be updated, and the entire process must be repeated, it is challenging to employ in incremental mining.

- *FP-Tree:* Han came up with the FP-Tree [24]. The FP-Tree represents all of the important frequent information from a data source due to its compact design. With nodes in the path arranged in decreasing order of frequency, each FP-Tree path represents a set of frequently recurring items. All overlapping item sets share the same prefix path, which is the main advantage of the FP-Tree. As a result, the information in the data set is significantly compressed [9].
- *Pass1:* It searches the data initially before locating evidence for each item. The frequent item sets are then sorted in decreasing order according to their support after the infrequent item sets are discarded.
- *Pass 2:* Nodes correspond to item set and have a counter.
- It first reads 1 transaction at a time and maps it to a path.
- When transactions share items (when they have the same prefix), pathways can overlap because in this scenario, counters are incremented.
- A linked list is created by maintaining pointers between nodes that contain the same item (dotted lines). The more the pathways overlap, the higher the compression will be. In the memory, FP-Tree might fit.
- Frequent item sets are extracted from the FP-Tree [22].

III. EXPERIMENT

The two algorithms discussed above were built in Python, and their results on the Groceries dataset were compared by altering the number of characteristics and instances. Execution time is the performance-comparing factor [25].

IV. DATASET DESCRIPTION

Based on how much execution time each algorithm used to compile a list of every valid association rule, they were evaluated. Table 1 contains all of the data from the grocery dataset. The findings are displayed in Fig. 1. The graph's longitudinal axis displays time, while its latitudinal axis displays several algorithms [30]. The Apriori and FP- growth algorithms execution times are depicted on the charts in Fig. 1. The size of the created FP-tree will be significantly less due to the overlapping of frequent items because the Grocery dataset and it contain a lot of frequent things. Hence, the experimental findings demonstrated the proposed FP- growth algorithm's advantage in terms of execution time consumption. The FP-growth offers the fastest execution. The most expensive algorithms, however, are Apriori and other algorithms.

Table 1: Dataset Description

Datasets Name	Number of Instances	Number of Attributes
Groceries dataset	38766	03

V. PERFORMANCE ANALYSIS

Another development in association rule mining and frequent pattern mining is the FP Growth Algorithm, which addresses the two shortcomings of the Apriori Algorithm [3, 38]. Three essential qualities are necessary for FP-Growth to function effectively: (1) A divide-and-conquer approach is used to condense the search space and extract minor patterns from the mining issue in conditional databases into a number of smaller challenges. (2) The FP-Growth algorithm avoids the challenging process of constructing the candidate item set for a large number of candidate item sets, and (3) the database is compressed into a highly condensed, much smaller data structure known as the FP tree to avoid expensive and repetitive database scans [11, 36].

The performance assessment of three frequent item set mining techniques is demonstrated in this experiment. Based on how much memory each algorithm used to compile a list of every valid association rule, they were compared [21]. The results obtained are shown in Fig. 1. The graph longitudinal axis shows the memory in time in seconds, and the latitudinal axis shows the different algorithms. The charts portrayed in Fig. 1 show the execution time of the Apriori and FP-growth algorithms [30].Figure 2 and Figure 3 shows the lift vs confidence obtained by using Apriori and FP-Growth algorithms. Apriori TID, which has a slightly longer execution time than Apriori, has been shown to be the most expensive one in this category. FP-growth outperformed the earlier algorithms and was able to exhibit moderate behaviour. The best performance among all the other algorithms has once again been demonstrated by FP-Growth [23].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

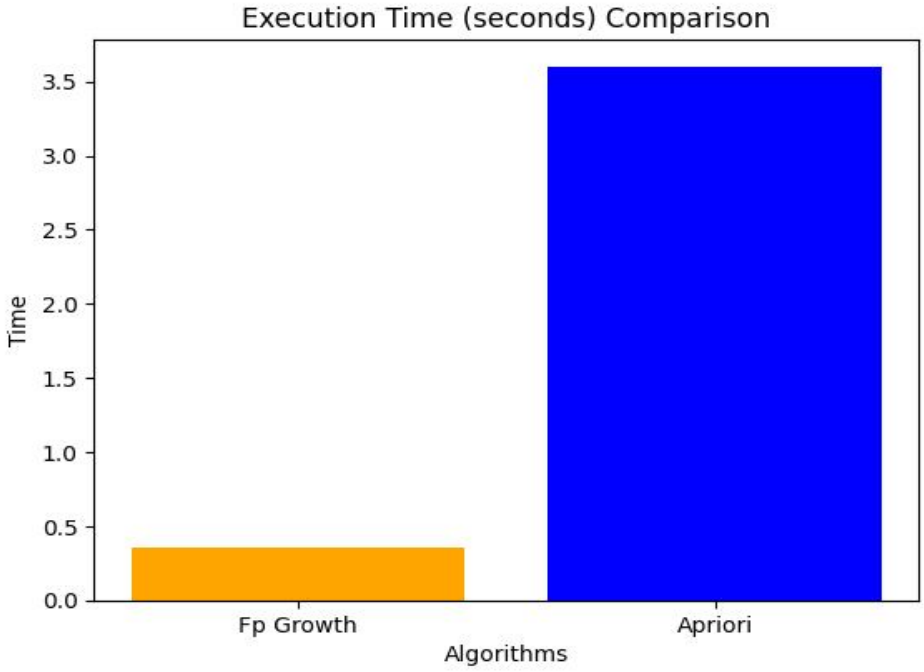


Figure 1: Execution time Comparison of Apriori and FP Growth algorithm

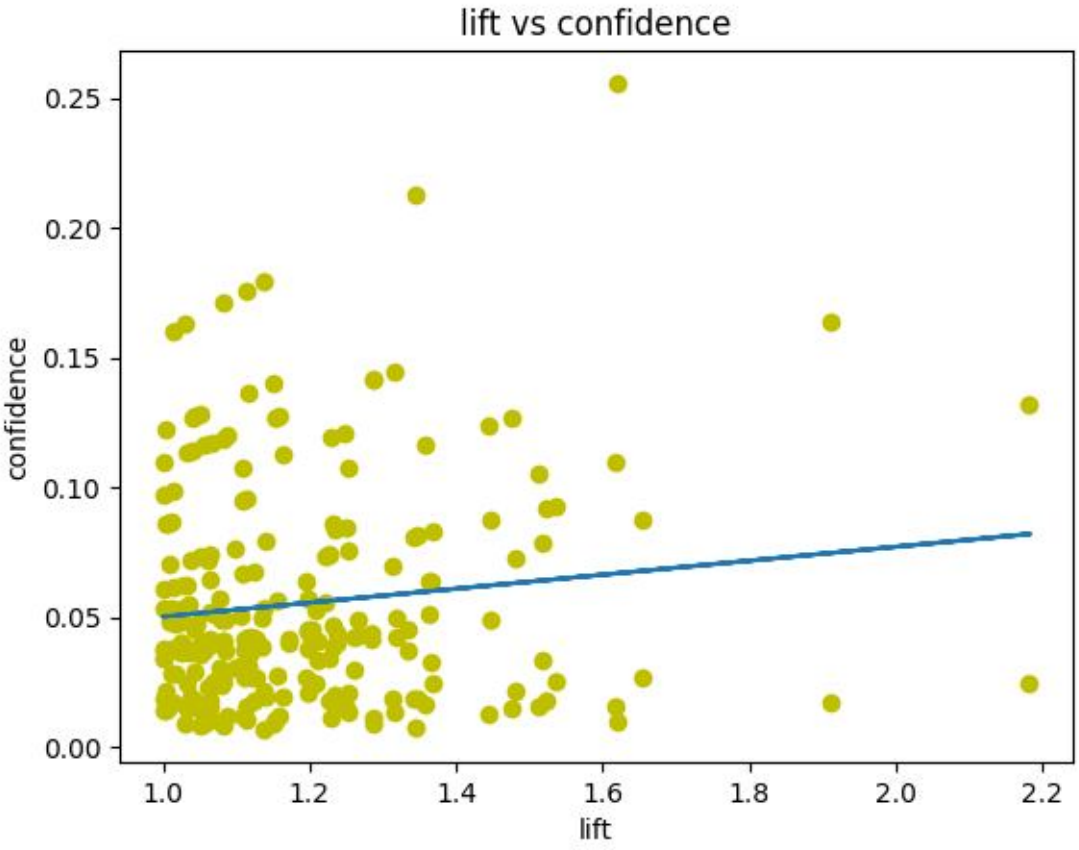


Figure 2: Lift vs confidence (Apriori Algorithm)

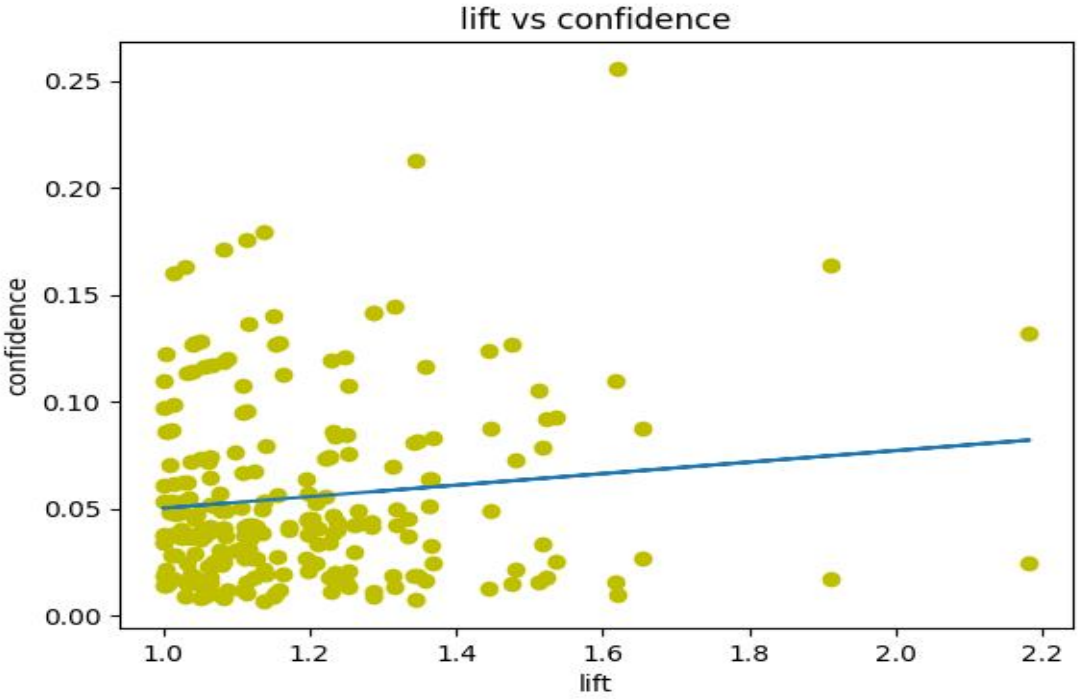


Figure 3: Lift vs confidence (FP-Growth Algorithm)

Table 2: Comparative analysis of mining algorithms

	APRIORI	FP-Growth
Objective	Data mining methodology called Apriori identifies common patterns and association rules in huge transactional databases.	A technique for locating common patterns without candidate creation is proposed by FP-growth.
Techniques	Breath first search & Apriori property.	Divide and conquer
Database scan	A candidate item set is generated after scanning the database each time.	The database is only scanned twice.
Time	Execution takes a long time since the database must be scanned for each candidate item set generation.	Less time as compared to APriori algorithm.
Drawback	There are too many candidate items. There are too many database passes need a lot of RAM.	FP-Tree requires more memory and is expensive to construct
Advantage	1.Easy to use and comprehend algorithm 2.Multiple scans	1. Faster than Apriori. 2.No creation of candidates
Data	Horizontal	Horizontal
Format storage structure	Array	Tree (FP-Tree)

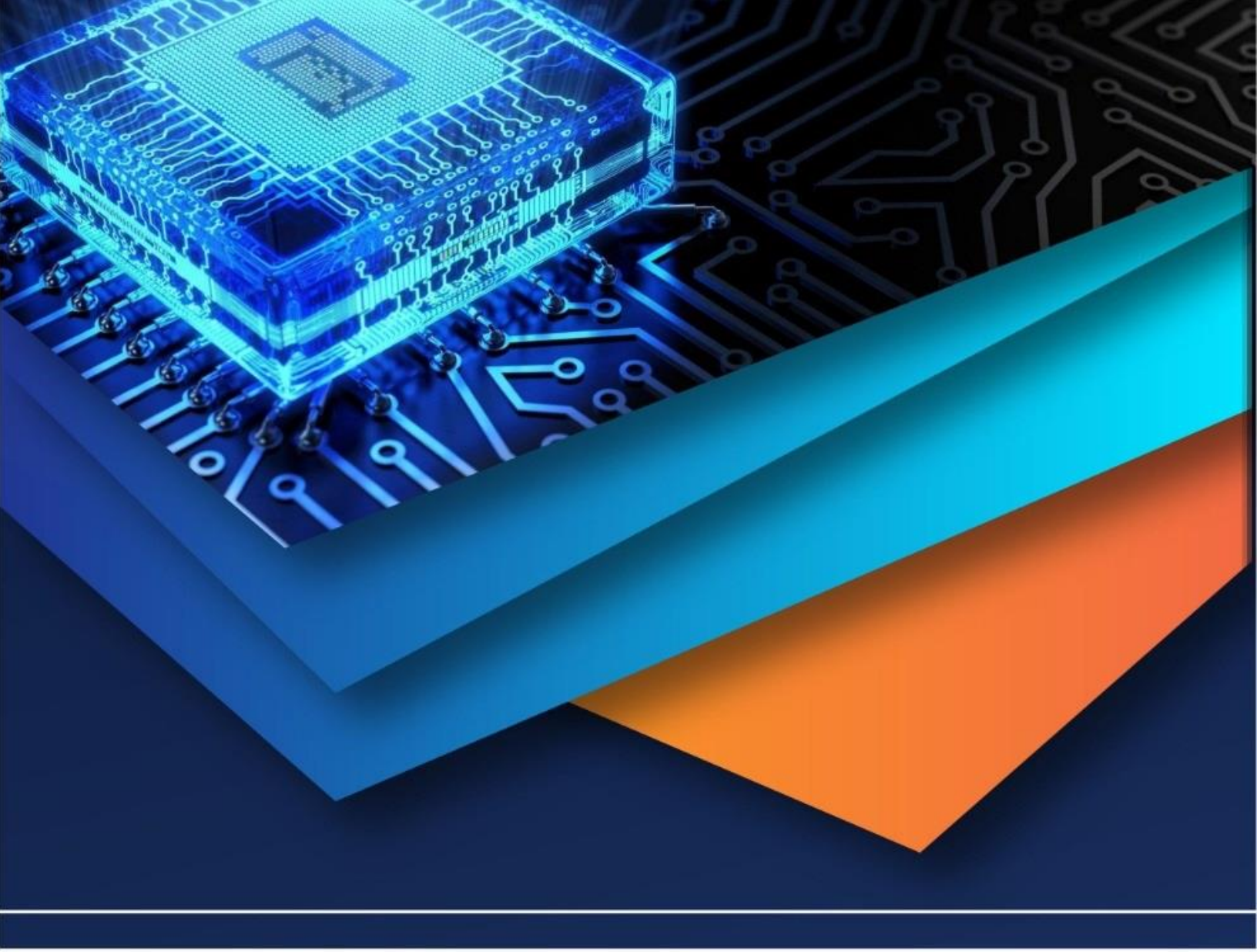
VI. CONCLUSION

Apriori and FP Growth are two frequent pattern mining algorithms that are tested and examined in this study. The comparison was done using identical database transactions in order to better understand these techniques. The key difficulties in the context of frequent pattern mining were high numbers of database scans, lengthy execution times, and memory requirements for a large transactional database. Because it takes less time to execute and uses less memory than other frequent pattern mining algorithms, FP-growth outperformed them, according to this study.

REFERENCES

[1] Agarwal, R.C., Agarwal, C.C. and Prasad, V.V.V. (2001) A tree projection algorithm for generation of frequent item sets. Journal of Parallel and Distributed Computing, 61(3), Pp. 350–371.
[2] Bhadoria et. al. Analysis of Frequent Itemset Mining on Variant Datasets published in int.J.comp. Tech.appl., vol(2)5, ISSN:2229-6093, Pp. 1328-1333.
[3] Jiawei Han · Hong Cheng · Dong Xin · Xifeng Yan, "Frequent pattern mining: current status and future Directions," Data Mining Knowl Discov, vol. 15, no. I, p. 32, 2007.
[4] http://en.wikipedia.org/wiki/Database_transaction [on 11th Nov 2012].
[5] Sourav S. Bhowmick Qiankun Zhao, "Association Rule Mining: A Survey," Nanyang Technological University,Singapore, 2003.
[6] Jian Pei, Jiawei Han, "Mining Frequent patterns without candidate generation," in SIGMOD '00 Proceedings of the 2000 ACM SIGMOD international conference on Management of data, New York, NY, USA, 2000, pp. 1-12.
[7] Panchal Mayur, Ladumor Dhara, Kapadiya Jahnvi, Desai Piyusha, Patel Tushar S., "An Analytical Study of Various Frequent Itemset Mining Algorithms," Research Journal of Computer and Information Technology Sciences, p. 4, 2013.
[8] Imielienskin T. and Swami A. Agrawal R., "Mining Association Rules Between set of items in largedatabases," in Management of Data, 1993,p. 9.
[9] Patil, Manoj, and Tejashri Patil. "Apriori Algorithm against Fp Growth Algorithm: A Comparative Study of Data Mining Algorithms." Available at SSRN 4113695 (2022).
[10] Agrawal R, Srikant R (1994) Fast algorithms for mining association rules. In: Proc. 20th int. conf. very large databases, VLDB, vol. 1215, pp. 487–499.
[11] Borgelt C., "Efficient Implementations of Apriori and Eclat," in 1st IEEE ICDM Workshop on Frequent Item Set, 2003, p. 9
[12] Goethals, Bart. "Survey on frequent pattern mining." Univ. of Helsinki 19 (2003): 840-852.
[13] Mohammed J. Zaki, "Scalable Algorithms for Association Mining," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, pp. 372-390, 2002.
[14] Agrawal R, Mannila H, Srikanth R, Toivonen H, Verkamo AI (1996) Fast discovery of association rules. In: Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (Eds.) Advances in knowledge discovery and data mining, pp. 307– 328.
[15] Ke-Chung L, Liao IE, Sheng C (2011) An improved frequent pattern growth method for mining association rules. Expert Syst Appl 38(5):5154.
[16] Panchal Mayur, Ladumor Dhara, Kapadiya Jahnvi, Desai Piyusha, Patel Tushar S., "An Analytical Study of Various Frequent Itemset Mining Algorithms," Research Journal of Computer and Information Technology Sciences, p. 4, 2013.
[17] Chistopher.T, PhD Saravanan Suba, "A Study on Milestones of Association Rule Mining," International Journal of Computer Applications, p. 7, June 2012.
[18] Yabing J (2013) Research of an improved apriori algorithm in data mining association rules. Int J Comput Commun Eng 2(1):25
[19] Srinivasan Parthasarathy, and Wei Li Mohammed Javeed Zaki, "A Localized Algorithm for Parallel Association Mining," in In 9th ACM Symp. Parallel Algorithms & Architectures. , 1997.
[20] Zaki MJ (1997) Fast mining of sequential patterns in very large databases. University of Rochester Computer Science Department, New York.
[21] Borah A, Nath B (2021) Comparative evaluation of pattern mining techniques: an empirical study. Complex Intell. Syst. 7:589–619.
[22] Kavitha, M., and S. T. Selvi. "Comparative study on Apriori algorithm and Fp growth algorithm with pros and cons." International Journal of Computer Science Trends and Technology (I JCS T)–Volume 4 (2016).
[23] Hasan, Md Mahamud, and Sadia Zaman Mishu. "An adaptive method for mining frequent itemsets based on apriori and FP growth algorithm." 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2). IEEE, 2018.
[24] Anurag Choubey, Ravindra Patel,J.L.Rana, "A Survey Of Efficient Algorithms And New Approach For Fast Discovery Of Frequent Item Set For Association Rule Mining", International Journal of Soft Computing and Engineering,2011.
[25] Garg, Kanwal, and Deepak Kumar. "Comparing the performance of frequent pattern mining algorithms." International Journal of Computer Applications 69.25 (2013).
[26] Goswami D.N et. al. "An Algorithm for Frequent Pattern Mining Based On Apriori" (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, Pp. 942-947.
[27] SathishKumar et al. "Efficient Tree Based Distributed Data Mining Algorithms for mining Frequent Patterns" International Journal of Computer Applications (0975 – 8887) Volume 10– No.1, November 2010.
[28] Deepak Garg et. al. "Comparative Analysis of Various Approaches Used in Frequent Pattern Mining" (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence.
[29] Nasreen, Shamila, et al. "Frequent pattern mining algorithms for finding associated frequent patterns for data streams: A survey." Procedia Computer Science 37 (2014): 109-116.
[30] Shawkat, Mai, et al. "An optimized FP-growth algorithm for discovery of association rules." The Journal of Supercomputing (2022): 1-28.
[31] Chen, I. Z., et al. "Image processing and capsule networks." Advances in Intelligent Systems and Computing 19.20 (2020): 137-139.
[32] Hasan, Md Mahamud, and Sadia Zaman Mishu. "An adaptive method for mining frequent itemsets based on apriori and FP growth algorithm." 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2). IEEE, 2018.
[33] Olawuni, Deborah Monisola, et al. "Inhibitors to women's right to the occupation of land: a closer look at Ajebamidele Community in Ile-Ife, Nigeria." Property Management (2022).

[34] Borah, Anindita, and Bhabesh Nath. "Tree based frequent and rare pattern mining techniques: a comprehensive structural and empirical analysis." SN Applied Sciences 1 (2019): 1-18.
[35] Srikrishnaswetha, Kone, Sandeep Kumar, and Md Rashid Mahmood. "A study on smart electronics voting machine using face recognition and aadhar verification with iot." Innovations in Electronics and Communication Engineering: Proceedings of the 7th ICIECE 2018. Springer Singapore, 2019.
[36] Siswanto, Boby, Evawaty Tanuar, and Rissa Rahmania. "Reshaped and Reduced Dimensionality Reduction Data Technique on Association Rule Mining." 2021 3rd International Symposium on Material and Electrical Engineering Conference (ISMEE). IEEE, 2021.
[37] Fitzsimon, Jayden, et al. "A Shapley Value Index for Market Basket Analysis: Efficient Computation Using an Harsanyi Dividend Representation." International Game Theory Review (IGTR) 24.04 (2022): 1-29.
[38] Chu, Tsai-Pin, Fan Wu, and Shih-Wen Chiang. "Mining frequent pattern using item-transformation method." Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05). IEEE, 2005.
[39] Hu, Ya-Han, and Yen-Liang Chen. "Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism." Decision support systems 42.1 (2006): 1-24.
[40] Hadzic, Fedja, Henry Tan, and Tharam S. Dillon. Mining of data with complex structures. Vol. 333. New York: Springer, 2010.
[41] Han, Jiawei, Hong Cheng, Dong Xin, and Xifeng Yan. "Frequent pattern mining: current status and future directions." Data mining and knowledge discovery 15, no. 1 (2007): 55-86.
[42] Aggarwal, Charu C., Mansurul A. Bhuiyan, and Mohammad Al Hasan. Frequent pattern mining algorithms: A survey. Springer International Publishing, 2014.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY