

Research on Frequent Itemset Mining Using the Apriori Algorithm

Shiwen Zhu

1024040921

Nanjing University of Posts and

Telecommunications

School of Computer Science

Nanjing, China

Abstract—The Apriori algorithm is one of the most well-known algorithms for frequent itemset mining in the field of data mining. This study employs the Apriori algorithm for frequent itemset mining, aiming to identify high-support frequent itemsets from a given transactional dataset. Through a stepwise analysis of the dataset, the Apriori algorithm first generates candidate 1-itemsets and filters out the frequent 1-itemsets based on a minimum support threshold. Then, the algorithm iteratively generates candidate k-itemsets, computes their support, and filters out frequent itemsets until no new frequent itemsets can be found. Experimental results demonstrate that the Apriori algorithm is effective in identifying frequent itemsets in large-scale datasets, and its efficiency remains relatively stable as the size of the itemsets increases. This paper illustrates the application of the Apriori algorithm in practical data analysis, proving its feasibility and effectiveness in frequent itemset mining tasks. Through experimental analysis with varying support thresholds, the study further validates the algorithm's advantages and limitations in real-world applications.

Keywords—Apriori algorithm, frequent itemsets, data mining

I. INTRODUCTION

Frequent itemset mining, as a fundamental task in data mining, aims to discover frequently occurring itemsets from large-scale transactional data. These itemsets can reveal potential associations, making them widely applicable in fields such as market analysis, recommendation systems, and various other domains. Since its introduction by Agrawal and Srikant in 1994, the Apriori algorithm has become one of the classic algorithms in frequent itemset mining. Based on the "Apriori property"—that any subset of a frequent itemset must also be frequent—the algorithm generates and filters candidate itemsets in each iteration, significantly reducing computational complexity^[1].

However, despite its conceptual clarity and ease of implementation, the Apriori algorithm still faces significant performance bottlenecks when dealing with large-scale datasets, particularly in terms of the high computational cost associated with multiple database scans and the candidate itemset generation process^[2]. To address these issues, many scholars have proposed optimization methods based on Apriori. For instance, some approaches use data compression techniques to reduce the number of database scans, while others improve the candidate itemset generation process to enhance algorithm efficiency^[3].

In recent years, research on the Apriori algorithm has continued to deepen, with advancements not only in its theoretical development but also in its practical applications.

For example, Fournier-Viger et al. proposed an optimized Apriori algorithm based on horizontal scanning, which reduces the number of database scans and improves efficiency^[4]. Additionally, with the development of big data technologies, some studies have combined the Apriori algorithm with the MapReduce framework, leveraging the advantages of distributed computing to further improve the ability to process large-scale datasets^[5]. An improved algorithm proposed by Jiao in 2013 showed that the enhanced approach is reasonable and effective, extracting more valuable information^[6]. Moreover, hash-based techniques have been introduced to boost the efficiency of the Apriori algorithm when handling large-scale datasets^[7]. These advancements ensure that the Apriori algorithm remains highly relevant in the era of big data.

The application fields of the Apriori algorithm are also expanding, ranging from business intelligence to bioinformatics^[8]. In the commercial sector, the Apriori algorithm is used for market basket analysis to analyze customer purchasing patterns and increase sales of specific products^[9]. In bioinformatics, the algorithm has been applied to the analysis of gene expression data to uncover association rules between genes^[10].

Although the Apriori algorithm has certain limitations, its basic framework continues to be widely used in the field of frequent itemset mining and has achieved significant results in many practical applications. This study aims to enhance the efficiency of the Apriori algorithm for large-scale datasets and, through experimental analysis, to verify its applicability in real-world data. We hope to provide new insights and directions for the further development and optimization of the Apriori algorithm.

II. RELATED WORKS

Since its inception, the efficiency and performance of the Apriori algorithm have been a hot topic of research. Jiao, Ya Bing proposed an improved Apriori algorithm in 2013, which demonstrated its rationality and effectiveness through experimental results, capable of extracting more valuable information. Furthermore, D Cheng et al. optimized the Apriori algorithm using Amazon Web Services and GPU to enhance its data mining speed^[11]. L Alarabi proposed a method for accelerating frequent itemset mining based on the MapReduce framework in 2017^[12]. Sharma A et al. explored cloud computing-based association rule mining strategies in 2021^[13].

The Apriori algorithm has been widely applied in various fields due to its capability in mining frequent itemsets. In

sports data information management, the Apriori algorithm, combined with web log mining technology, is used to collect user behavior data and reveal the relationships between different pieces of information through frequent itemset mining and association rule mining, thereby improving the accuracy and efficiency of retrieval. In the field of medical diagnosis, the Apriori algorithm is used to assist doctors in making more accurate diagnoses and risk assessments. For instance, a study utilized the Apriori algorithm to detect rehabilitation nursing staff in hospitals, designing and constructing a medical intelligent system. In e-commerce, the Apriori algorithm is employed to mine potential customers by analyzing transaction data to discover patterns in customer purchasing behavior.

With the advent of the big data era, the parallelization and distributed implementation of the Apriori algorithm have become a focus of research. Agrawal, R. and Srikant, R. introduced the Apriori algorithm in 1994, and subsequent researchers have explored its implementation on Hadoop-MapReduce frameworks and Spark. Qiu, H. et al. proposed a parallel frequent itemset mining algorithm, Yafim, based on Spark in 2014^[14]. Rathee, S. et al. presented the R-Apriori algorithm in 2015, an efficient Apriori algorithm based on Spark^[15].

The performance of the Apriori algorithm is also often compared with other algorithms to assess its performance under different circumstances. Sharma, A. and Ganpati, A. provided a comparative review of association rule mining algorithms in 2021^[16]. Wicaksono, D. et al. compared the performance of the Apriori algorithm with the FP-growth algorithm in discovering frequent data patterns^[17]. First, confirm that you have the correct template for your paper size. This template has been tailored for output on the A4 paper size. If you are using US letter-sized paper, please close this file and download the Microsoft Word, Letter file.

III. PROBLEM STATEMENT

In the context of the big data era, extracting valuable information from massive datasets has become particularly important. Frequent itemset mining, as a fundamental technique in the field of data mining, aims to discover combinations of items that frequently occur together in a dataset. The Apriori algorithm, as a classic method for frequent itemset mining, has been widely researched and applied due to its intuitive strategy and broad applicability.

Despite the theoretical and practical successes of the Apriori algorithm, its efficiency and scalability issues have gradually emerged when dealing with large-scale datasets. The algorithm requires multiple scans of the entire dataset, and generates a large amount of redundant calculations when creating candidate item sets, which limits its performance in big data environments.

The main objective of this paper is to optimize the Apriori algorithm to enhance its efficiency and accuracy when processing large-scale datasets. The research will focus on reducing the number of data scans, optimizing the generation strategy of candidate item sets, and parallel processing. It is expected that through the optimizations of this study, the time complexity and space complexity of the Apriori algorithm when dealing with large-scale datasets will be significantly reduced, while maintaining or improving the accuracy of the mining results.

To achieve these enhancements, the paper will also consider the impact of data preprocessing techniques on the efficiency of the Apriori algorithm. By carefully selecting and transforming the data before mining, the algorithm can operate more efficiently, leading to faster discovery of frequent itemsets. Furthermore, the study will evaluate the effectiveness of various parameter tuning strategies to optimize the algorithm's performance for different types of datasets.

IV. ALGORITHMS

The Apriori algorithm is a foundational approach in the field of data mining, specifically designed for frequent itemset mining and association rule learning. It operates on transactional databases, identifying items that frequently co-occur and using these to infer association rules that reveal underlying trends within the data. This section provides a detailed description of the Apriori algorithm, including its methodology and the mathematical formulas that underpin its operation.

A. Input and Output

1) *Input*: A transactional database D , where each transaction T_i is a set of items.

2) *Output*: A set of frequent itemsets L that meet or exceed a minimum support threshold $min_support$, and a set of strong association rules that meet or exceed a minimum confidence threshold $min_confidence$.

B. Algorithm Description

1) *Initialization*: Set a minimum support threshold $min_support$, which determines the frequency below which an itemset is considered uninteresting. Set a minimum confidence threshold $min_confidence$, which determines the strength of the implication in the rules.

2) *Generate Initial Candidates*: Create C_1 , the set of all single items, and calculate their support in the database D . Select items with support greater than or equal to $min_support$ to form the first set of frequent itemsets L_1 .

3) *Iterative Generation of Candidates*: For each k – items L_{k-1} , generate C_k using the AprioriGen function:

$$C_k = \{A \cup B | A \in L_{k-1}, B \in L_{k-1}, A \neq B, \text{ and } \forall C \in A \cup B, C \in L_{k-2}\}$$

This function creates new candidates by combining items from the previous level of frequent itemsets, ensuring that all non-empty subsets of the union are also frequent.

4) *Count Support and Prune*: Calculate the support of each candidate in C_k and add those that meet or exceed $min_support$ to L_k . Remove candidates with support below $min_support$, thus pruning the search space.

5) *Repeat Until No More Candidates*: Repeat steps 3 and 4 until no more candidates can be generated or until the support of the candidates falls below $min_support$.

6) *Generate Association Rules*: For each frequent itemset X , generate all non-empty subsets Y and calculate the confidence of the rule $X \Rightarrow Y$ using the formula:

$$Confidence(X \Rightarrow Y) = \frac{Support(X \cup Y)}{Support(X)}$$

Select rules where the confidence is greater than or equal to $min_confidence$ to form the final set of strong association rules.

7) *Evaluation Metrics*: Support Degree is the proportion of the number of occurrences of several related data to the total number of data sets, or the probability of the occurrence of several data associations:

$$Support(X, Y) = P(XY) = \frac{num(XY)}{num(AllSamples)}$$

Confidence Degree is the probability that a transaction including Y also includes X :

$$Confidence(X \Rightarrow Y) = P(X | Y) = \frac{P(XY)}{P(Y)}$$

Lift Degree is the ratio of the probability of the occurrence of X under the condition of containing Y to the probability of containing X :

$$Lift(X \Rightarrow Y) = \frac{P(X | Y)}{P(X)} = \frac{Confidence(X \Rightarrow Y)}{P(X)}$$

The lift degree reflects the relationship between X and Y ; if it is greater than 1, $X \Rightarrow Y$ is a strong association rule; if it is less than or equal to 1, $X \Rightarrow Y$ is not a strong association rule.

C. Operational Process

The Apriori algorithm employs a "bottom-up" approach, iteratively extending frequent subsets by one item at a time, and testing groups of candidates against the data. The process terminates when no further successful extensions are found.

The algorithm begins by generating a candidate set of single items and their corresponding support degrees, pruning candidates with support degrees lower than the minimum support degree to obtain the frequent set of single items. This set is then used to generate the candidate frequent set of two items, pruning candidates with support degrees lower than the minimum support degree to get the frequent set of two items, and so on. The iteration will stop when it is unable to find a frequent set of $k + 1$ items.

V. EVALUATION

In this section, we evaluate the performance of the Apriori algorithm when applied to a specific dataset containing 1892 items. The goal is to mine frequent itemsets and generate association rules through the Apriori algorithm. Below is a detailed evaluation of the algorithm's performance.

A. Dataset and Parameter Settings

1) *Dataset Size*: The dataset contains 1892 items, distributed across multiple transactions.

2) *Minimum Support Threshold*: Set to 0.01, meaning that item sets with a support degree not lower than 1% are considered frequent.

3) *Minimum Confidence Threshold*: Set to 0.07, indicating that only rules with a confidence level not lower than 7% are considered strong association rules.

B. Frequent Itemset Results

A total of 335 frequent itemsets were identified, with support degrees ranging from 0.01004 to 0.09989. This indicates a significant number of frequent patterns in the dataset, suitable for further association rule mining. The experimental results are shown in Fig. 1 to Fig. 4.

1. Itemset: {9}, Support: 0.05550	31. Itemset: {679}, Support: 0.02061	61. Itemset: {586}, Support: 0.01488
2. Itemset: {153}, Support: 0.01533	32. Itemset: {162}, Support: 0.01903	62. Itemset: {124}, Support: 0.01858
3. Itemset: {401}, Support: 0.01533	33. Itemset: {119}, Support: 0.02114	63. Itemset: {207}, Support: 0.01691
4. Itemset: {151}, Support: 0.03866	34. Itemset: {125}, Support: 0.01858	64. Itemset: {23}, Support: 0.02696
5. Itemset: {71}, Support: 0.03224	35. Itemset: {53}, Support: 0.02748	65. Itemset: {57}, Support: 0.02061
6. Itemset: {117}, Support: 0.01903	36. Itemset: {163}, Support: 0.01533	66. Itemset: {129}, Support: 0.01596
7. Itemset: {262}, Support: 0.01857	37. Itemset: {58}, Support: 0.02748	67. Itemset: {83}, Support: 0.01744
8. Itemset: {41}, Support: 0.01634	38. Itemset: {153}, Support: 0.01638	68. Itemset: {132}, Support: 0.01226
9. Itemset: {76}, Support: 0.02060	39. Itemset: {580}, Support: 0.01057	69. Itemset: {130}, Support: 0.02060
10. Itemset: {15}, Support: 0.05205	40. Itemset: {126}, Support: 0.02484	70. Itemset: {136}, Support: 0.02061
11. Itemset: {405}, Support: 0.01163	41. Itemset: {168}, Support: 0.01374	71. Itemset: {4}, Support: 0.04670
12. Itemset: {89}, Support: 0.02431	42. Itemset: {83}, Support: 0.06342	72. Itemset: {77}, Support: 0.01488
13. Itemset: {29}, Support: 0.04070	43. Itemset: {169}, Support: 0.01797	73. Itemset: {142}, Support: 0.02404
14. Itemset: {18}, Support: 0.03558	44. Itemset: {171}, Support: 0.03191	74. Itemset: {131}, Support: 0.01684
15. Itemset: {61}, Support: 0.02696	45. Itemset: {170}, Support: 0.02126	75. Itemset: {282}, Support: 0.01216
16. Itemset: {177}, Support: 0.01216	46. Itemset: {146}, Support: 0.02326	76. Itemset: {59}, Support: 0.02643
17. Itemset: {186}, Support: 0.01216	47. Itemset: {248}, Support: 0.01268	77. Itemset: {34}, Support: 0.01744
18. Itemset: {62}, Support: 0.01857	48. Itemset: {253}, Support: 0.01338	78. Itemset: {11}, Support: 0.05497
19. Itemset: {51}, Support: 0.03025	49. Itemset: {144}, Support: 0.01216	79. Itemset: {140}, Support: 0.02061
20. Itemset: {6}, Support: 0.07135	50. Itemset: {94}, Support: 0.02808	80. Itemset: {45}, Support: 0.01638
21. Itemset: {223}, Support: 0.01110	51. Itemset: {94}, Support: 0.02220	81. Itemset: {79}, Support: 0.02114
22. Itemset: {7}, Support: 0.07082	52. Itemset: {229}, Support: 0.01638	82. Itemset: {17}, Support: 0.02220
23. Itemset: {18}, Support: 0.03594	53. Itemset: {170}, Support: 0.01374	83. Itemset: {150}, Support: 0.01638
24. Itemset: {19}, Support: 0.05097	54. Itemset: {171}, Support: 0.01110	84. Itemset: {172}, Support: 0.01956
25. Itemset: {121}, Support: 0.02484	55. Itemset: {24}, Support: 0.03912	85. Itemset: {63}, Support: 0.02646
26. Itemset: {40}, Support: 0.01110	56. Itemset: {215}, Support: 0.01110	86. Itemset: {108}, Support: 0.01110
27. Itemset: {173}, Support: 0.01427	57. Itemset: {234}, Support: 0.01374	87. Itemset: {28}, Support: 0.03350
28. Itemset: {66}, Support: 0.01318	58. Itemset: {97}, Support: 0.02484	88. Itemset: {69}, Support: 0.02748
29. Itemset: {192}, Support: 0.01321	59. Itemset: {207}, Support: 0.01586	89. Itemset: {263}, Support: 0.01718
30. Itemset: {473}, Support: 0.01004	60. Itemset: {208}, Support: 0.01057	90. Itemset: {71}, Support: 0.02060

Fig. 1. Frequent item sets 1 to 90.

91. Itemset: {151}, Support: 0.02326	181. Itemset: {168}, Support: 0.01427	191. Itemset: {197}, Support: 0.01163
92. Itemset: {154}, Support: 0.01488	182. Itemset: {148}, Support: 0.01797	192. Itemset: {138}, Support: 0.01864
93. Itemset: {159}, Support: 0.01638	183. Itemset: {173}, Support: 0.01533	193. Itemset: {210}, Support: 0.01638
94. Itemset: {143}, Support: 0.02167	184. Itemset: {253}, Support: 0.01427	194. Itemset: {100}, Support: 0.02378
95. Itemset: {141}, Support: 0.01956	185. Itemset: {293}, Support: 0.01797	195. Itemset: {139}, Support: 0.02854
96. Itemset: {156}, Support: 0.01797	186. Itemset: {28}, Support: 0.03180	196. Itemset: {92}, Support: 0.01243
97. Itemset: {157}, Support: 0.01858	187. Itemset: {93}, Support: 0.02061	197. Itemset: {68}, Support: 0.01458
98. Itemset: {189}, Support: 0.01638	188. Itemset: {119}, Support: 0.01638	198. Itemset: {250}, Support: 0.01638
99. Itemset: {422}, Support: 0.01163	189. Itemset: {44}, Support: 0.01427	199. Itemset: {179}, Support: 0.01857
100. Itemset: {69}, Support: 0.01858	190. Itemset: {108}, Support: 0.01427	200. Itemset: {131}, Support: 0.01638
101. Itemset: {425}, Support: 0.01268	191. Itemset: {111}, Support: 0.02220	201. Itemset: {132}, Support: 0.01321
102. Itemset: {500}, Support: 0.01216	192. Itemset: {72}, Support: 0.03426	202. Itemset: {179}, Support: 0.01857
103. Itemset: {200}, Support: 0.02526	193. Itemset: {43}, Support: 0.01638	203. Itemset: {137}, Support: 0.01638
104. Itemset: {95}, Support: 0.01321	194. Itemset: {108}, Support: 0.01427	204. Itemset: {131}, Support: 0.01638
105. Itemset: {225}, Support: 0.01268	195. Itemset: {265}, Support: 0.01374	205. Itemset: {211}, Support: 0.01321
106. Itemset: {355}, Support: 0.01110	196. Itemset: {22}, Support: 0.03277	206. Itemset: {246}, Support: 0.01864
107. Itemset: {245}, Support: 0.01110	197. Itemset: {226}, Support: 0.01691	207. Itemset: {230}, Support: 0.01638
108. Itemset: {189}, Support: 0.01638	198. Itemset: {185}, Support: 0.02326	208. Itemset: {137}, Support: 0.01638
109. Itemset: {251}, Support: 0.01216	199. Itemset: {264}, Support: 0.01321	209. Itemset: {212}, Support: 0.01374
110. Itemset: {252}, Support: 0.01427	200. Itemset: {63}, Support: 0.02484	210. Itemset: {88}, Support: 0.01374
111. Itemset: {99}, Support: 0.01427	201. Itemset: {135}, Support: 0.02220	211. Itemset: {61}, Support: 0.01797
112. Itemset: {147}, Support: 0.01797	202. Itemset: {209}, Support: 0.01691	212. Itemset: {283}, Support: 0.01638
113. Itemset: {24}, Support: 0.03708	203. Itemset: {116}, Support: 0.01110	213. Itemset: {139}, Support: 0.01638
114. Itemset: {183}, Support: 0.01956	204. Itemset: {137}, Support: 0.01374	214. Itemset: {168}, Support: 0.02378
115. Itemset: {187}, Support: 0.01488	205. Itemset: {284}, Support: 0.01057	215. Itemset: {426}, Support: 0.01638
116. Itemset: {283}, Support: 0.01427	206. Itemset: {364}, Support: 0.01321	216. Itemset: {369}, Support: 0.01864
117. Itemset: {2}, Support: 0.09408	207. Itemset: {188}, Support: 0.01163	217. Itemset: {132}, Support: 0.01638
118. Itemset: {3}, Support: 0.09408	208. Itemset: {40}, Support: 0.01057	218. Itemset: {139}, Support: 0.01638
119. Itemset: {202}, Support: 0.01427	209. Itemset: {116}, Support: 0.01797	219. Itemset: {139}, Support: 0.01638
120. Itemset: {12}, Support: 0.02060	210. Itemset: {137}, Support: 0.01797	220. Itemset: {11}, Support: 0.05212

Fig. 2. Frequent item sets 91 to 180.

181. Itemset: {482}, Support: 0.01858	231. Itemset: {482}, Support: 0.01374	241. Itemset: {119}, Support: 0.01638
182. Itemset: {121}, Support: 0.01216	232. Itemset: {222}, Support: 0.01321	242. Itemset: {332}, Support: 0.01638
183. Itemset: {54}, Support: 0.02326	233. Itemset: {237}, Support: 0.01374	243. Itemset: {519}, Support: 0.01110
184. Itemset: {84}, Support: 0.01488	234. Itemset: {275}, Support: 0.01586	244. Itemset: {268}, Support: 0.01638
185. Itemset: {213}, Support: 0.01858	235. Itemset: {280}, Support: 0.01427	245. Itemset: {174}, Support: 0.01638
186. Itemset: {124}, Support: 0.01216	236. Itemset: {153}, Support: 0.01216	246. Itemset: {215}, Support: 0.01638
187. Itemset: {212}, Support: 0.01691	237. Itemset: {186}, Support: 0.01691	247. Itemset: {414}, Support: 0.01864
188. Itemset: {446}, Support: 0.01321	238. Itemset: {348}, Support: 0.01216	248. Itemset: {66}, Support: 0.01638
189. Itemset: {64}, Support: 0.01110	239. Itemset: {210}, Support: 0.01638	249. Itemset: {183}, Support: 0.01586
190. Itemset: {256}, Support: 0.01576	240. Itemset: {184}, Support: 0.01797	250. Itemset: {271}, Support: 0.01638
191. Itemset: {175}, Support: 0.02167	241. Itemset: {680}, Support: 0.01268	251. Itemset: {245}, Support: 0.01374
192. Itemset: {112}, Support: 0.02061	242. Itemset: {360}, Support: 0.01163	252. Itemset: {48}, Support: 0.01330
193. Itemset: {385}, Support: 0.01864	243. Itemset: {239}, Support: 0.01903	253. Itemset: {322}, Support: 0.01110
194. Itemset: {152}, Support: 0.01857	244. Itemset: {513}, Support: 0.01321	254. Itemset: {159}, Support: 0.01903
195. Itemset: {180}, Support: 0.02061	245. Itemset: {283}, Support: 0.01638	255. Itemset: {207}, Support: 0.01638
196. Itemset: {60}, Support: 0.03025	246. Itemset: {189}, Support: 0.01858	256. Itemset: {139}, Support: 0.01903
197. Itemset: {411}, Support: 0.01374	247. Itemset: {298}, Support: 0.01268	257. Itemset: {203}, Support: 0.01543
198. Itemset: {296}, Support: 0.01374	248. Itemset: {164}, Support: 0.01094	258. Itemset: {498}, Support: 0.01110
199. Itemset: {594}, Support: 0.01216	249. Itemset: {654}, Support: 0.01057	259. Itemset: {217}, Support: 0.01374
200. Itemset: {100}, Support: 0.01374	250. Itemset: {113}, Support: 0.01748	260. Itemset: {164}, Support: 0.01638
201. Itemset: {80}, Support: 0.02173	251. Itemset: {683}, Support: 0.01057	261. Itemset: {435}, Support: 0.01427
202. Itemset: {42}, Support: 0.03224	252. Itemset: {422}, Support: 0.01084	262. Itemset: {86}, Support: 0.01321
203. Itemset: {23}, Support: 0.04334	253. Itemset: {655}, Support: 0.01321	263. Itemset: {162}, Support: 0.01638
204. Itemset: {279}, Support: 0.01110	254. Itemset: {165}, Support: 0.01268	264. Itemset: {188}, Support: 0.02060
205. Itemset: {54}, Support: 0.01638	255. Itemset: {166}, Support: 0.01586	265. Itemset: {107}, Support: 0.01638
206. Itemset: {216}, Support: 0.01110	256. Itemset: {194}, Support: 0.01163	266. Itemset: {190}, Support: 0.01321
207. Itemset: {27}, Support: 0.02220	257. Itemset: {47}, Support: 0.01374	267. Itemset: {308}, Support: 0.01857
208. Itemset: {91}, Support: 0.02326	258. Itemset: {328}, Support: 0.01216	268. Itemset: {162}, Support: 0.01638
209. Itemset: {133}, Support: 0.01268	259. Itemset: {372}, Support: 0.01268	269. Itemset: {684}, Support: 0.01864
210. Itemset: {231}, Support: 0.01004	260. Itemset: {415}, Support: 0.01488	270. Itemset: {781}, Support: 0.02220

Fig. 3. Frequent item sets 181 to 270.

271. Itemset: {146}, Support: 0.01374	341. Itemset: {157}, Support: 0.01110	351. Itemset: {7}, Support: 0.01216
272. Itemset: {90}, Support: 0.01858	342. Itemset: {185}, Support: 0.01268	352. Itemset: {7}, Support: 0.01216
273. Itemset: {165}, Support: 0.01858	343. Itemset: {25}, Support: 0.01488	353. Itemset: {2}, Support: 0.01374
274. Itemset: {247}, Support: 0.01216	344. Itemset: {20}, Support: 0.01691	354. Itemset: {3}, Support: 0.01857
275. Itemset: {219}, Support: 0.01163	345. Itemset: {166}, Support: 0.01858	355. Itemset: {2}, Support: 0.01374
276. Itemset: {201}, Support: 0.02126	346. Itemset: {12}, Support: 0.01004	
277. Itemset: {196}, Support: 0.01857	347. Itemset: {17}, Support: 0.01216	
278. Itemset: {158}, Support: 0.01163	348. Itemset: {12}, Support: 0.01748	
279. Itemset: {115}, Support: 0.01864	349. Itemset: {18}, Support: 0.01748	
280. Itemset: {2}, Support: 0.02126	350. Itemset: {11}, Support: 0.01216	
281. Itemset: {1}, Support: 0.01857	351. Itemset: {1}, Support: 0.01374	
282. Itemset: {19}, Support: 0.01216	352. Itemset: {1}, Support: 0.01488	
283. Itemset: {71}, Support: 0.01163	353. Itemset: {1}, Support: 0.01216	
284. Itemset: {2}, Support: 0.02126	354. Itemset: {164}, Support: 0.01268	
285. Itemset: {5}, Support: 0.01427	355. Itemset: {2}, Support: 0.01638	
286. Itemset: {58}, Support: 0.01797	356. Itemset: {202}, Support: 0.01084	
287. Itemset: {1}, Support: 0.01748	357. Itemset: {2}, Support: 0.01110	
288. Itemset: {2}, Support: 0.01057	358. Itemset: {2}, Support: 0.01374	
289. Itemset: {147}, Support: 0.01121	359. Itemset: {2}, Support: 0.01374	
290. Itemset: {24}, Support: 0.01163	360. Itemset: {43}, Support: 0.01321	
291. Itemset: {60}, Support: 0.01427	361. Itemset: {19}, Support: 0.01163	
292. Itemset: {1}, Support: 0.02060	362. Itemset: {19}, Support: 0.01163	
293. Itemset: {121}, Support: 0.01163	363. Itemset: {166}, Support: 0.01163	
294. Itemset: {2}, Support: 0.01857	364. Itemset: {144}, Support: 0.01163	
295. Itemset: {2}, Support: 0.01163	365. Itemset: {108}, Support: 0.01163	
296. Itemset: {18}, Support: 0.01427	366. Itemset: {108}, Support: 0.01163	
297. Itemset: {18}, Support: 0.01427	367. Itemset: {16}, Support: 0.01857	
298. Itemset: {128}, Support: 0.01427	368. Itemset: {119}, Support: 0.01163	
299. Itemset: {28}, Support: 0.01004	369. Itemset: {16}, Support: 0.01163	
300. Itemset: {27}, Support: 0.01163	370. Itemset: {2}, Support: 0.01163	

Fig. 4. Frequent item sets 271 to 355.

C. Association Rule Results

A significant number of association rules were generated from the frequent itemsets, all meeting the minimum confidence threshold. The confidence levels of these rules

range from 0.10582 to 0.86364, demonstrating a wide variation in rule strength. The results are shown in Table 1.

TABLE I. ASSOCIATION RULE RESULTS

Rule	Confidence	Rule	Confidence
{9}, {10}	0.38095	{10}, {9}	0.54795
{9}, {1}	0.19048	{1}, {9}	0.10582
{19}, {35}	0.22115	{35}, {19}	0.39655
{2}, {71}	0.22472	{71}, {2}	0.65574
{3}, {71}	0.29197	{71}, {3}	0.65574
{5}, {7}	0.23684	{7}, {5}	0.20149
{58}, {5}	0.65385	{5}, {58}	0.29825
{1}, {5}	0.17460	{5}, {1}	0.28947
{2}, {5}	0.11236	{5}, {2}	0.17544
{147}, {6}	0.73529	{6}, {147}	0.18519
{24}, {6}	0.58571	{6}, {24}	0.30370
{40}, {7}	0.45763	{7}, {40}	0.20149
{1}, {7}	0.24868	{7}, {1}	0.35075
{131}, {7}	0.77273	{7}, {131}	0.25373
{2}, {7}	0.11236	{7}, {2}	0.14925
{3}, {7}	0.16058	{7}, {3}	0.16418
{98}, {7}	0.47368	{7}, {98}	0.20149
{18}, {22}	0.47059	{22}, {18}	0.51613
{128}, {20}	0.57447	{20}, {128}	0.27551
{25}, {23}	0.30159	{23}, {25}	0.23171
{57}, {53}	0.56410	{53}, {57}	0.43137
{57}, {185}	0.53846	{185}, {57}	0.70000
{105}, {26}	0.54545	{26}, {105}	0.39344
{26}, {63}	0.45902	{63}, {26}	0.59574
{20}, {77}	0.32653	{77}, {20}	0.48485
{160}, {77}	0.42222	{77}, {160}	0.28788
{2}, {11}	0.10674	{11}, {2}	0.18269
{17}, {79}	0.54762	{79}, {17}	0.57500
{2}, {69}	0.18539	{69}, {2}	0.63462
{3}, {69}	0.24088	{69}, {3}	0.63462
{1}, {131}	0.13228	{131}, {1}	0.56818
{1}, {2}	0.16402	{2}, {1}	0.17416
{1}, {3}	0.14815	{3}, {1}	0.20438
{1}, {50}	0.12169	{50}, {1}	0.74194
{166}, {14}	0.57143	{14}, {166}	0.24096
{2}, {3}	0.43820	{3}, {2}	0.56934
{202}, {2}	0.70370	{2}, {202}	0.10674
{2}, {148}	0.11798	{148}, {2}	0.61765
{2}, {43}	0.16292	{43}, {2}	0.70732
{2}, {139}	0.14045	{139}, {2}	0.69444
{43}, {3}	0.60976	{3}, {43}	0.18248
{3}, {139}	0.16058	{139}, {3}	0.61111
{293}, {23}	0.64706	{23}, {293}	0.26829
{160}, {20}	0.55556	{20}, {160}	0.25510
{44}, {55}	0.63043	{55}, {44}	0.72500
{105}, {63}	0.50000	{63}, {105}	0.46809
{100}, {39}	0.53333	{39}, {100}	0.44444
{16}, {233}	0.21053	{233}, {16}	0.64516
{16}, {113}	0.23158	{113}, {16}	0.66667
{16}, {48}	0.30526	{48}, {16}	0.46032
{42}, {23}	0.52459	{23}, {42}	0.39024
{27}, {86}	0.54762	{86}, {27}	0.92000
{2}, {3, 71}	0.16854	{3}, {2, 71}	0.21898
{71}, {2, 3}	0.49180	{2, 3}, {71}	0.38462
{2, 71}, {3}	0.75000	{3, 71}, {2}	0.75000
{2}, {3, 69}	0.14607	{3}, {2, 69}	0.18978
{69}, {2, 3}	0.50000	{2, 3}, {69}	0.33333
{2, 69}, {3}	0.78788	{3, 69}, {2}	0.78788
{3}, {2, 43}	0.14599	{2}, {43, 3}	0.11236
{43}, {2, 3}	0.48780	{2, 3}, {43}	0.25641
{43, 3}, {2}	0.80000	{2, 43}, {3}	0.68966
{2}, {3, 139}	0.10674	{3}, {2, 139}	0.13869
{139}, {2, 3}	0.52778	{2, 3}, {139}	0.24359
{2, 139}, {3}	0.76000	{3, 139}, {2}	0.86364

D. Performance Analysis

1) *Support Degree Distribution*: The support degrees of the frequent itemsets show a broad distribution, with the majority of itemsets having low support but a notable few with higher support, such as itemset {1} with a support degree of 0.09989. This suggests that certain items are more prevalent or significant within the dataset.

2) *Confidence Analysis*: The confidence of the generated association rules varies significantly, indicating that while some rules are strongly supported by the data, others are less so. High-confidence rules, such as those with confidence levels above 0.5, may be particularly valuable for decision-making.

3) *Rule Quality*: The quality of the rules is reflected in their confidence values. For instance, the rule involving itemset {9, 10} with a confidence of 0.38095 indicates a strong relationship between these items. Such high-confidence rules can provide actionable insights for business strategies.

VI. CONCLUSION

In this study, we have implemented and evaluated the Apriori algorithm to mine frequent itemsets and generate association rules from a dataset comprising 1892 items. The algorithm was applied with a minimum support threshold of 0.01 and a minimum confidence threshold of 0.07, which allowed us to identify 335 frequent itemsets with varying support degrees. The results demonstrated the ability of the Apriori algorithm to uncover significant patterns within the dataset, with support degrees ranging from as low as 0.01004 to as high as 0.09989.

The frequent itemsets served as the basis for generating association rules, which were filtered to ensure a minimum confidence level of 0.07. This stringent criterion ensured that only the strongest and most reliable rules were retained for further analysis. The rules generated provide valuable insights into the underlying relationships between items in the dataset, which can be instrumental for decision-making processes in various domains such as retail, marketing, and inventory management.

The evaluation of the Apriori algorithm's performance revealed its effectiveness in handling the dataset, despite the challenges posed by the large number of items. The algorithm successfully identified a substantial number of frequent itemsets and generated a set of high-confidence association rules. However, the process also highlighted the need for optimization, particularly in terms of runtime and memory consumption, to handle even larger datasets efficiently.

In conclusion, the Apriori algorithm has proven to be a robust tool for frequent itemset mining and association rule generation. The insights gained from this analysis can be leveraged to enhance business strategies, optimize operations, and predict trends based on the identified patterns in the data. Furthermore, the findings from this study provide an empirical foundation for further improvements to the algorithm, particularly in enhancing data processing efficiency and accuracy. In the future, we look forward to applying the Apriori algorithm to a broader range of datasets and exploring its potential applications across various industry contexts to achieve deeper insights and decision support.

REFERENCES

- [1] Agrawal, R. "Fast Algorithms for Mining Association Rules." VLDB, 1994.
- [2] Han, Jiawei, Jian Pei, and Hanghang Tong. Data mining: concepts and techniques. Morgan kaufmann, 2022.
- [3] Ye Y, Chiang C C. A parallel apriori algorithm for frequent itemsets mining[C]//Fourth International Conference on Software Engineering Research, Management and Applications (SERA'06). IEEE, 2006: 87-94.
- [4] Fournier-Viger P, Lin J C W, Kiran R U, et al. A survey of sequential pattern mining[J]. Data Science and Pattern Recognition, 2017, 1(1): 54-77.
- [5] Saabith A L S, Sundararajan E, Bakar A A. Parallel implementation of apriori algorithms on the Hadoop-MapReduce platform-an evaluation of literature[J]. Journal of Theoretical and Applied Information Technology, 2016, 85(3): 321.
- [6] Yabing J. Research of an improved apriori algorithm in data mining association rules[J]. International Journal of Computer and Communication Engineering, 2013, 2(1): 25.
- [7] Tanha S, Biswas R T, Ritu T R, et al. Improved Apriori Algorithm Using Hash Technique[C]//2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM). IEEE, 2023: 1-6.
- [8] Pamnani H K, Raja L, Ives T. Developing a novel H-Apriori algorithm using support-leverage matrix for association rule mining[J]. International Journal of Information Technology, 2024: 1-11.
- [9] Lekireddy B R, Reddybathina N S R, Michael G, et al. Market-Based Analysis: Apriori approach to analyze purchase patterns[J]. EAI Endorsed Transactions on Scalable Information Systems, 2023, 10(5).
- [10] Yosef A, Roth I, Shnaider E, et al. Horizontal Learning Approach to Discover Association Rules[J]. Computers, 2024, 13(3): 62.
- [11] Cheng D, Rao J, Guo Y, et al. Improving mapreduce performance in heterogeneous environments with adaptive task tuning[C]//Proceedings of the 15th International Middleware Conference. 2014: 97-108.
- [12] Alarabi L. St-hadoop: A mapreduce framework for big spatio-temporal data[C]//Proceedings of the 2017 ACM International Conference on Management of Data. 2017: 40-42.
- [13] Sharma A, Ganpati A. Association rule mining algorithms: a Comparative review[J]. Int. Res. J. Eng. Technol, 2021, 8(11): 848-853.
- [14] Qiu H, Gu R, Yuan C, et al. Yafim: a parallel frequent itemset mining algorithm with spark[C]//2014 IEEE international parallel & distributed processing symposium workshops. IEEE, 2014: 1664-1671.
- [15] Rathee S, Kaul M, Kashyap A. R-Apriori: an efficient apriori based algorithm on spark[C]//Proceedings of the 8th workshop on Ph. D. Workshop in information and knowledge management. 2015: 27-34.
- [16] Sharma A, Ganpati A. Association rule mining algorithms: a Comparative review[J]. Int. Res. J. Eng. Technol, 2021, 8(11): 848-853.
- [17] Singh A K, Kumar A, Maurya A K. An empirical analysis and comparison of apriori and FP-growth algorithm for frequent pattern mining[C]//2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies. IEEE, 2014: 1599-1602.