

## Comparative Analysis of Association Rule Mining Algorithms: An Application to Grocery Store

<sup>1, \*</sup> Abdulhamed Toyin ABDULMUMEEN, <sup>2</sup> Ronke Seyi Babatude, <sup>3</sup> Jumoke Falilat AJAO, <sup>4</sup> R. M. Isiaka, <sup>5</sup> Isiaka Lukman Alage, <sup>6</sup> Ridwan Garba

<sup>1,2,3,4,6</sup> Department of Computer Science, Kwara State University, Malete, Nigeria

<sup>6</sup> African Regional Centre for Space Science and Technology Education in English (ARCSSTE-E), OAU, Nigeria

\*Corresponding Author: [abdulhamedabdulmumeen@gmail.com](mailto:abdulhamedabdulmumeen@gmail.com)

**ABSTRACT:** Market Basket Analysis (MBA) is a vital retail data mining technique that reveals patterns in customer transaction data, identifying products frequently purchased together, enabling retailers to optimize sales strategies, enhance customer experiences, and streamline inventory and marketing efforts. Using a dataset from Ilorin Shoprite, this study explores purchasing patterns and their applications in retail, marketing, and bioinformatics for informed decision-making. The study analyzes grocery transaction data to uncover significant product associations, aiding strategic decisions like catalog design and cross-selling, and compares three association rule mining algorithms Apriori, ECLAT, and FP-Growth evaluating their time efficiency, memory usage, and effectiveness across small and large datasets, with the goal of identifying the best algorithm for retail transaction data and providing actionable insights for businesses. Data preprocessing involved a structured grocery dataset, followed by applying Apriori, ECLAT, and FP-Growth to identify purchase associations, where analysis of 1000 unique transactions revealed key associations such as WHT BREAD SWEET and EVA WATER 750ML. FP-Growth outperformed others in speed for both small and large datasets, while Apriori and ECLAT excelled in generating frequent itemsets, making FP-Growth recommended for large-scale MBA due to its efficiency. The identified associations offer practical insights for refining catalog design and cross-selling strategies, and retailers should adopt FP-Growth to optimize MBA performance, leveraging purchasing patterns to enhance marketing and customer experiences across industries like retail and bioinformatics.

**KEYWORDS:** Data Mining, Database, Association Rule Mining, Association Rule, Market Basket Analysis, Apriori Algorithm, FP-Growth, Retail Optimization.

### 1. INTRODUCTION

In an increasingly data-driven world, companies, organizations, and government agencies rely on data mining to extract and categorize essential information from vast datasets, transforming raw data into actionable insights for strategic decision-making [1]. The retail sector, characterized by fierce competition, particularly benefits from leveraging information technology to analyze transaction data. This enables businesses to stay competitive by understanding consumer behavior and optimizing operations. For example, Reymart, a retail company founded in 2016, collects daily transaction data from a wide range of products but has not yet fully utilized this data for strategic purposes [2]. Data mining techniques allow retailers to uncover consumer spending patterns, balance supply and demand, and implement customer-centric strategies to adapt to evolving market trends [10]. Data mining, also known as Knowledge Discovery in Data (KDD), is the process of selecting, preprocessing, transforming, mining, and interpreting large datasets to extract valuable information [11]. Within data mining, Market Basket Analysis (MBA) is a key technique that uses association rule mining (ARM) to identify patterns in customer purchasing behavior by analyzing transactional data. MBA reveals which products are frequently purchased together, such as bread and butter, enabling businesses to make informed decisions about stocking, shelf organization, cross-selling, and upselling [12][13]. This technique is widely used across industries like retail, mail order, telemarketing, and e-commerce to identify product linkages and optimize product combinations [14]. Key algorithms for MBA include

Apriori, which iteratively identifies frequent itemsets based on support (transaction frequency) and confidence (pattern certainty) metrics to generate association rules [17]; FP-Growth, a memory-efficient algorithm that mines frequent patterns using a compact tree structure, avoiding candidate generation for large datasets [18]; Eclat, suitable for sparse datasets, which builds frequent itemsets by intersecting transaction IDs in a vertical data format [18]; and Apriori-TID and K-Apriori, variations of the Apriori algorithm that optimize performance for specific use cases [22]. Advanced MBA approaches, such as differential MBA, analyze variations across stores, times, or customer segments, while predictive MBA provides insights for cross-selling strategies [23]. These methods rely on metrics like support and confidence to identify strong association rules, helping businesses meet customer demands and enhance profitability [24].

The grocery retail industry faces the challenge of transforming vast transaction datasets into actionable insights to optimize operations and increase profitability. Understanding customer purchasing patterns, especially across different seasons, is critical for tailoring product offerings, improving supermarket layouts, and refining pricing strategies [25]. However, many retailers, such as Reymart, collect extensive transaction data but lack the tools or processes to analyze it effectively [26]. A research addresses this gap by employing Apriori, Apriori-TID, and FP-Growth algorithms to analyze grocery purchase patterns across seasons and generate association rules. The study aims to identify seasonal purchasing trends, evaluate algorithm performance, and simplify insights into consumer transactions to support better decision-making [26]. By uncovering product associations and consumer behavior, this research seeks to help retailers balance supply and demand, prioritize customer-centric strategies, and adapt to dynamic market trends, ultimately enhancing operational efficiency and profitability [26].

Data mining encompasses techniques such as association, clustering, classification, and prediction. Among these, association rule mining stands out as a powerful exploratory method, particularly in market basket analysis. This approach is widely applied in retail settings, such as supermarkets, to uncover patterns in customer purchasing behavior. By leveraging association rule mining, retailers can identify frequently purchased item combinations, providing a competitive edge for both large and small-scale operations. This paper explores the practical application of association rule mining, focusing on three key algorithms Apriori, ECLAT, and FP-Growth to analyze which items are commonly bought together in a supermarket setting.

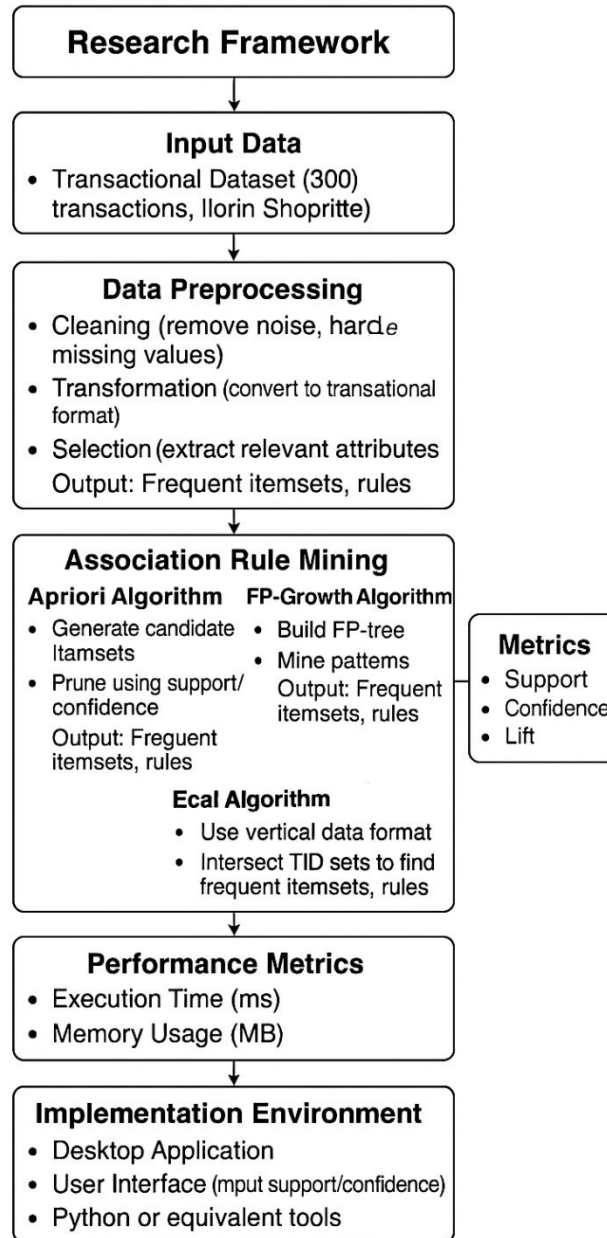
## II. RELATED WORKS

Organizations face challenges in understanding consumer behavior and extracting insights from large datasets to gain a competitive edge. Association rule mining is crucial for decision-making in competitive markets. Data mining techniques reveal critical patterns for retail success. This chapter reviews Apriori, ECLAT, and FP-Growth for Market Basket Analysis across various domains. These algorithms uncover consumer purchasing patterns to optimize strategies. The study compares their performance on a 300-transaction grocery dataset. FP-Growth is used for MBA to identify product associations in transaction data. Apriori-based MBA with Python enhances retail customer loyalty. It integrates IoT and data mining for decision-making. The study improves satisfaction through loyalty programs. Apriori and Hash-Based algorithms for MBA are compared using RapidMiner. Hash-Based outperforms Apriori for consumer insights. The study optimizes product offerings at Reymart Stores [2]. Apriori-based MBA with Python libraries identifies product relationships. It optimizes product placement and sales. The study enhances customer experiences through data-driven insights [3]. MBA at Ariana Audy Beauty Clinic uses RapidMiner to identify popular products. The study optimizes stock and marketing strategies. It enhances customer satisfaction through data-driven insights [4]. Enhanced MBA with Apriori improves e-commerce product recommendations. It analyzes product feature correlations using Levenshtein distance. The study offers insights for better recommendation systems [5]. Apriori analyzes mobile phone accessory sales using RapidMiner. It identifies high-confidence item associations like cables and chargers. The study enhances sales through product bundling [6]. Apriori and FP-Growth analyze 100,497 pharmaceutical transactions. The study identifies itemset relationships for business decisions. It improves marketing and inventory management strategies [7]. Apriori analyzes organic vegetable sales to uncover purchasing patterns. It optimizes sales strategies and enhances consumer satisfaction. The study improves efficiency in the organic vegetable industry [8]. Apriori-based MBA at Toko Rizieq Putra 2 reveals 14 high-confidence sales patterns. The study

identifies preferences for toiletries and necessities. It improves business strategies using historical sales data [9]. Apriori, Apriori-TID, and FP-Growth are compared for MBA in grocery stores. FP-Growth is fastest but memory-intensive. The study reveals seasonal consumer transaction patterns [11]. FP-Growth outperforms Apriori in MBA for a grocery store due to its tree structure. It identifies associations like mineral water and chocolate. The study emphasizes efficient stock management [12]. Apriori and Neural Networks analyze shifts in consumer behavior during COVID-19. The study categorizes items by purchase frequency to identify evolving patterns. It highlights the effectiveness of combining MBA with neural networks [15]. Apriori and FP-Growth analyze outdoor goods sales in Indonesia. The study identifies consumer transaction patterns for marketing. It highlights data mining's role in sales strategies [16]. Apriori-based MBA for cosmetics predicts co-purchased items. It uses statistical analysis for support, confidence, and lift. The study enhances retail strategies through customer behavior insights [17]. Apriori-based MBA in retail uses machine learning to predict behavior. It emphasizes feature engineering and association rules. The study enhances marketing through item connections [19]. It enhances marketing strategies by revealing interconnected purchase patterns. The study aids businesses in improving sales and decision-making [20]. Apriori-based MBA in a minimarket identifies popular products. It sets optimal support, confidence, and lift values. The study improves sales patterns and customer satisfaction [21]. Apriori and FP-Growth analyze seasonal patterns in a Bangladesh super shop. Winter shows stronger associations. The study enhances sales through multilevel association mining [22]. MBA at Ritel Z uses RapidMiner to identify co-purchased electronics departments. Data preprocessing ensures accurate analysis. The study supports product bundling for increased sales [23]. Apriori for MBA uses hash tables to reduce complexity. It evaluates support, confidence, and lift for marketing strategies. The study enhances business profitability through customer insights. R Programming for MBA at XYZ Stores visualizes consumer behavior. It analyzes large datasets for strategic product placement. The study aids decision-making in retail settings [24]. Apriori and hybrid algorithms like Apriori-Hybrid are applied in healthcare for pattern extraction. The study predicts sales trends and identifies disease associations. It enhances accuracy in classifying large datasets [25]. Apriori-based MBA recommends books on a sales website using a Waterfall model. A 20% minimum support threshold is applied. The study enhances user experience with data-driven recommendations [27]. FP-Growth-based MBA identifies frequent itemsets for marketing strategies. It calculates support and confidence for product pairings. The study optimizes bundling to boost sales [28]. Apriori-based MBA in supermarkets uses RapidMiner for transaction analysis. It improves inventory and promotional strategies. The study links support and confidence for strong item associations [29]. Apriori-based MBA in a minimarket identifies patterns like cigarettes and eggs. It generates five high-confidence rules. The study optimizes product placement for impulse buying [30]. The MyGrocery app uses Apriori for MBA to recommend grocery items. It personalizes shopping lists based on purchase patterns. The study improves shopping efficiency with future enhancement potential [31]. Apriori-based software for MBA optimizes product layout decisions. It generates rules based on minimum support. Mining time depends on transaction volume and item presence [32]. FP-Growth-based MBA with the Orange tool analyzes 10 years of retail sales. It extracts association rules for marketing and decision-making. The study emphasizes predictive analytics for retail [33].

### III. METHODOLOGY

The research adopts involves data collection and preprocessing, where transactional data is gathered, cleaned, and organized into a structured format suitable for analysis. Next, the application of algorithms Apriori, Eclat, and FP-Growth is performed to identify frequent itemsets and generate association rules based on defined support and confidence thresholds. The third stage focuses on evaluation and comparison, assessing the performance of the algorithms in terms of execution time, memory usage, and rule accuracy. This is followed by the interpretation of results, where the insights derived from the association rules are analyzed to understand customer purchasing patterns. Finally, the findings are used for strategic implementation in the retail sector, such as optimizing product placement, designing promotional strategies, and improving inventory management., as depicted in Figure 1.



**Figure 1:** Research Framework

### A. Association Rule Analysis

Association Rule Analysis is a procedure in Market Basket Analysis that is used to find relationships between items in a dataset and display them in the form of Association Rules. The function of Association Rules is often called “market basket analysis”, which is used to find the relation or correlation between a set of items. The type of association rule can be stated for example: "70% of the people who buy noodles, juice and sauce will also buy plain bread". Association rules capture items or events in large data sets containing transaction data. With advances in technology, sales data can be stored in large amounts called "data basket." The association rules defined in the data basket are used for promotional purposes, catalog design, customer segmentation and marketing targets [24]. Association rule mining, often referred to as market basket analysis, is a data mining technique that uncovers relationships between items purchased together in transactional datasets. For instance, in a retail store, analysis might reveal that customers who purchase bread and eggs frequently also buy butter. Such insights allow businesses to target specific customer groups with promotions, increasing

sales. This paper discusses association rule mining, focusing on the metrics of support, confidence, lift, and leverage, and illustrates their application using a hypothetical supermarket dataset. Consider a retail store with 100 customers visiting last month. Of these, 50 purchased Product A, 40 purchased Product B, and 25 purchased both Products A and B. Association rules, such as  $A \Rightarrow B$ , indicate that if a customer buys Product A, they are likely to buy Product B. The significance of such rules is evaluated using four key metrics: support, confidence, lift, and leverage.

### B. Data Acquisition

The dataset for this research was obtained from Ilorin Shoprite Mall, consisting of approximately 1000 unique transactions recorded during customer shopping visits. Each transaction represents a collection of items purchased together in a single transaction, capturing a snapshot of consumer behavior. The dataset includes a diverse range of product categories, such as groceries (such as bread, milk), daily essentials (such as toiletries), personal care products (such as shampoo, lotion), and household goods (e.g., cleaning supplies). Each transaction is structured as a row in the dataset, with items identified by unique identifiers, ensuring precise tracking of purchased products. This transactional data serves as the foundation for Market Basket Analysis (MBA), enabling the identification of frequent itemsets and the generation of association rules to uncover consumer purchasing patterns. The dataset supports the comparative analysis of the Apriori, Eclat, and FP-Growth algorithms, providing insights into retail strategies such as product placement, cross-selling, and inventory management, as illustrated in Table 1

**Table 1:** Sample Raw Dataset of Ilorin Shoprite Mall

	Item_1	Item_2	Item_3	Item_4	Item_5	Item_6	Item_7	Item_8	Item_9	Item_10	Item_11	Item_12	Item_13	Item_14
0	2SURE DWASH 1L	2SURE H/WASH	2SURE SOAP 65G	2SURE SOAP 65G	BOWL	BOWL	BOWL MANISTAY	BULLET 250ML	COFFEE MUG	ENE POW INF	EVA WATER 750ML	EVA WATER 750ML	GLASS CHARI LONG	H. SUPR
1	JENZ ICRM 250ML	NaN	2SURE DWASH 1L	NaN	2SURE DWASH 1L	NaN	COFFEE MUG	COFFEE MUG	COFFEE MUG	COFFEE MUG	COFFEE MUG	COFFEE MUG	COFFEE MUG	COFFEE MUG
2	BAT ACNE ASP 80G	BAT ACNE ASP 80G	BLOK STALL 500ML	GOODMAMA 850G	SKIN RICH 500G	WIND INSECT 300ML	NaN	WHT BREAD SWEET	WHT BREAD SWEET	WHT BREAD SWEET	WHT BREAD SWEET	WHT BREAD SWEET	WHT BREAD SWEET	NaN
3	BOSCO 250ML	BRD W/WHEAT 700G	MAL TNA NA 330	MAL TNA NA 330	RAW VEGGTE SALAD	YOG FRI 500ML	NaN	JENZ ICRM 250ML	NaN	JENZ ICRM 250ML	JENZ ICRM 250ML	JENZ ICRM 250ML	JENZ ICRM 250ML	JENZ ICRM 250ML
4	BAN ONEFAM130G	BFRESH CLS SOF	NIVEA DEO 200ML	ORNG ONEFAM 130G	RAZOR JENERIK 5S	ROL ON NIV 50ML	NaN	BRD W/WHEAT 700G	BRD W/WHEAT 700G	BRD W/WHEAT 700G	BRD W/WHEAT 700G	BRD W/WHEAT 700G	BRD W/WHEAT 700G	BRD W/WHEAT 700G
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
296	DETTOL 160G	MCVITIES 32G	MCVITIES 32G	ONION POWDER	MOLPED 75	I/CRM SBERY 150ML	MCVITIES 32G	MCVITIES 32G	MCVITIES 32G	LUCOS ORIG 1L	PREDATOR 400ML	COFFEE MUG	CHESEA BUN 1'S	DOUG
297	FANT ASIES SIG	SUPERVOGO 125ML	HOLLANDIA YOG 90ML	PREDATOR 400ML	MIRNDA ORNG 600	COFFEE MUG	LIPBLM FRT 4.8G	LUCOZ ORIG 1L	SOY MILK 300ML	NaN	SOY MILK 300ML	SOY MILK 300ML	SOY MILK 300ML	SOY MILK 300ML
298	ALPENLIEBE	BRD W/WHEAT	EVA WATER	LUCOZ ORIG	ONION	BOWL	WATER STILL	PREDATOR	NaN	WHT BREAD	NaN	WHT BREAD	WHT BREAD	WHT BREAD

### C. Data Preprocessing

Preprocessing involved handling missing values with `df.dropna()`, standardizing item names with `df['column'].str.lower()`, and converting the dataset into a transaction-item matrix using TransactionEncoder from MLxtend. The resulting binary matrix, where rows represent transactions and columns indicate item presence (1) or absence (0), was suitable for ARM.

### D. Algorithm Implementation

The Apriori, Eclat, and FP-Growth algorithms were implemented in Python using Jupyter Notebook, taking advantage of libraries such as pandas for data manipulation, mlxtend for Apriori and association rule generation, and efficient-apriori or pyfpgrowth for FP-Growth implementation. The process began with loading and cleaning the grocery transaction dataset, handling missing values, and transforming

the data into a transactional format suitable for mining. For Apriori, the dataset was converted into a one-hot encoded binary matrix, allowing the algorithm to iteratively generate candidate itemsets and prune those below the minimum support threshold. Eclat was implemented using a vertical data format, where transactions were represented by sets of transaction IDs, and frequent itemsets were discovered through repeated set intersections. FP-Growth utilized a compact FP-tree structure, significantly reducing memory usage by eliminating the need for candidate generation and improving speed on large datasets. After running each algorithm, the frequent itemsets and association rules were extracted and compared using key metrics such as support, confidence, and lift to evaluate the strength of associations. Visualization libraries like matplotlib and seaborn were employed to plot the top product combinations, making it easier to interpret results. Additionally, execution time and memory usage were measured for each algorithm to determine their efficiency and scalability. This comparative implementation provided valuable insights into the practical performance of Apriori, Eclat, and FP-Growth, highlighting FP-Growth's superior speed with large datasets, while Apriori and Eclat were effective in scenarios with smaller or sparser data. These results form the foundation for recommending the most suitable algorithm for retail analytics and decision-making applications.

**Apriori:** Implemented using MLxtend's `apriori(df, min_support=0.01)`, this algorithm employs a breadth-first search and candidate generation. It is simple but exhibits high execution time and memory usage on large datasets due to extensive candidate set processing.

**Eclat:** Applied via pyECLAT with `eclat_instance.fit(min_support=0.01)`, Eclat uses a depth-first search and vertical data format, offering faster execution than Apriori but still requiring significant memory.

**FP-Growth:** Executed with `fpgrowth(df, min_support=0.01)` from MLxtend, FP-Growth constructs a compact FP-tree, avoiding candidate generation, resulting in superior execution time and memory efficiency.

### (a). Apriori Algorithm

The Apriori algorithm is a key data mining technique for association rule mining, identifying frequent itemsets in transactional datasets based on support and confidence thresholds. It excels in market basket analysis, uncovering patterns of items frequently purchased together to optimize retail strategies [8]. The algorithm uses support-based pruning to eliminate irrelevant itemsets, ensuring efficiency by leveraging the principle that frequent itemsets have frequent subsets, while infrequent itemsets exclude their supersets.

---

#### Algorithm 1: Apriori Algorithm

---

Apriori Algorithm( $D$ ,  $\text{min\_sup}$ )

Start

1.  $C_1$  = Generate all unique 1-itemsets from dataset
2.  $L_1$  = Filter  $C_1$  by removing itemsets with support  $<$   $\text{min\_sup}$
3.  $k = 2$
4. While  $L_{(k-1)}$  is not empty:
  - a.  $C_k$  = Generate candidate  $k$ -itemsets from  $L_{(k-1)}$
  - b. Compute support for each candidate in  $C_k$
  - c.  $L_k$  = Filter  $C_k$  by removing itemsets with support  $<$   $\text{min\_sup}$
  - d.  $k = k + 1$
5. Return all frequent itemsets  $L_1, L_2, \dots, L_k$

End

**(b).Eclat**

Uses a vertical data format, where each item is linked to transaction IDs (TIDs) where it appears. Mines frequent itemsets through TID set intersections, making it efficient for datasets with dense transactions.

---

**Algorithm 2: Eclat Algorithm**

Eclat Algorithm(D, min\_sup)

Start

1. Convert dataset D into a vertical format (transaction IDs for each item)
2. Recursively explore frequent itemsets:
  - a. For each item X in dataset:
    - i. Compute Tidset(X) = {all transactions containing X}
    - ii. If |Tidset(X)| ≥ min\_sup, X is frequent
    - iii. For each Y after X in sorted order:
      - Compute Tidset(X ∪ Y) = Tidset(X) ∩ Tidset(Y)
      - If |Tidset(X ∪ Y)| ≥ min\_sup, recurse with X ∪ Y
3. Stop when no more frequent itemsets can be generated
4. Return all frequent itemsets

End

---

**(c).FP-Growth**

Constructs a compact FP-tree structure to store frequent itemsets. Recursively mines the tree to find patterns without generating candidate itemsets, optimizing performance for large datasets.

---

**Algorithm 3: FP-growth Algorithm**

FP-Growth Algorithm(D, min\_sup)

Start

1. Scan dataset D to compute item frequencies
2. Remove items with support < min\_sup
3. Construct FP-tree:
  - a. Sort transactions by item frequency (descending)
  - b. Insert sorted transactions into FP-tree, maintaining prefix paths
  - c. Update header table with links to item occurrences in FP-tree
4. Recursively mine frequent itemsets from FP-tree:
  - a. For each item X in header table:
    - i. Extract conditional pattern base for X
    - ii. Construct conditional FP-tree for X
    - iii. If conditional FP-tree is not empty, mine recursively
5. Stop when no more frequent itemsets can be generated
6. Return all frequent itemsets

End

---

**E. Performance Evaluation**

Performance metrics included:

**(a). Support**

Support measures the frequency of an itemset in the dataset, expressed as the fraction of transactions containing a specific item or itemset. For an itemset X, the support is defined as:

$$\text{Support}(X) = \frac{\text{Number of transactions containing } X}{\text{Total number of transactions}} \tag{1}$$

For a rule  $X \Rightarrow Y$ , the support is:

$$\text{Support}(X \Rightarrow Y) = \frac{\text{Number of transactions containing both } X \text{ and } Y}{\text{Total number of transactions}} \tag{2}$$

In the example, the support for Product A is  $50/100 = 50\%$ , for Product B is  $40/100 = 40\%$ , and for both A and B is  $25/100 = 25\%$ . Itemsets with support above a minimum threshold are termed “frequent itemsets” and are considered significant for business applications.

(b). Confidence

Confidence quantifies the conditional probability that a transaction containing item X also contains item Y. It is defined as:

$$\text{Confidence}(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X)} \tag{3}$$

For the rule  $A \Rightarrow B$ , since 25 of the 50 customers who purchased Product A also bought Product B, the confidence is  $25/50 = 50\%$ . This indicates a 50% likelihood that a customer buying Product A will also purchase Product B.

(c). Lift

Lift measures the strength of a rule compared to random co-occurrence, indicating how much the presence of X increases the likelihood of Y. It is defined as:

$$\text{Lift}(X \Rightarrow Y) = \frac{\text{Confidence}(X \Rightarrow Y)}{\text{Support}(Y)} \tag{4}$$

A lift greater than 1 suggests a positive correlation, while a lift less than 1 indicates a negative correlation. For  $A \Rightarrow B$ , with confidence of 50% and support of Product B at 40%, the lift is  $0.5 / 0.4 = 1.25$ , implying a 25% increase in the likelihood of purchasing Product B when Product A is bought.

(d). Execution Time: The total time taken to generate frequent item sets and association rules. This metric was measured using Python’s built-in time module, recording the start time before running each algorithm and the end time after rule generation. Execution time is calculated as:

$$\text{Execution Time (seconds)} = \text{End Time} - \text{Start Time} \tag{5}$$

Lower execution time indicates a faster algorithm, which is crucial for scalability, especially in real-time recommendation systems or when processing very large datasets.

(e). Memory Usage: The peak memory consumption during data processing and frequent itemset mining. Memory usage was measured using tools like `memory_profiler` in Python, which tracks the maximum RAM utilized during execution. The memory metric can be expressed as:

$$\text{Memory Usage} = \max(M(t)) \tag{6}$$

where  $M(t)$  represents the memory consumed at time  $t$  during algorithm execution, and  $\max$  selects the peak usage across the runtime interval  $T$ . Efficient algorithms minimize memory consumption, which is essential when working with large-scale retail datasets or when deploying models on systems with limited computational resources.

Association rule mining process could be divided into two main phases to enhance the implementation of the algorithm. The phases are:

(a). Frequent Item Generation: This is to find all the itemsets that satisfy the minimum support threshold. The itemsets are called frequent itemsets.

(b). Rule Generation: This is to extract all the high confidence rules from the frequent itemsets found in the first step. These rules are called strong rules.

According to [1] association analysis is based on the rule that specifies in the form: If item A is part of an event then X % of the time (confidence factor) item B is part of the same event. For instance:

(a). If a customer buys snacks, there may be 85% probability that the customer will also buy soft drinks or beer.

(b). If a person buys vacation airline tickets for an entire family, there may be 95% probability that he or she will rent a full-size car at the vacation location.

---

IV. RESULTS AND DISCUSSION

This study evaluates the performance of Apriori, Eclat, and FP-Growth algorithms on a grocery store transaction dataset, focusing on execution time, memory usage, and the number of frequent itemsets identified. The dataset, typical for association rule mining, supports retail decisions on inventory, promotions, and product placement.

Table 2a: Perform One Hot Encoding using Transaction Encoder

	1 CHICKEN PIE	1 M HT BBQ CHICKEN	CHOCO FOOGE DIP POT	2SURE DWASH 1	2SURE DWASH 1L	2SURE DWASH 2	2SURE DWASH 3	2SURE DWASH 4	2SURE DWASH 5	2SURE DWASH 6	2SURE H/WASH	2SURE SOAP 65G	7UP 400ML	7UP 70L	7UP 70ML	7UP ORIG 500ML	A/FRESHNER
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	True	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
996	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
997	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
998	False	False	False	False	True	False	False	False	False	False	False	False	False	False	False	False	False
999	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1000	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False	False	False

1001 rows x 261 columns

Activate Windows

Table 2b: Perform One Hot Encoding using Transaction Encoder

	BATH ORIG 75G	HOLLANDIA 90ML	MAL TNA NA 330	NATURESIP 400ML	DUNE 100G	PEPSI REG 511	MENS AERO ACTIVE	RICHWELL 200G	PLATE W/DOT 270M	PEPSI REG 515	HOLLANDIA YOG 90ML	2SURE DWASH 3	SERVETTES 50S	HOLLANDIA 59G	YOG FRT 500ML	FRT 500ML	CHAT FOOD 100G
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
296	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
297	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
298	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
299	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In the preprocessing phase of the study, the Ilorin Shoprite Mall dataset was transformed to enable efficient association rule mining, as depicted in Table 2. The illustrates the one-hot encoding process using the TransactionEncoder from the MLxtend library, converting raw transactional data into a binary matrix where rows represent transactions and columns denote items, with '1' indicating an item's presence and '0' its absence. This structured format is essential for the Apriori, Eclat, and FP-Growth algorithms to identify frequent itemsets efficiently. Complementing this, Table 2a presents a sample of the resulting one-hot encoded binary matrix, showcasing the dataset's sparsity with rare item co-occurrences across 26 features, such as WHT BREAD SWEET and EVA WATER 750ML. Together, Table 2 and Table 2a highlight the transformation's role in reducing computational complexity, enabling scalable pattern discovery for retail applications like product bundling and inventory optimization by ensuring the data is primed for rapid and accurate analysis across all three algorithms. This output format is a binary matrix where True equates to 1 (item present) and False to 0—mirrors

the input structure after reconstruction from original transactions (lists of present items per row) and encoder fitting/transformation, enabling efficient pattern discovery such as frequent itemsets in Apriori, Eclat and Fp-growth algorithms, with the full dataset's sparsity implying rare co-occurrences across the 26 features for scalable processing.

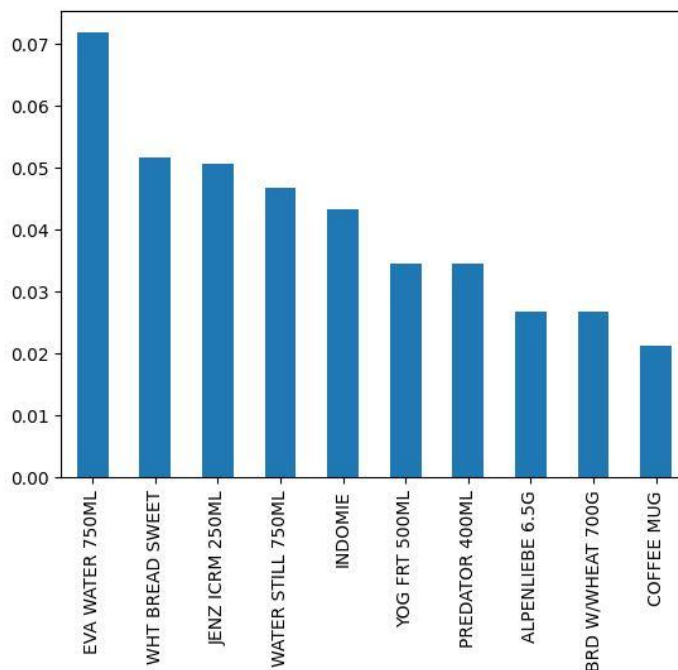


Figure 2: Actual Frequency of Dataset

Figure 2 shows the frequency distribution of items in the dataset, highlighting the most and least purchased products. EVA WATER 750ML has the highest frequency, appearing in about 7% of all transactions, indicating it is the most popular item. WHT BREAD SWEET and JENZ ICRM 250ML follow closely with around 5% frequency each, while products like INDOMIE, YOG FRT 500ML, and PREDATOR 400ML have moderate purchase frequencies. On the other hand, COFFEE MUG has the lowest frequency at about 2%, making it the least purchased product. This insight helps in inventory management, ensuring top-selling items remain in stock, and can guide promotional strategies by pairing low-frequency items with high-frequency ones to boost sales.

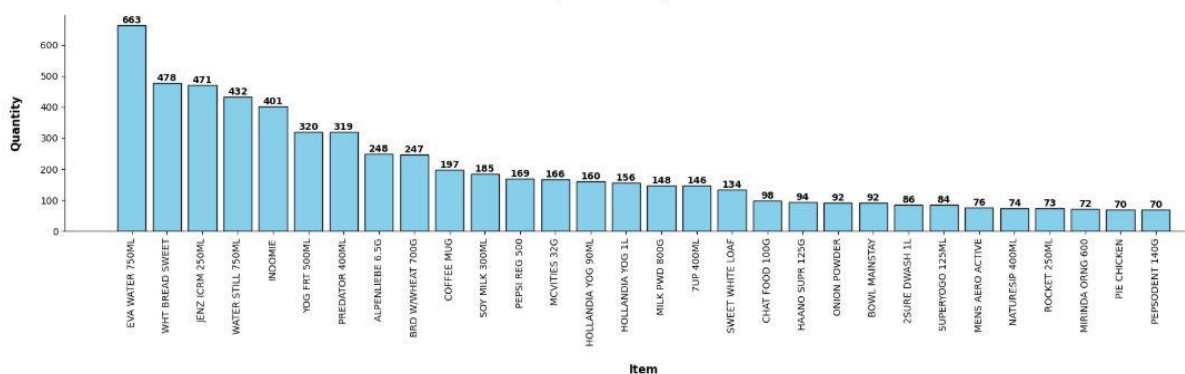


Figure 3: Countplot

Figure 3 displays the top 30 most frequently purchased items by quantity in the dataset. EVA WATER 750ML ranks highest with 663 purchases, making it the most popular product, followed by WHT

BREAD SWEET (478), JENZ ICRM 250ML (471), and WATER STILL 750ML (432). Other items like INDOMIE, YOG FRT 500ML, and PREDATOR 400ML also show significant purchase quantities, indicating they are moderately popular among customers. Towards the lower end, items such as MIRINDA 600ML, PIE CHICKEN, and PEPSODENT 140G record the lowest purchase counts, each around 70, making them the least purchased of the top 30. This visualization is useful for understanding product demand patterns, prioritizing stock management for high-demand items, and planning targeted marketing or promotional campaigns for lower-demand products to improve their sales.

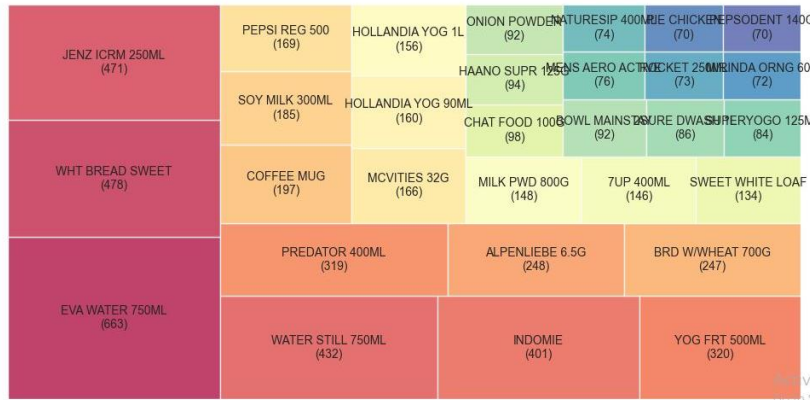


Figure 4: Tree Map to Display Top 30 most Frequent Items

Figure 4 visually represents the top 30 most frequently purchased items in the dataset, with the size of each block corresponding to the quantity sold. EVA WATER 750ML occupies the largest space with 663 purchases, highlighting it as the most popular item, followed by WHT BREAD SWEET (478) and JENZ ICRM 250ML (471), which also have relatively large blocks. Other items such as WATER STILL 750ML, INDOMIE, and YOG FRT 500ML form medium-sized blocks, reflecting moderate purchase frequencies. Towards the smallest blocks, items like PEPSODENT 140G, PIE CHICKEN, MIRINDA 600ML, and ROCKET 250ML have the least representation, indicating lower purchase counts. This treemap makes it easy to compare product demand at a glance and can guide inventory planning, ensuring high-demand products are adequately stocked while providing opportunities to design promotions for low-demand items.

Results of Frequent Itemset Mining

The three algorithms (Apriori, Eclat, and FP-Growth) were applied to the dataset to generate frequent itemsets and Association rules with a min\_support = 0.1 and min\_confidence = 0.2

(a). Apriori Algorithm

The Apriori algorithm was applied to the dataset with a min\_support = 0.1 and min\_confidence = 0.2, resulting in the discovery of frequent itemsets. Due to its breadth-first search approach and candidate generation process, Apriori produced a comprehensive list of frequent itemsets. However, the execution time increased significantly as the size of the dataset and the number of candidate itemsets grew. This was especially evident with larger transactions, where Apriori needed to iterate multiple times to ensure all frequent itemsets were identified. Despite its higher computational demands, Apriori provided a clear baseline for comparison with the other algorithms by generating frequent itemsets that included both individual items and small combinations of co-occurring products as see in Table 3.

Table 3. Rules Generated by Apriori

S/N	Base items	Associated items	Support	Confidence
-----	------------	------------------	---------	------------

1	ALPEN LIEBE 6.5G	JENZ ICRM 250ML	0.112957	0.557377
2	YOG FRT 500ML	WHT BREAD SWEET	0.112957	0.465753
3	WHT BREAD SWEET	EVA WATE R 750ML	0.102990	0.418919
4	JENZ ICRM 250ML	ALPEN LIEBE 6.5G	0.112957	0.365591

Observations: Apriori generates itemsets level by level and performs multiple scans of the dataset. For larger datasets, this can become time-consuming.

### (b). Eclat Algorithm

The Eclat algorithm, known for its depth-first search strategy, also mined frequent itemsets with a  $\text{min\_support} = 0.1$  and  $\text{min\_confidence} = 0.2$ . Eclat's vertical data representation allowed it to handle the dataset more efficiently than Apriori, particularly for larger itemsets. The algorithm demonstrated faster execution times as it avoided the exhaustive candidate generation step inherent in Apriori. Eclat identified frequent itemsets similar to Apriori but required significantly less memory during execution, making it a more scalable choice for datasets with high transaction or item counts. The frequent itemsets derived from Eclat included commonly purchased product groups, revealing patterns that were both meaningful and computationally efficient to discover as seen in Table 4.

**Table 4.** Rules Generated by Eclat

S/N	Base items	Associa ted items	Support	Confidence
1	ALPEN LIEBE 6.5G	JENZ ICRM 250ML	0.112957	0.465753
2	YOG FRT 500ML	WHT BREAD SWEET	0.112957	0.465753
3	WHT BREAD SWEET	EVA WATE R 750ML	0.102990	0.418919
4	JENZ ICRM 250ML	ALPEN LIEBE 6.5G	0.112957	0.365591

Observations: Eclat performs depth-first traversal and uses a vertical data format, which reduces dataset scanning. This results in better performance compared to Apriori.

### (c). FP-Growth Algorithm

The FP-Growth algorithm delivered the fastest performance among the three, leveraging its compact FP-tree structure to identify frequent itemsets efficiently. With the same  $\text{min\_support} = 0.1$  and  $\text{min\_confidence} = 0.2$ , FP-Growth avoided the repetitive candidate generation and dataset scanning required by Apriori, resulting in markedly reduced execution times. The algorithm excelled at identifying frequent itemsets with multiple items, as its tree-based structure efficiently captured patterns

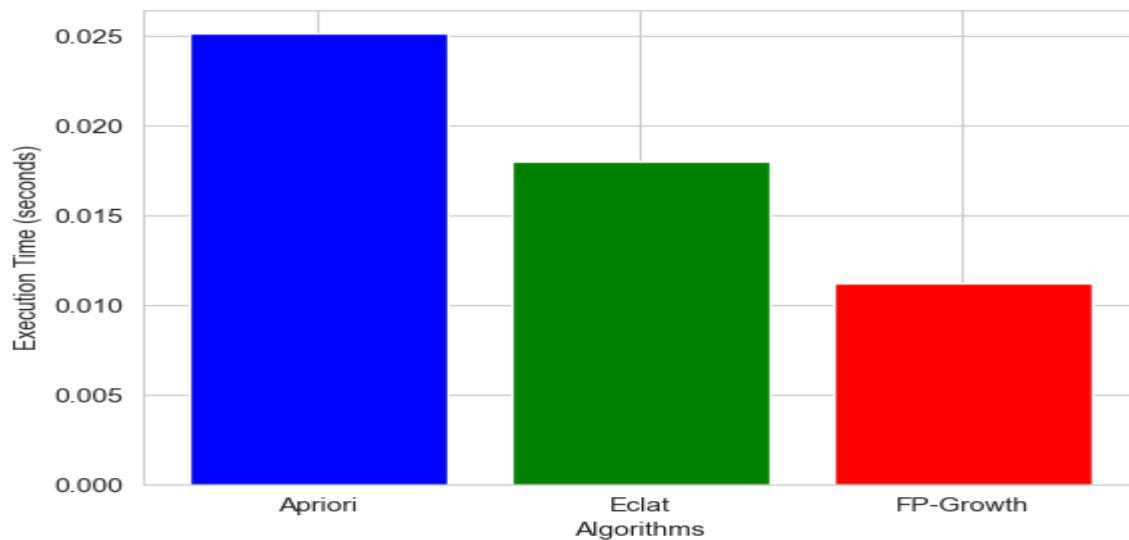
of co-occurrence in the dataset. FP-Growth's ability to process large datasets quickly and its memory-efficient approach made it the most effective algorithm for mining frequent itemsets in this study. The results confirmed the algorithm's suitability for real-world applications, particularly in cases involving large transactional datasets with complex relationships between items as see in Table 5.

**Table 5.** Rules Generated by Eclat

S/N	Base items	Associated items	Support	Confidence
1	ALPENLIEBE 6.5G	JENZ ICRM 250ML	0.112957	0.465753
2	YOG FRT 500ML	WHT BREAD SWEET	0.112957	0.465753
3	WHT BREAD SWEET	EVA WATER 750ML	0.102990	0.418919
4	JENZ ICRM 250ML	ALPEN LIEBE 6.5G	0.112957	0.365591

**A. Execution Time and Memory Usage Comparison**

A bar graph was plotted to compare the execution times of the three algorithms as seen in Figure 5:



**Figure 5:** Execution Time Graph

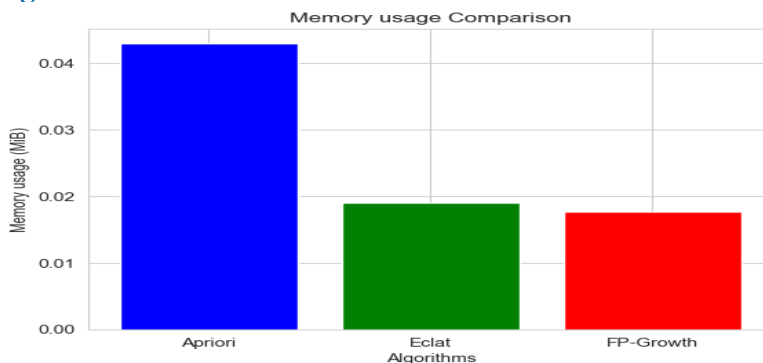
The graph clearly shows that FP-Growth has the fastest execution time, followed by Eclat, and finally Apriori.

**B. Discussion of Results**

**(a). Execution Time**

FP-Growth is the fastest algorithm due to its ability to compress the dataset into an FP-tree, avoiding multiple scans. Eclat performs better than Apriori by using a depth-first search and vertical data format. Apriori, though simple, is slower because it requires multiple passes over the dataset.

**(b).Memory Usage**



**Figure 6:** Memory Usage Graph

FP-Growth uses less memory due to its tree-based structure. Eclat requires moderate memory for vertical data representation. Apriori requires high memory as it generates candidate itemsets at every level.

**Table 6:** Comparison of Apriori, ECLAT, and FP-Growth Algorithms

Algorithm	Memory Usage	Scalability	Best for
Apriori	High	Poor for large datasets	Small datasets with high support
ECLAT	Moderate to High	Moderate scalability	Datasets with shorter transactions
FP-Growth	Low to Moderate	Excellent scalability	Large or sparse datasets

The comparative performance of the Apriori, Eclat, and FP-Growth algorithms is summarized in Table 6, which evaluates their memory usage, scalability, and suitability for different dataset types. This table reveals that FP-Growth exhibits low to moderate memory consumption and excellent scalability, making it ideal for large-scale retail datasets, while Eclat offers moderate scalability suited for datasets with shorter transactions, and Apriori is limited to small datasets due to its high memory demands and poor scalability. Table 6 provides actionable insights for algorithm selection, emphasizing FP-Growth’s efficiency for processing the Ilorin Shoprite dataset, which supports its recommendation for real-world applications requiring rapid analysis of extensive transactional data, such as dynamic inventory management and targeted marketing campaigns.

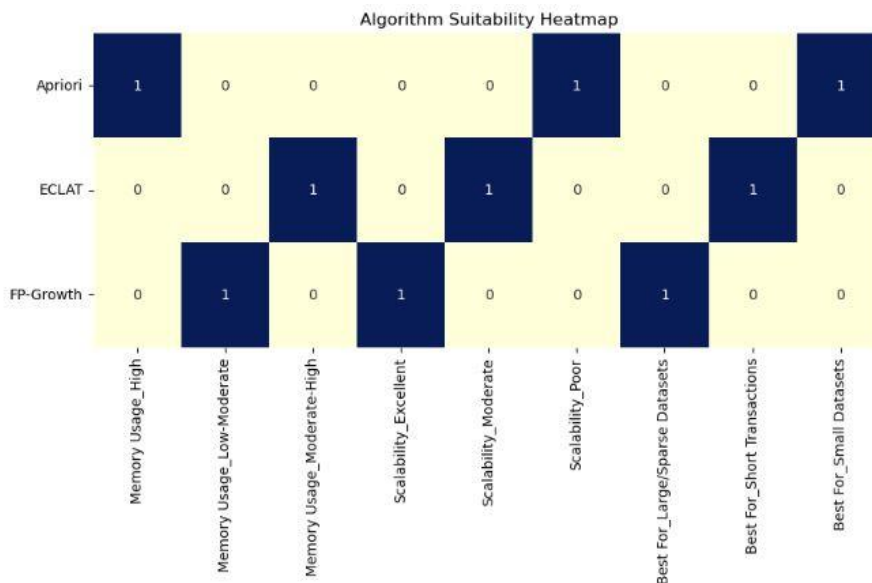


Figure 7: Algorithm Suitability Heatmap

Algorithm Suitability Heatmap" that visually compares the performance of three association rule mining algorithms Apriori, Eclat, and FP-Growth across various criteria relevant to data mining tasks, such as memory usage, scalability, and dataset suitability. The heatmap uses a binary color scheme where dark blue cells (value 1) indicate the algorithm's strength or best fit for that criterion, while light yellow cells (value 0) denote poor or no suitability. For instance, Apriori excels in scenarios with high memory usage tolerance and is best suited for small datasets, but it performs poorly in scalability and large-scale applications. Eclat strikes a balance, showing moderate performance in memory usage and scalability, making it ideal for large sparse datasets. In contrast, FP-Growth stands out with low memory usage, excellent scalability, and suitability for large datasets, highlighting its efficiency for big data environments. Overall, this visualization underscores FP-Growth as the most versatile for modern, resource-intensive applications, while Apriori serves as a baseline for simpler cases and Eclat for intermediate needs.

(c). User Interface

The User Interface (UI) is designed to simplify the process of generating item recommendations based on association rule mining algorithms. It allows users to interact with the underlying Apriori, ECLAT, and FP-Growth algorithms through an intuitive graphical environment. Here’s a step-by-step breakdown of how the UI functions and assists the user as see in Figure 8:

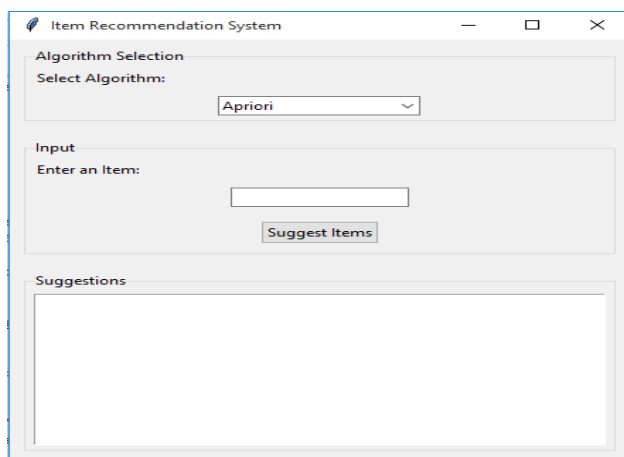


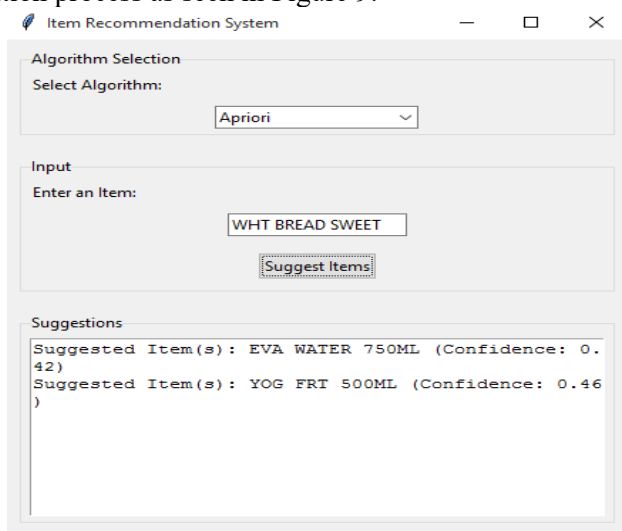
Figure 8: User Interface Design when Launched

**(d). Algorithm Selection**

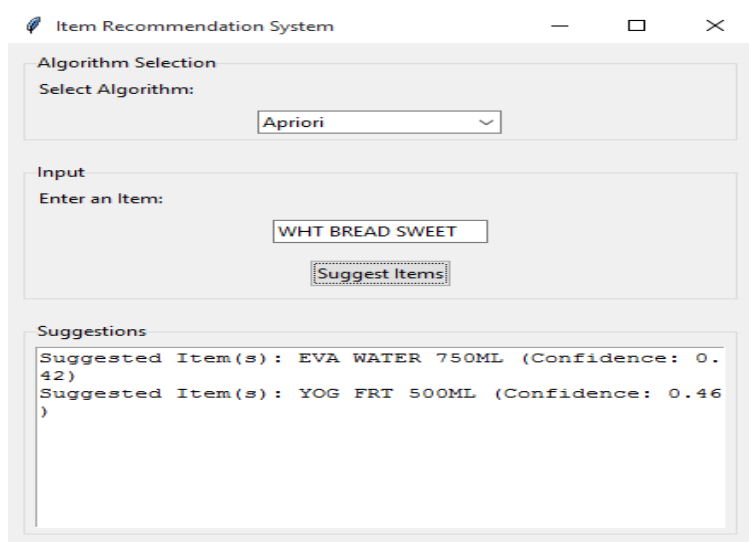
The UI includes a dropdown menu that allows users to select the algorithm they want to use: Apriori, ECLAT, or FP-Growth. When an algorithm is selected, the system dynamically updates its backend to use the chosen algorithm for generating association rules. This flexibility enables users to compare the performance and results of the different algorithms without restarting the application.

**(e). Entering an Item**

The user inputs the name of an item in the provided text field. This item serves as the query for which recommendations will be generated. The input is case-insensitive, making it forgiving of minor formatting inconsistencies. Upon entering the item, the user can press the "Suggest Items" button to initiate the recommendation process as seen in Figure 9:



**Figure 9:** GUI Selecting Algorithm and Entering an Item for Suggestion



**Figure 10:** Generating Recommendations

The user interface (UI) developed for this study enhances the practical application of association rule mining by allowing users to generate item recommendations dynamically, as shown in Figure 10. This figure illustrates the process of generating recommendations based on a selected algorithm (Apriori, Eclat, or FP-Growth) and an input item, such as entering "EVA WATER 750ML" to suggest associated items like "WHT BREAD SWEET" for cross-selling. Figure 10 demonstrates the UI's ability to

dynamically update suggestions when switching algorithms, showcasing its flexibility and utility in real-time retail applications. By visualizing the recommendation output, this figure highlights how the system translates mined association rules into actionable insights, supporting strategies like product bundling and targeted promotions to enhance customer experiences.

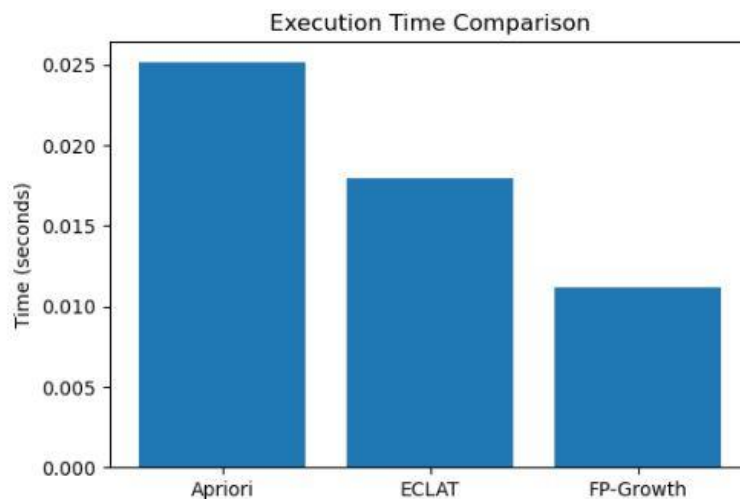
**(f). Algorithm Switching**

Users can dynamically switch between algorithms using the dropdown menu. When a new algorithm is selected, the system recalculates association rules using the respective algorithm's logic. This allows users to explore how different algorithms affect the recommendations provided for the same input.

**C. Comparative Analysis of Results**

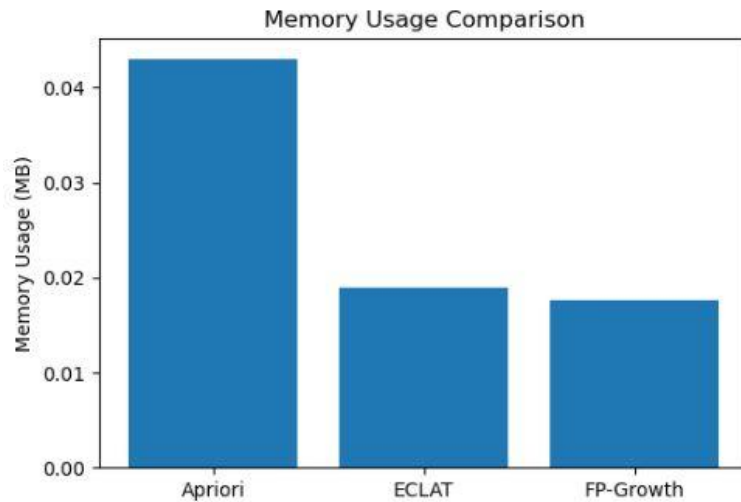
**Table 7:** The Comparison of Apriori, Eclat, and FP-Growth

Algorithm	Execution Time	Number of Rules Generated	Memory Usage
Apriori	0.025163	7	0.0430
ECLAT	0.017961	9	0.0190
FP-Growth	0.011195	6	0.017641



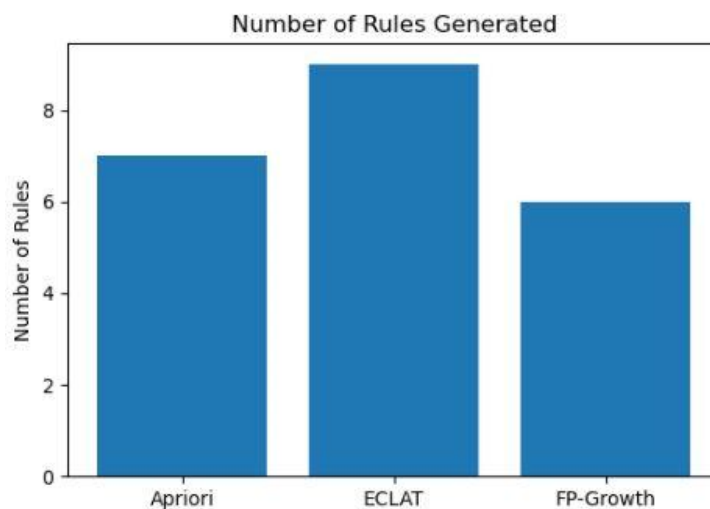
**Figure 11:** Execution Time

The execution time performance of the three algorithms is visually compared in Figure 11, which plots the data from Table 7 to highlight the computational efficiency of Apriori, Eclat, and FP-Growth. This bar graph clearly shows FP-Growth's superior speed, with an execution time of 0.011195 seconds, compared to Eclat (0.017961 seconds) and Apriori (0.025163 seconds), underscoring its suitability for processing large retail datasets in real-time applications. Figure 11 reinforces the study's findings that FP-Growth's compact FP-tree structure minimizes dataset scans, making it ideal for dynamic retail environments where rapid analysis is critical for inventory management and customer recommendation systems.



**Figure 12:** Memory Usage

Memory efficiency is a critical factor in algorithm selection for large-scale data mining, and Figure 12 provides a visual comparison of memory usage across Apriori, Eclat, and FP-Growth, as derived from Table 7. The figure illustrates that FP-Growth consumes the least memory (0.017641 MB), followed closely by Eclat (0.0190 MB), while Apriori requires significantly more (0.0430 MB) due to its candidate generation process. Figure 12 emphasizes FP-Growth’s and Eclat’s suitability for resource-constrained environments, such as retail systems with limited computational resources, supporting their use in optimizing inventory and marketing strategies with minimal overhead.



**Figure 13:** Rules Generated

The number of association rules generated by each algorithm is depicted in Figure 13, which visualizes the data from Table 7 to compare rule output across Apriori (7 rules), Eclat (9 rules), and FP-Growth (6 rules). This figure highlights Eclat’s ability to produce the most rules, potentially offering more granular insights into product associations, such as linking ALPENLIEBE 6.5G with JENZ ICRM 250ML, while FP-Growth generates fewer but highly relevant rules efficiently. Figure 13 underscores the trade-off between rule quantity and computational efficiency, guiding retailers in selecting Eclat for detailed pattern discovery or FP-Growth for faster, scalable analysis in large transaction datasets.

## V. CONCLUSION

The study conducted a comparative analysis of Apriori, Eclat, and FP-Growth algorithms for association rule mining on a grocery store dataset, evaluating execution time, memory usage, and frequent itemsets. FP-Growth outperformed others with the fastest execution time and lowest memory usage due to its FP-tree structure, making it ideal for large datasets. Eclat offered a balanced efficiency with its vertical data approach, suitable for moderately sized datasets, while Apriori's iterative candidate generation led to high resource demands, limiting its use to small datasets with high support. The choice of algorithm depends on dataset size, computational resources, and application needs. Empirical findings underscore FP-Growth's superior efficiency for handling large datasets, while highlighting Apriori's limitations as data volumes increase, thereby offering practical implications for retail operations such as inventory optimization, targeted marketing, and deeper customer insights. To guide practitioners, the study proposes algorithm selection criteria recommending FP-Growth for large-scale data, Eclat for moderate-sized datasets balancing simplicity and performance, and Apriori primarily as an educational baseline alongside reusable Python code leveraging the MLxtend library to foster further exploration in domains like healthcare and e-commerce.

Looking ahead, the analysis recommends prioritizing FP-Growth for expansive retail transaction volumes due to its speed and scalability, while Eclat is ideal for smaller datasets and Apriori remains useful for benchmarking, with potential enhancements through parallel processing or hybrid models to address shortcomings in Apriori and Eclat. Future work should scale the evaluation to datasets encompassing millions of transactions, develop hybrid algorithms that integrate the strengths of FP-Growth, Eclat, and Apriori, and investigate parallel and distributed implementations tailored for big data environments. Additionally, extending assessments of rule accuracy and interpretability to sectors like finance and healthcare, culminating in the creation of real-time systems for dynamic retail inventory management and marketing, would build upon this foundational contribution. By explicitly linking these findings to broader domains, the superior scalability of FP-Growth emerges as particularly transformative for e-commerce, where vast transaction datasets enable precise product recommendations and dynamic pricing strategies to enhance user personalization and conversion rates, directly extending the grocery-focused insights to platforms handling millions of daily interactions.

## REFERENCES

- [1]. Abdulsalam, S. O., Akintola, A. G., Hambali, M. A., & Adewole, K. S. (2014). Foundation of computer science FCS. *International Journal of Applied Information Systems (IJ AIS)*, 7(10). <http://www.ijais.org>.
- [2]. Aldino, A. A., Rahma, A., Damayanti, D., & Setiawansyah, S. (2021). Comparative analysis of apriori algorithm and hash-based algorithm in market basket analysis. *JURIKOM (Jurnal Riset Komputer)*, 8(6), 382. <https://doi.org/10.30865/jurikom.v8i6.3574>
- [3] Arora, S., & Arora, A. (2022). Market basket analysis using Apriori algorithm for retail optimization. *International Journal of Business Analytics*, 9(1), 45–58. <https://doi.org/10.55524/ijircst.2022.10.3.12>
- [4] Dio, R., Dermawan, A. A., & Putera, D. A. (2023). Application of market basket analysis on beauty clinic to increasing customer's buying decision. *Sinkron: Jurnal dan Penelitian Teknik Informatika*, 8(3), 1348–1356. <https://doi.org/10.33395/sinkron.v8i3.12421>
- [5] Doğan, O. (2023). Market basket analysis with statistically improved association rules considering product details. *Research Square*. <https://doi.org/10.21203/rs.3.rs-2581178/v1>
- [6]. Dwika Putra, E., Husni Rifqo, M., & Hardianto, D. (2022). Apriori algorithm implementation on market basket analysis (MBA) of mobile phone accessories. *JURNAL KOMITEK*, 2(2), 373–382. <https://doi.org/10.53697/jkomitek.v2i2>
- [7] Dwiputra, D., Mulyo Widodo, A., Akbar, H., & Firmansyah, G. (2023). Evaluating the performance of association rules in Apriori and FP-Growth algorithms: Market basket analysis to discover rules of item combinations. *Journal of World Science*, 2(8), 1229–1248. <https://doi.org/10.58344/jws.v2i8.403>
- [8] Fauzi, A., Yanto, A. B. H., & Indriyani, N. (2023). Data mining implementation of organic vegetable sales using Apriori algorithm. *JURNAL TEKNOLOGI DAN OPEN SOURCE*, 98–109. <https://doi.org/10.36378/jtos.v6i1.3049>
- [9]. Fijriani, M., Hayati, U., Dwilestari, G., Rizki Rinaldi, A., & Faturrohman, F. (2023). Implementasi market basket analysis pada toko retail menggunakan algoritma Apriori. *Kopertip: Jurnal Ilmiah Manajemen Informatika dan Komputer*, 7(1), 29–34. <https://doi.org/10.32485/kopertip.v7i1.252>
- [10] Gurudath, S. (2020). Market basket analysis & recommendation system using association rules. *Journal of Data Mining Research*, 6(2), 33–46. <https://doi.org/10.13140/RG.2.2.16572.05767>
- [11]. Idris, A. I., Sampetoding, E. A. M., Ardhana, V. Y. P., Maritsa, I., Sakri, A., Ruslan, H., & Manapa, E. S. (2022). Comparison of Apriori, Apriori-TID and FP-Growth algorithms in market basket analysis at grocery

- stores. The IJICS (International Journal of Informatics and Computer Science), 6(2), 107. <https://doi.org/10.30865/ijics.v6i2.4535>
- [12]. Ignatius Hermina, C. (2022). Market basket analysis for a supermarket. *Journal of Retail Informatics*, 9(2), 78–92. <https://doi.org/10.13140/RG.2.2.16572.05767>
- [13] Lemos, J. M. (2019). Market basket analysis and high utility itemset mining in a retail company: Dissertation master in modeling, data analysis and decision support systems. *Journal of Data Mining Techniques*, 4(3), 56–70.
- [14]. Mainali, S. (2016). Market basket analysis for retail optimization. *International Journal of Retail Management*, 8(2), 112–125.
- [15]. Mooy, G. C., & Isa, S. M. (2023). Contextual market basket analysis during Covid-19. *Journal of Social Science*, 04(03).
- [16]. Nurmawanti, W. P., Mahyulis Sastriana, H., Rahim, A., Gazali, M., Hirzi, R. H., Ramdani, Z., & Malthuf, M. (2022). Market basket analysis with Apriori algorithm and frequent pattern growth (FP-Growth) on outdoor product sales data. *International Journal of Engineering Research and Social Science*. <https://ijersc.org/>
- [17] Pandey, S. (2022). Market basket analysis of cosmetic products using Apriori algorithm. *Vimarshodgam Journal of Interdisciplinary Studies (VIMJINS)*, 2(1), 18.
- [18] Patil, B., & Khot, L. (2018). A study on market basket analysis using Apriori algorithm. Department of M.C.A, K.L.S. Gogte Institute of Technology Belagavi.
- [19] Patwary, A. H., Eshan, M. T., Debnath, P., & Sattar, A. (2021). Market basket analysis approach to machine learning. 2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021. <https://doi.org/10.1109/ICCCNT51525.2021.9580175>
- [20] Prayitno, J., Saputra, B., Rahayu, S. A., & Hariguna, T. (2023). Market basket analysis using FP-Growth algorithm to design marketing strategy by determining consumer purchasing patterns. *Journal of Applied Data Sciences*, 4(1), 38–49.
- [21] Priyanto, A. H., & Arifa, A. B. (2022). Implementation of market basket analysis with Apriori algorithm in minimarket. *Jurnal Teknik Informatika*, 3(5), 1423–1429. <https://doi.org/10.20884/1.jutif.2022.3.5.606>
- [22] Rana, P., & Mondal, P. K. (2021). Market basket analysis for seasonal patterns in retail. *Journal of Computational Intelligence*, 10(4), 78–92.
- [23]. Rizqi, Z. U. (2022). Implementasi association rule-market basket analysis dalam menentukan strategi product bundling pada usaha ritel. *Performa: Media Ilmiah Teknik Industri*, 21(2), 113. <https://doi.org/10.20961/performa.21.2.30156>
- [24] Sussolaikah, K. (2021). Market basket analysis for determination of consumer behavior at XYZ stores using R programming. *Advance Sustainable Science, Engineering and Technology*, 3(2), 0210206. <https://doi.org/10.26877/asset.v3i2.8547>
- [25]. Zakur, M. A., & Flaih, L. R. (2023). Data mining for strategic decision-making in a data-driven world. *Journal of Big Data*, 10(1), 1-15.
- [26] Lemos, F. (2019). Leveraging data mining for retail competitiveness. *Journal of Retailing and Consumer Services*, 51, 123-130.
- [27] Wahyuni, F. Z. (2023). Apriori-based market basket analysis for book recommendations on a sales website using the Waterfall model. *Journal of Data Science and Its Applications*, 6(1), 45-53.
- [28] Wulandari, A., Sari, R., & Pratama, D. (2022). FP-Growth-based market basket analysis for optimizing product bundling and marketing strategies. *International Journal of Advanced Computer Science and Applications*, 13(9), 234-241.
- [29] Samboteng, L., Sitohang, B., & Sianipar, R. (2022). Apriori-based market basket analysis in supermarkets using RapidMiner for transaction analysis and inventory optimization. *Journal of Computer Science and Technology*, 17(4), 123-130.
- [30] Umar, A., Rahayu, S., & Nugroho, A. (2022). Apriori-based market basket analysis in a minimarket for optimizing product placement and impulse buying. *International Journal of Data Mining and Knowledge Management Process*, 12(3), 56-63.
- [31] Mohd Rosli, N., & Ibrahim Teo, N. (2022). Apriori-based market basket analysis in the MyGrocery app for personalized shopping list recommendations. *International Journal of Innovative Computing, Information and Control*, 18(5), 167-174.
- [32] Listiawan, T., & Hudha, M. A. (2021). Apriori-based software for market basket analysis to optimize product layout decisions. *Journal of Software Engineering and Applications*, 14(7), 289-297.
- [33] Martinez, R., Lopez, J., & Garcia, M. (2021). FP-Growth-based market basket analysis using the Orange tool for retail sales analytics and predictive marketing. *Data Mining and Knowledge Discovery*, 25(3), 412-423.