National Research University Higher School of Economics
Faculty of Computer Science
Course – Master's in Data Science

MASTER'S THESIS

English Title:
Detecting Pattern in Purchase History: Given a dataset on order history, explore association rules learning methods for pattern recognition.

Russian Title:
Обнаружение закономерностей в истории покупок: учитывая набор данных в истории заказов, изучите правила ассоциаций, изучите методы распознавания закономерностей

Student Name: Siddharth Kumar Shukla
SE-2, Data Intelligence & Cloud Efficiency Team, PayPal

Supervisor Name: Maksimovskaya Anastasia Maksimovna
HSE Lecturer

July, 2023

ABSTRACT

The rapid growth of online shopping has generated magnanimous amount of purchase data. Taking Amazon alone, an average of 310 million customers orders daily. This generates a huge amount of data which provides an opportunity for researchers and business units to fetch valuable insights into customer behavior.

Pattern detection in a customer's historical data provides insights that can drive personalized marketing strategies, optimize inventory management, identify cross selling opportunities, and improve fraud detection. It helps the business to make data-driven decisions and deliver enhanced customer experiences.

The study explores various association rule mining techniques such as Apriori Algorithm & FP-growth algorithm. These algorithms analyze the dataset to generate rules that describe the relationship between two items. The research evaluates the performance of these algorithms in terms of the quality of the discovered patterns.

To conduct the analysis, a real-world dataset of purchase history of an electronics store is used. The dataset is pre-processed to handle missing values, remove outliers and other data quality issues.

The result of this study provides valuable insights into the effectiveness of different association rules. The results have significant implications for businesses which seek to optimize their marketing efforts and deliver personalized experiences to their customers which leads to better customer experience and increased revenue.

1. INTROCUCTION

1.1 What is Pattern Analysis

Pattern Analysis using users historical purchase data refers to a process of identifying and understanding patterns and association rules within data representing customers' purchasing behavior. It involves extracting useful information from the transaction dataset. As the world moves towards a digital era, both online and offline shopping will generate huge chunk of data which can be used to fetch useful information that helps business generate better customer experience.

Pattern Analysis aims to discover the relationship and dependencies among items such as frequently co-purchased items, sequential purchase patterns, association between customer and items, etc. Business can understand customer purchasing behavior and will help them to enhance the customer experience.

Pattern Analysis can help business to identify useful patterns which can be used for providing personalized recommendations to customers, better inventory management, run marketing campaigns & customer segmentations. This helps business improve their customer experience along with the increase in their revenues.

1.2 Why is Pattern Analysis needed?

Detecting Patterns in the customer historical purchase data is required for numerous reasons:

1. Understanding Customer Behavior:

Analyzing historical purchase data of customers helps business gain deep insights into customers behavior, purchasing habits & preferences. It helps in understanding what items the customer frequently purchases, what factors affect their buying decisions, etc. This understanding helps in improving customer targeting, developing effective marketing strategies, and giving customers a personalized experience.

2. Enhancing Marketing Strategies:

By understanding the customer purchase behavior, business can optimize their marketing efforts. They can create promotions based on the discovered association rules like creating bundled offers or recommend related products. This helps in maximizing the return on marketing investments.

3. Personalized Recommendations:

Pattern Analysis helps business to understand individual customers and classify their customers into different segmentations. These segmentations are created based upon the purchasing pattern of the customers. Customers staying in a specific region can be grouped

under one segment and suggestions can be made to this specific group based upon their common interests.

4. Inventory Management:

Analyzing patterns in customer purchase data helps businesses optimize the inventory management process. By identifying the frequently purchased products, identifying seasonal demand products business can make better forecast of their inventories. This leads to better stock management, reduced cost, and customer satisfaction.

5. Customer Segmentation:

Pattern analysis helps in segmenting customers based on their purchase patterns and behavior. By grouping customers with similar interests, business can create marketing campaigns, loyalty programs, etc. for each segment. Customer segmentation helps in understanding their diversity, identify high-valued customers and tailor their strategies to meet specific needs.

2. LITERATURE REVIEW

2.1 Structure of Work

This section provides a detailed analysis on the methodology to find patterns in the user purchase historical data. This study was divided into multiple steps which are listed below.

Data Collection:
To perform this study, a real-world dataset of purchase history of an electronics store is used. The dataset contains 2.6 million purchased products from online store.
The dataset contains data from April 2020 to November 2020 from a large home appliance and electronics online store.
The link to the data source is provided in the reference section.

Exploratory Data Analysis:
Exploratory Data Analysis (EDA) involves a systematic approach to understand and summarize the dataset.
Understanding data involves familiarizing yourself with dataset structure, variables, and data types. Post this, identify the target variables and check for missing values, outliers, and potential data quality issues.
Data Cleaning involves substituting missing values or completely removing the column if there are too many missing values. In the current scope, the missing values in categorical features are substituted with the mode.
Each variable was analyzed and its corresponding relationship with other variables using the correlation matrix.
For numerical features mean, mode and standard deviation is calculated and for categorical features mode is calculated.

Multivariate Analysis:
For investigating interactions & patterns involving multiple variables simultaneously, heatmaps technique has been utilized.

Data Modelling:
Data modelling is the process of finding the best model which minimizes the difference between the model's predictions and the observed data points. In this study, Apriori Algorithm and FP-Tree Algorithm have been used. The algorithm has been run for different parameter values and a comparison is made between the quality of these 2 algorithms.

2.2 Related Work

This research is inspired by previous research on detecting patterns in customers historical purchase data.

Most of the previous research mainly focuses on finding the customer behavior from the historical purchase data. This study has been done for both online and offline purchases. There has been some research work which tries to implement dynamic pricing based upon

the customer sentiments. Logistic Regression has been used for classifying customers into different segments and finding the dynamic price for each of the segment.
Few recent studies have tried to implement algorithms like content-based filtering, collaborative filtering, rule-based hybrid approaches for finding patterns in the historical purchase data.

Market Basket Analysis (MBA) is a useful algorithm which finds the relationship between 2 products that are frequently purchased. However, MBA strictly focusses on binary relationships which prevent it from examining the product relationships from a broader perspective. In the paper titled "*Discovering customer purchase patterns in product communities: An empirical study on co-purchase behavior in an online marketplace*", patterns are discovered using Stochastic Block Modelling (SBM) community detection technique. Examining the discovered communities, the products are segmented and labelled according to their roles in the network by calculating the network characteristics. SBM community detection method discovers communities based on connection patterns of the nodes. Connections represent customers' purchases, so SBM method groups the products based on their buyers' purchase patterns.
There are two different nodes namely buyers and products. An edge is drawn between a buyer and a product in this model if the buyer has purchased that product. One of the features of the SBM is that it can detect whether the network has a community structure or not.

One of the inspiring research projects involves Radio Frequency Identification (RFID) technique. This approach is useful for finding hot spots, dead spots, and path patterns of customer movements in a grocery store. Least common subsequence (LCS) is characterized by grouping main shopping path sequences in travel order, the most repeatedly appearing nodes amongst all sequences are regarded as hot spots and the rare ones are regarded as dead spots.

In paper with heading "*Recommendations based on purchase patterns*", purchase history of users is analyzed to find their purchase patterns related to user behavior.
Patterns are then used to predict the category of next possible purchase in a particular location. The current approach includes content-based and collaborative filtering. In this paper, historical data is used to understand purchase patterns and then location-based recommendations is applied. Given a set of category sequences, the problem is to find purchase patterns related to the user behavior that can be used to make recommendations. The classic approach is to use sequential pattern analysis where frequent occurring patterns are extracted from all subsequences of the given set of sequences. Each pattern is associated with a support value.

3. EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis (EDA) is a crucial part in data analysis that involves exploring the dataset and summarizing the main characteristics in the dataset. Its primary goal is to discover hidden insights and formulate initial hypotheses. EDA allows initial assessment of data quality. It is used to handle missing values, outliers, inconsistencies, and other data issues that may affect the result.

3.1 About our Data

The dataset contains 2.6 million purchased products from online store.
The dataset contains data from April 2020 to November 2020 from a large home appliance and electronics online store.
There are 2633521 rows and 8 columns in the dataset.

	event_time	order_id	product_id	category_id	category_code	brand	price	user_id
0	2020-04-24 11:50:39 UTC	2294359932054536986	1515966223509089906	2.268105e+18	electronics.tablet	samsung	162.01	1.515916e+18
1	2020-04-24 11:50:39 UTC	2294359932054536986	1515966223509089906	2.268105e+18	electronics.tablet	samsung	162.01	1.515916e+18
2	2020-04-24 14:37:43 UTC	2294444024058086220	2273948319057183658	2.268105e+18	electronics.audio.headphone	huawei	77.52	1.515916e+18
3	2020-04-24 14:37:43 UTC	2294444024058086220	2273948319057183658	2.268105e+18	electronics.audio.headphone	huawei	77.52	1.515916e+18
4	2020-04-24 19:16:21 UTC	2294584263154074236	2273948316817424439	2.268105e+18	NaN	karcher	217.57	1.515916e+18

Fig 1

The data type of the columns can be seen in Fig 2

```
event_time        object
order_id           int64
product_id         int64
category_id      float64
category_code     object
brand             object
price            float64
user_id          float64
dtype: object
```

Fig 2

On further analysis, it was observed that 16% or *category_id* values, 23% of *category_code* values, 19% of *brand_values*, 16% of price values and 78% of *user_id* values were missing.

To visualize the missing values, the below graph was plotted using heatmap tool.

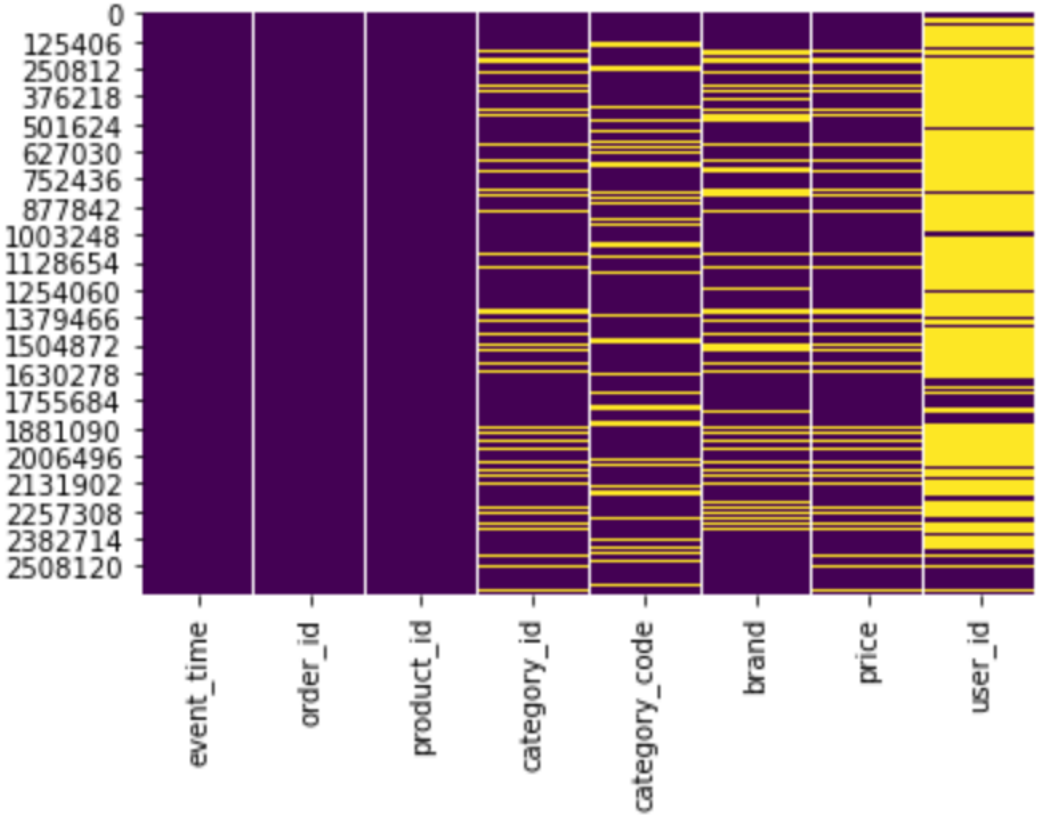

Fig 3

For the numerical features (*price*), the missing values were substituted with the mean. For the categorical features (*category_code*, *brand*), the missing values were substituted with the mode of the respective columns.

3.2 Correlation Analysis

Correlation analysis helps in identifying the relationships between variables. It determines if variables are positively or negatively related and how closely they are dependent on each other. It also assists in feature engineering.

In our study, we examined the correlation between *price*, *order_id*, *category_id*, and *product_id*.

The below pic depicts the correlation between different variables.

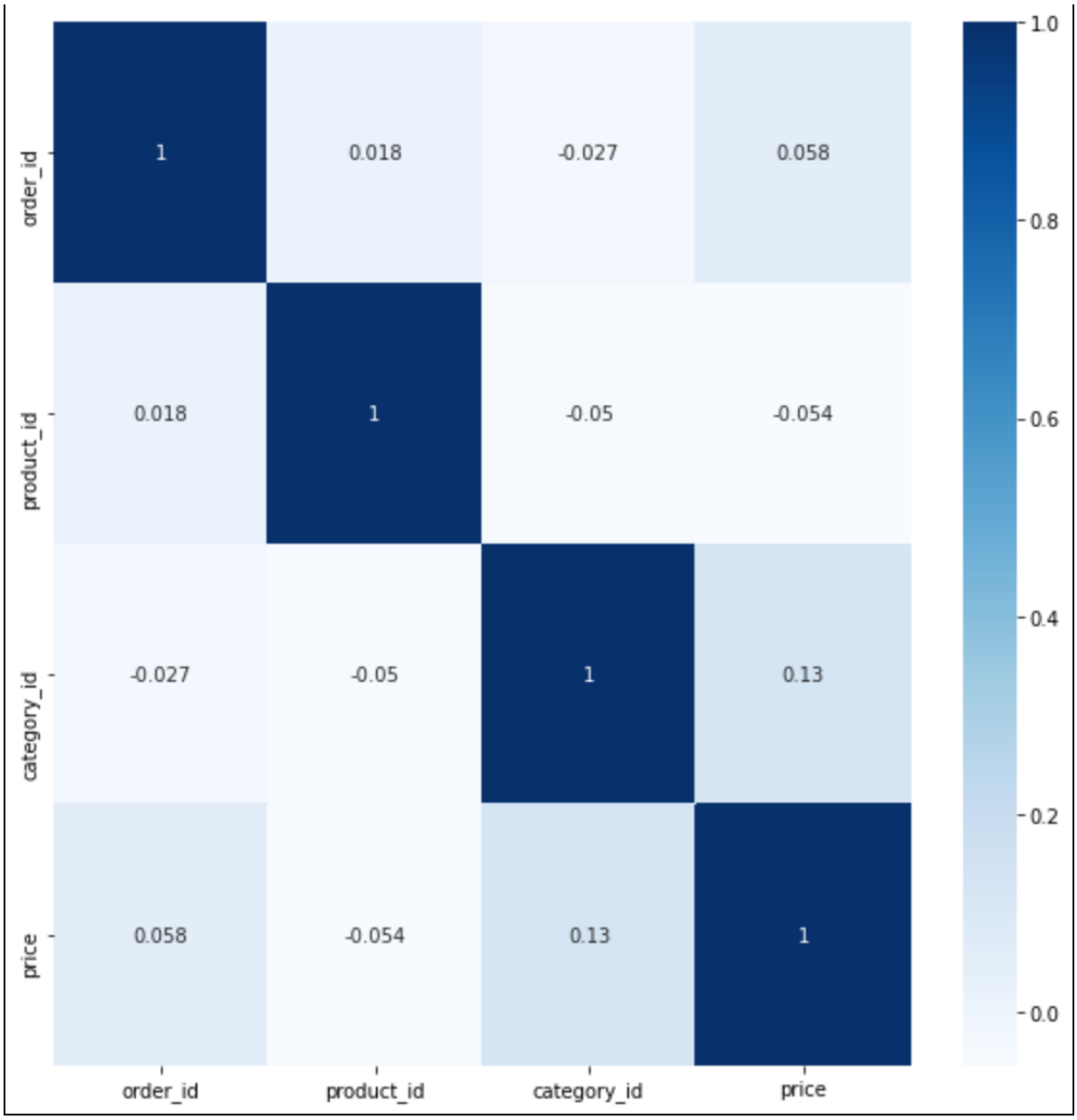

Fig 4

3.3 Results of EDA

After pre-processing of the dataset, we observed some interesting facts which are listed below

- The maximum value of order placed is 52295.243. In this order, 3 products were purchased out of which, 2 products belong to Samsung brand.
- The product which is most frequently bought lies close to the median price whereas the less frequent items are either too left or too right to the median.

- Every product is associated with only 1 *category_id*. Hence, there is a one-to-one mapping between *product_id* and *category_id*.
- The most frequently bought product generates the maximum revenue.
- Samsung is the brand which has maximum sales.

DATA MODELLING

In this study, we are going to use Apriori Algorithm and FP-Tree Algorithm for finding patterns in the customers historical purchase data.
Both algorithms are executed with different input parameters on the same dataset and a comparison is made.

4.1 Apriori Algorithm

Apriori algorithm is designed to discover frequent item-sets from large transactional dataset. It follows a breadth-first search approach.
The Apriori property states that if an itemset is frequent, then its subsets must also be frequent. Using this idea, we eliminate the candidate item-sets that contain infrequent subsets.

Apriori Algorithm works as follows:

1. Dataset is scanned to generate frequency of the item-sets. Items that don't meet the minimum support threshold are eliminated and we are left with frequent 1-item-sets.
2. Item-sets of length 2 are generated. Candidates are formed by taking union of two frequent 1-itemsets.
3. The frequency of 2-itemsets is calculated and item-sets which don't meet the minimum support threshold are pruned.
4. The above steps are repeated generating candidate item-sets of increasing length until no frequent item-sets can be generated.
5. After identification of all frequent item-sets, algorithm generates association rules from these item-sets.
6. Confidence of a rule is calculated by dividing support of rule's itemset by support of rule's antecedent. Rules that meet minimum confidence threshold are significant.

4.2 FP-Tree Algorithm

FP-Tree (Frequent Pattern Tree) Algorithm is used for mining frequent item-sets in large transactional datasets. It differs from Apriori algorithm by utilizing a tree-based data structures called FP-Tree which enables more efficient mining of frequent item-sets.

FP-Tree Algorithm works as follows:

1. FP-Tree consists of nodes representing item-sets, linked together to form item-sets based on the occurrence in transactions. Each node contains item name, count of its occurrence and links to its child nodes.
2. Remove infrequent items from the dataset. This results in lesser memory consumption and speeds up the algorithm.
3. Sort items in descending order based on their frequencies.
4. For each transaction, insert sorted items into the FP-Tree. If item already exists, increment its count otherwise create a new branch from rood node and add item as a child node.

5. Repeat the above steps for all transactions.
6. Starting from least frequent item, recursively mine frequent item-sets from the FP-Tree.
7. From frequent item-sets obtained, association rules can be generated. Based upon the minimum confidence threshold criteria, relevant rules are filtered out.

4.3 FP-Tree Vs Apriori Algorithm

Using compact FP-Tree data structure, the need for candidate generation can be avoided which reduces the number of scans of the dataset.
FP-Tree algorithm can handle sparse dataset more effectively compared to Apriori algorithm as it compresses the dataset and eliminates the need to generate candidate item-sets.

FP-Tree has shown better performance than Apriori Algorithm in the past researches. However, FP-Tree requires more memory due to construction and storage of FP-Tree data structure.

5. RESULT

In this study, we aimed at understanding the patterns from the customer transaction data.
For analyzing the patterns, we used two algorithms namely Apriori Algorithm and FP-Tree Algorithm. In this section we'll present the results obtained and compare the performance of these two algorithms.
We executed the algorithms with different set of parameters.

5.1 Apriori Algorithm

Apriori Algorithm is executed with different support values and confidence level and the results are attached below

Minimum Support Value = 0.06
Minimum Confidence Level = 0.1

```
Frequent itemsets:
frozenset({'electronics.audio.headphone', 'electronics.smartphone'})
frozenset({'appliances.kitchen.refrigerators', 'electronics.smartphone'})
frozenset({'computers.notebook', 'electronics.smartphone'})

Association rules:
frozenset({'electronics.audio.headphone'}) => frozenset({'electronics.smartphone'}) (Confid
ence: 0.33798827154358413)
frozenset({'appliances.kitchen.refrigerators'}) => frozenset({'electronics.smartphone'}) (C
onfidence: 0.3034166994879874)
frozenset({'computers.notebook'}) => frozenset({'electronics.smartphone'}) (Confidence: 0.2
8796114259280675)
```

Minimum Support Value = 0.5
Minimum Confidence Level = 0.5

```
Frequent itemsets:
frozenset({'electronics.smartphone', 'electronics.video.tv'})
frozenset({'electronics.audio.headphone', 'electronics.smartphone'})
frozenset({'appliances.kitchen.refrigerators', 'electronics.smartphone'})
frozenset({'appliances.environment.vacuum', 'electronics.smartphone'})
frozenset({'computers.notebook', 'electronics.smartphone'})

Association rules:
frozenset({'electronics.video.tv'}) => frozenset({'electronics.smartphone'}) (Confidence: 0
.32999320652173914)
frozenset({'electronics.audio.headphone'}) => frozenset({'electronics.smartphone'}) (Confid
ence: 0.33798827154358413)
frozenset({'appliances.kitchen.refrigerators'}) => frozenset({'electronics.smartphone'}) (C
onfidence: 0.3034166994879874)
frozenset({'appliances.environment.vacuum'}) => frozenset({'electronics.smartphone'}) (Conf
idence: 0.3167927004562215)
frozenset({'computers.notebook'}) => frozenset({'electronics.smartphone'}) (Confidence: 0.2
8796114259280675)
```

5.2 FP-Tree Algorithm

FP-Tree Algorithm is executed with different support values and confidence level and the results are attached below

Minimum Support Value = 0.5
Minimum Confidence Level = 0.5

support	itemsets
0.628472858276852	frozenset({'electronics.smartphone'})
0.130304695609697	frozenset({'computers.notebook'})
0.105228877897865	frozenset({'appliances.kitchen.refrigerators'})
0.104577557957298	frozenset({'electronics.audio.headphone'})
0.0880808450876229	frozenset({'electronics.video.tv'})
0.076021249313061	frozenset({'computers.notebook', 'electronics.smartphone'})
0.0760110724389896	frozenset({'appliances.environment.vacuum'})
0.0737823370173617	frozenset({'appliances.kitchen.washer'})
0.069212920559321	frozenset({'electronics.audio.headphone', 'electronics.smartphone'})
0.0650200484419206	frozenset({'computers.peripherals.mouse'})
0.0627200749017932	frozenset({'appliances.kitchen.refrigerators', 'electronics.smartphone'})
0.0593209989619589	frozenset({'electronics.video.tv', 'electronics.smartphone'})
0.0592090533471739	frozenset({'appliances.kitchen.kettle'})
0.0515865746677251	frozenset({'appliances.environment.vacuum', 'electronics.smartphone'})
0.0501211048014492	frozenset({'appliances.environment.air_conditioner'})

Minimum Support Value = 0.06
Minimum Confidence Level = 0.1

support	itemsets
0.0880808450876229	frozenset({'electronics.video.tv'})
0.628472858276852	frozenset({'electronics.smartphone'})
0.104577557957298	frozenset({'electronics.audio.headphone'})
0.105228877897865	frozenset({'appliances.kitchen.refrigerators'})
0.130304695609697	frozenset({'computers.notebook'})
0.0760110724389896	frozenset({'appliances.environment.vacuum'})
0.0737823370173617	frozenset({'appliances.kitchen.washer'})
0.0650200484419206	frozenset({'computers.peripherals.mouse'})
0.069212920559321	frozenset({'electronics.audio.headphone', 'electronics.smartphone'})
0.0627200749017932	frozenset({'appliances.kitchen.refrigerators', 'electronics.smartphone'})
0.076021249313061	frozenset({'computers.notebook', 'electronics.smartphone'})

5.3 Analyzing Association Rules

It can be observed that these items are bought frequently:

1. Headphone & Smartphone
2. Computer notebook & Smartphone
3. Television set & Smartphone
4. Refrigerator & Smartphone
5. Vacuum Cleaner & Smartphone

One conclusion that can be made here is that customers who come for buying expensive products such as Computer, Television set, Vacuum Cleaner, etc. also end up buying Smartphones. The reason behind this can be that the shop offers some discount on Smartphones who buy expensive items.

5.4 FP-Tree Algorithm vs Apriori Algorithm

1. Data Structure: FP-Tree Algorithm uses tree structure for mining frequent items whereas Apriori Algorithm relies on itemset lattice.

2. Memory Usage: FP-Tree Algorithm eliminates the need to store the entire database. Apriori Algorithm needs to maintain item-sets and support value for each iteration and hence Apriori Algorithm requires more memory.

3. Runtime Efficiency: FP-Tree algorithm generally takes less computational time because it constructs tree in a single dataset scan whereas Apriori Algorithm scans the dataset multiple times and generates large item-sets which results in more computational time.

4. Implementation Complexity: FP-Tree Algorithm is relatively complex to implement due to construction and transversal of FP-Tree structure.

6. CONSLUSION

Association Rule mining plays an important role in extracting valuable insights from the transactional dataset of customers.
The study begins by providing overview of two famous association rule mining algorithms that is Apriori Algorithm & FP-Tree Algorithm. Association rules are discovered using both these algorithms using different set of parameters.
Various performance metrics such as execution time, memory consumption is measured to make comparison between these two algorithms.

Overall this study provides an understanding of strengths and limitations of the FP-Tree algorithm & Apriori algorithm for association rule mining.

7. REFERENCES

1. Python Notebook - https://github.com/f2015712/thesis_sid/blob/main/initial.ipynb

2. Dataset Source - https://www.kaggle.com/datasets/mkechinov/ecommerce-purchase-history-from-electronics-store

3. https://www.hindawi.com/journals/wcmc/2022/4434714/

4.https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=32b8e33e5dc8fe42184a6efc6468e5bf10408b5d

5. https://www.sciencedirect.com/science/article/abs/pii/S0957417404001101

6. https://www.sciencedirect.com/science/article/pii/S187705091401309X

7. https://iopscience.iop.org/article/10.1088/1742-6596/1255/1/012057/meta

8. http://www.ijmlc.org/papers/462-C015.pdf

9. https://mecs-press.org/ijmecs/ijmecs-v12-n5/IJMECS-V12-N5-2.pdf

10. https://www.mdpi.com/0718-1876/16/7/162

11. https://arxiv.org/abs/2012.08777

12. https://www.sciencedirect.com/science/article/pii/S1877050916313084