

Enhancing E-Commerce Through Recommendation Systems and Market Basket Analysis



Executive Summary:

In the ever-evolving landscape of e-commerce, the significance of recommendation systems and market basket analysis cannot be overstated. These tools have the potential to enhance user experiences, boost sales, and drive business growth. This executive summary presents a comprehensive analysis of the research conducted to determine the most effective algorithm for implementing these strategies.

**Literature Review**

The project embarked on a meticulous literature survey to understand the foundations and nuances of market basket analysis and recommendation systems. It is evident from the literature that these techniques have been instrumental in various industries, from retail to online marketplaces. Market basket analysis, in particular, has been a cornerstone of retail operations for many years. Identifying patterns in customer purchasing behavior allows businesses to make data-driven decisions, optimize inventory, and increase cross-selling and up-selling opportunities.

Recommendation systems, on the other hand, have gained prominence in the digital era. These systems use algorithms to analyze user data and provide personalized product or content recommendations. They are widely utilized in e-commerce, streaming services, and online content platforms. A well-implemented recommendation system can significantly improve user engagement, retention, and revenue.

**Algorithmic Research**

The research phase of this project focused on exploring the intricacies of association rule mining, a fundamental aspect of market basket analysis. Association rule mining involves discovering relationships or patterns within large datasets. In the context of e-commerce, this translates to understanding the purchasing habits of customers—what items tend to be bought together, and how these insights can be leveraged for targeted marketing and sales strategies.

Three prominent algorithms—Apriori, FP-Growth, and ECLAT—were scrutinized to determine their suitability for recommendation systems in e-commerce.

**Algorithm Comparison**

One of the central objectives was to discern which of the three algorithms stands out as the most efficient and effective in this context. The findings from extensive experimentation offer valuable insights:

FP-Growth Emerges as the Fastest: Among the algorithms investigated, FP-Growth displayed the highest efficiency, making it a promising choice for e-commerce recommendation systems. This algorithm is known for its ability to construct a compact data structure called the FP-tree, which allows for faster mining of frequent item sets.

FP-Growth and Apriori Yield Comparable Results: In terms of average lift, support, and confidence, FP-Growth and Apriori demonstrated similar performance. This suggests that either algorithm could be a viable option for market basket analysis and recommendation systems. However, FP-Growth's efficiency gives it an edge in scenarios where real-time or near-real-time recommendations are crucial.

ECLAT's Efficiency Challenges: ECLAT, while a robust algorithm in many respects, exhibited significant computational inefficiency. Its runtime was notably slower, requiring substantially more time to process data compared to Apriori and FP-Growth. This could be attributed to its vertical data format, which may not be well-suited for large datasets.

**The Algorithmic Dilemma**

The research journey highlighted the ongoing debate and confusion surrounding the choice of algorithm in the field of recommendation systems and market basket analysis. This confusion arises from the trade-offs between speed and computational complexity. Determining the best algorithm is critical due to its far-reaching implications.

**The Significance of Algorithmic Choice**

The choice of algorithm for recommendation systems and market basket analysis in e-commerce is a pivotal decision that directly impacts user experiences, operational costs, and competitive advantage. Understanding the significance of this choice is essential for organizations seeking to thrive in the competitive e-commerce landscape.

User Experience Enhancement: The selected algorithm directly influences the quality of recommendations made to users. A more efficient algorithm can provide real-time recommendations, enhancing user satisfaction and engagement. Users are more likely to continue using a platform that understands their preferences and needs.

Operational Cost Optimization: Computational efficiency directly impacts operational costs. An inefficient algorithm can strain computational resources, leading to increased expenses. By selecting an algorithm that strikes a balance between accuracy and efficiency, organizations can optimize their operations and allocate resources more effectively.

Competitive Advantage: In the fiercely competitive e-commerce landscape, staying ahead requires efficient recommendation systems. The right algorithm can provide a competitive edge by driving sales, increasing customer engagement, and fostering customer loyalty. Organizations that harness the power of recommendation systems can differentiate themselves and capture market share.

**Algorithmic Challenges and Confusion**

The ongoing debate and confusion surrounding algorithm choice in recommendation systems and market basket analysis are rooted in several factors:

Diverse Business Needs: Organizations have diverse business needs and objectives. What works for one company may not be suitable for another. This diversity contributes to the ongoing discussion regarding the ideal algorithm.

Data Volume and Complexity: The scale and complexity of data in e-commerce are continuously evolving. Algorithms that perform well with small datasets may not scale effectively to manage large, dynamic datasets. This complexity adds to the challenge of algorithm selection.

Real-Time Requirements: Some e-commerce platforms require real-time or near-real-time recommendation capabilities. This necessitates algorithms that can process and generate recommendations quickly. However, achieving real-time recommendations often comes at the cost of computational complexity.

Algorithmic Advancements: The field of recommendation systems is dynamic, with ongoing advancements and the emergence of new algorithms. This continuous evolution adds to the complexity of selecting the most suitable algorithm.

**Conclusion**

In conclusion, the research project has shed light on the significance of selecting the appropriate algorithm for enhancing e-commerce through recommendation systems and market basket analysis. The findings indicate that FP-Growth exhibits remarkable efficiency, while ECLAT lags behind due to its computational demands. The choice between FP-Growth and Apriori can be nuanced, depending on specific business requirements.

The evolving nature of e-commerce necessitates continual exploration and adaptation of algorithms to meet the demands of an ever-growing user base. As the digital marketplace continues to expand, the selection of the optimal algorithm becomes pivotal in delivering personalized recommendations and driving e-commerce success.

This executive summary encapsulates the key findings and implications of the research, offering a comprehensive understanding of the intricate relationship between recommendation systems, market basket analysis, and algorithmic choice in the realm of e-commerce. The importance of efficient algorithms in driving business growth and maintaining competitiveness cannot be overstated.

Contents

[Executive Summary: 3](#_Toc146476039)

[Introduction 11](#_Toc146476040)

[1.1 Goal of the project: 12](#_Toc146476041)

[1.2 The aim of the current project is: 12](#_Toc146476042)

[1.3 Objectives: 13](#_Toc146476043)

[1.4 Route Map: 13](#_Toc146476044)

[Literature Review 14](#_Toc146476045)

[2.1 Understanding Customer Buying Psychology Recommendation Systems, and Market Basket Analysis 14](#_Toc146476046)

[2.1.1 Customer Buying Psychology: A Foundation for Retail Strategy 14](#_Toc146476047)

[2.1.2 Transition to Digital Retail: The Rise of Recommendation Systems 14](#_Toc146476048)

[2.1.3 Personalization and Customer Engagement: The Role of Recommendation Systems 14](#_Toc146476049)

[2.1.4 Market Basket Analysis: From Transactional Data to Strategic Insights 15](#_Toc146476050)

[2.1.5 Unearthing Hidden Patterns: Market Basket Analysis and Customer Behavior 15](#_Toc146476051)

[2.1.6 Synthesis: Bridging Psychology, Technology, and Retail Strategy 15](#_Toc146476052)

[2.2 Market Basket Analysis 15](#_Toc146476053)

[2.2.1 History of market basket analysis 16](#_Toc146476054)

[2.2.2 How market basket analysis is used today? 16](#_Toc146476055)

[2.2.3 Importance and effectiveness of market basket analysis in the online retail market 17](#_Toc146476056)

[2.2.4 Challenges and limitations of MBA 17](#_Toc146476057)

[2.2.5 Applications 18](#_Toc146476058)

[2.3 Market Basket Analysis- Algorithms 20](#_Toc146476059)

[2.3.1 Association Rule 20](#_Toc146476060)

[2.3.2 Early Insights and Discovery: Pioneering Applications 20](#_Toc146476061)

[2.3.3 Methodologies: Unveiling Hidden Connections 20](#_Toc146476062)

[2.3.4 Terms associated with Association Rule 20](#_Toc146476063)

[2.3.5 Frequent Itemset Generation 23](#_Toc146476064)

[2.3.6 Algorithms for Frequent Itemset Generation 24](#_Toc146476065)

[2.3.7 Interpreting Results 25](#_Toc146476066)

[2.4 Relevant Research Work 25](#_Toc146476067)

[2.4.1 Importance of Algorithm Selection in Market Basket Analysis 25](#_Toc146476068)

[2.4.2 Risks of Incorrect Algorithm Selection 25](#_Toc146476069)

[2.4.3 Apriori Algorithm: Balancing Simplicity and Efficiency 26](#_Toc146476070)

[2.4.4 FP-Growth Algorithm: Optimizing Memory Usage and Performance 26](#_Toc146476071)

[2.4.5 Eclat Algorithm: Vertical Approach for Sparse Datasets 27](#_Toc146476072)

[2.4.6 Convergence and Considerations 27](#_Toc146476073)

[2.5 Conclusion: Navigating the Algorithmic Landscape 27](#_Toc146476074)

[Methodology 29](#_Toc146476075)

[3.1 Data Collection 29](#_Toc146476076)

[3.2 Project Setup 29](#_Toc146476077)

[3.3 Technology Stack 30](#_Toc146476078)

[3.4 Dataset Description 31](#_Toc146476079)

[3.5 Data Pre-Processing 34](#_Toc146476080)

[3.5.1 Handling Missing Values: 34](#_Toc146476081)

[3.6 Exploratory Data Analysis: 35](#_Toc146476082)

[Strategy for Basket Analysis 43](#_Toc146476083)

[4.1 Apriori Algorithm 43](#_Toc146476084)

[4.2 Frequent Pattern Growth Algorithm 46](#_Toc146476085)

[4.3 Eclat Algorithm 48](#_Toc146476086)

[4.4 Recommender System 50](#_Toc146476087)

[Implementation 51](#_Toc146476088)

[5.1 Apriori Algorithm 51](#_Toc146476089)

[5.2 FP-Growth Algorithm 54](#_Toc146476090)

[5.3 Eclat Algorithm 55](#_Toc146476091)

[5.4 Association Results: 57](#_Toc146476092)

[5.5 Comparing performance and Accuracy of Association rules. 57](#_Toc146476093)

[5.5.1 Analysis on Apriori algorithm 57](#_Toc146476094)

[5.5.2 Analysis on FP-Growth algorithm 58](#_Toc146476095)

[5.5.3 Analysis on ECLAT Algorithm 58](#_Toc146476096)

[5.6 Comparing the Time Complexity between algorithms: 59](#_Toc146476097)

[5.7 Accuracy by comparing the support, lift, confidence, conviction, and leverage. 59](#_Toc146476098)

[5.7.1 Support 59](#_Toc146476099)

[5.7.2 Lift 61](#_Toc146476100)

[5.7.3 Confidence 61](#_Toc146476101)

[5.7.4 Conviction 63](#_Toc146476102)

[5.7.5 Leverage 64](#_Toc146476103)

[5.8 Comparing Apriori, FPGrowth and ECLAT algorithm: 65](#_Toc146476104)

[Conclusion 66](#_Toc146476105)

[Future Work 68](#_Toc146476106)

[Bibliography 71](#_Toc146476107)

**List of Figures**

Figure 1: Project Methodology 29

Figure 2: Entity Relationship Diagram of Dataset 31

Figure 3: Shape of Aisles dataset 32

Figure 4: Shape of Department dataset 32

Figure 5: Shape of Order\_Products\_prior/Order\_Products\_train dataset 32

Figure 6: Shape of Orders dataset 33

Figure 7: Shape of products dataset 34

Figure 8: Missing values details in dataset 34

Figure 9: Removal of Missing values 34

Figure 10: Columns of Final Dataset 35

Figure 11: Peak Hours of the day 36

Figure 12: Peak days of the Week 37

Figure 13: Hours of the Day vs Day of the Week Heatmap 38

Figure 14: Days Since Prior Instacart Order 38

Figure 15: Top 15 most sold products 40

Figure 16: Top 15 most ordered products from Aisles 41

Figure 17: Top 15 most ordered products from Departments 42

Figure 18: Pie Chart – Percentage of products in each Department 43

Figure 19: Apriori Algorithm example 45

Figure 20: FP Growth Algorithm example 47

Figure 21 : Frequent Pattern Tree Structure 48

Figure 22: ECLAT Algorithm Flow chart 49

Figure 23: One-Hot Encoded dataset 52

Figure 24: Minimum Support Items of Apriori data 53

Figure 25: Correlation between Items of apriori data 53

Figure 26: Minimum Support Items of FP Growth data 54

Figure 27: Correlation between Items of FP Growth data 55

Figure 28: One-Hot Encoding for ECLAT data 55

Figure 29: Combining frequent item sets 56

Figure 30: Confidence threshold chart 57

Figure 31: Bar chart- Time consumptions by different algorithm 59

Figure 32: Support comparison between Apriori and FP-Growth 60

Figure 33: Lift comparison between Apriori and FP-Growth 61

Figure 34: Confidence comparison between Apriori and FP-Growth 62

Figure 35: Conviction comparison between Apriori and FP-Growth 63

Figure 36: Leverage comparison between Apriori and FP-Growth 64

**List of Tables**

*Tabel 1: Objectives of the Project 13*

*Tabel 2: Route Map of the Project 13*

*Table 3 : Description of Aisles data set 31*

*Table 4 : Description of Department data set 32*

*Table 5 : Description of Order\_Products\_prior/Order\_Products\_train data set 32*

*Table 6 : Description of Orders data set 33*

*Table 7 : Description of products data set 33*

*Tabel 8 : Comparison between Algorithms 66*

**List of Equation**

*Equation 1: To find Support of an itemset. 21*

*Equation 2: To find confidence between two itemset. 21*

*Equation 3: To find Lift for the effectiveness of rule. 22*

Introduction

In the ever-evolving realm of retail, where consumer behaviors are shaped by dynamic trends and a diverse array of products, the art of deciphering shopping patterns has emerged as a critical factor in optimizing business strategies (Kim, Fiore, Niehm, & Jeong, 2010). Market Basket Analysis (MBA), a data mining technique with its origins deeply rooted in the retail landscape, has become a fundamental tool for uncovering hidden connections within transactional data (Surjandari & Seruni, 2010). Its journey from its inception to its contemporary significance reflects the evolution of data-driven decision making and its pivotal role in modern retail operations.

Origins and Evolution: The Birth of Market Basket Analysis

The origins of Market Basket Analysis can be traced back to the early 1950s, a time when the concept of consumer behavior analysis was gaining prominence. As the scale of retail operations expanded and consumer preferences diversified, retailers and researchers recognized the need for a systematic approach to decode intricate shopping behaviors. The traditional method of manually observing customers in physical stores proved inadequate in capturing the complexity of modern shopping trends (Applebaum, 1951). In response, Market Basket Analysis emerged as a groundbreaking technique that harnessed data mining principles to uncover associations between items frequently purchased together.

Catalyst for Innovation: Navigating Complexity and Data Deluge

The initial application of Market Basket Analysis found its most iconic example in the supermarket landscape. The revelation that customers purchasing diapers often exhibited a preference for beer highlighted the power of this analytical methodology. This serendipitous discovery prompted retailers to strategically alter product placements, design marketing campaigns, and even reconfigure store layouts to capitalize on these newfound correlations (Market analysis, n.d.). As technology advanced and digital retail platforms emerged, MBA seamlessly transitioned from brick-and-mortar stores to the virtual realm, catering to the burgeoning world of e-commerce transactions.

Modern Significance: Data-Driven Insights in the Digital Age

In the contemporary landscape, characterized by a data-driven revolution, Market Basket Analysis has assumed an unparalleled importance. The proliferation of e-commerce platforms and the accumulation of massive datasets have magnified the relevance of MBA. Businesses are now armed with vast troves of transactional data, but the true value lies in extracting actionable insights from this abundance. MBA empowers retailers not only to understand historical purchase patterns but also to forecast future trends, predict consumer preferences, and enhance user experiences through personalized recommendations (Joshi, Khanna, Sabale, & Tathawade, 2017).

Moreover, in an era where personalized recommendations drive customer satisfaction and loyalty, Market Basket Analysis seamlessly integrates into the engines of recommendation systems (Kumaran & Sankar, 2013). By deciphering which products are frequently co-purchased, businesses can fine-tune their algorithms to deliver tailor-made suggestions that resonate with individual preferences.

Conclusion: Shaping the Future of Retail Strategy

In conclusion, the evolution of Market Basket Analysis from its humble beginnings to its current position as a linchpin in the data-driven retail landscape underscores its enduring relevance. As businesses harness its capabilities to unravel hidden correlations, optimize inventory management, and enhance customer experiences, it is evident that the insights derived from this analytical technique will continue to shape the trajectory of retail strategies (Glanz, Bader, & Iyer, 2012). In a world where data is hailed as the currency of success, Market Basket Analysis stands as a testament to the transformative power of mining transactional data to glean invaluable insights, driving informed decisions and redefining the contours of retail excellence.

1.1 Goal of the project:

The primary objective of this project is to aid sellers on Instacart in examining the prevailing consumer actions and predicting future buying tendencies. Utilizing data from customer transactions can provide insights into customer purchasing habits, enabling the provision of appropriate product bundles, exclusive offers, effective assortment planning, and efficient inventory management. These measures are aimed at retaining customers, augmenting sales, and extending the duration of customer relationships.

1.2 The aim of the current project is:

1. To Uncover Consumer Behavior Patterns through Market Basket Analysis
2. Implement the Apriori, FP-Growth, and Eclat algorithms.
3. Compare the outcomes and complexity of the later methods.
4. Investigate association rules to learn more about products that should be suggested together.

In this project, we focus on the below objectives to achieve the goal.

1.3 Objectives:

|  |  |
| --- | --- |
|  |  |
| 1 | To investigate through the methods that were previously employed to understand clients through market research. |
| 2 | To assess existing market basket analysis methodologies and apply them to a dataset to determine their dimension. |
| 3 | To determine if the techniques are effective by evaluating these outcomes. |
| 4 | Based on their performance, identify the best market basket analysis algorithm. |

*Tabel 1: Objectives of the Project*

The route map mentioned below is followed throughout the project to achieve the objectives.

1.4 Route Map:

|  |  |
| --- | --- |
|  |  |
| Literature Review | A thorough compilation of the published studies on customer purchasing habits and market basket analysis. A summary of the research's major concepts and conclusions, organized chronologically. |
| Main Methodology | Analyzing data to identify client purchasing behaviors using the market basket analysis approach, and then assessing the usefulness of various algorithms to business growth. |
| Conclusion | The research project's final summary. |
| Recommendation | Based on the recommendation system and the best algorithm for analysis, recommending relevant goods to the consumer. |

*Tabel 2: Route Map of the Project*

Literature Review

2.1 Understanding Customer Buying Psychology Recommendation Systems, and Market Basket Analysis

The dynamics of consumer behavior have long captivated researchers and practitioners in the field of marketing and retail. Understanding how customers perceive, evaluate, and make purchasing decisions is crucial for businesses seeking to tailor their strategies to individual preferences (Mende, Bolton, & Bitner, 2013). This literature review delves into the realm of customer buying psychology, tracing its evolution, and subsequently explores how the advent of recommendation systems and market basket analysis has reshaped the way businesses engage with customers and optimize their operations.

2.1.1 Customer Buying Psychology: A Foundation for Retail Strategy

Consumer buying psychology delves into the intricate processes that govern how individuals perceive products, evaluate their features, and ultimately decide to make a purchase (Singh, Katiyar, & Verma, 2014). Early research in this domain, notably driven by consumer psychology pioneers like Abraham Maslow, highlighted the significance of emotional and psychological factors in purchasing decisions. As the field progressed, cognitive theories such as the Theory of Planned Behavior and the Elaboration Likelihood Model provided frameworks to dissect how customers process information, assess risks, and form attitudes toward products.

2.1.2 Transition to Digital Retail: The Rise of Recommendation Systems

The shift to digital commerce brought forth new challenges and opportunities for businesses. The abundance of choices, coupled with the absence of in-person interactions, underscored the need for effective personalization strategies (Masthoff & Vassileva, 2015). The advent of recommendation systems emerged as a response to this challenge. Collaborative filtering, content-based filtering, and hybrid methods became the cornerstones of these systems, aimed at deciphering customer preferences by analyzing their historical behavior and product interactions.

2.1.3 Personalization and Customer Engagement: The Role of Recommendation Systems

Recommendation systems, fueled by customer behavior data, have revolutionized e-commerce by offering personalized product suggestions that align with individual preferences. This personalized approach leverages insights from customer buying psychology, catering to emotional triggers, cognitive biases, and preferences that influence purchasing decisions (Areelu & Akinsola, 2014). The seamless integration of these insights into recommendation algorithms has led to enhanced user engagement, increased conversions, and customer satisfaction.

2.1.4 Market Basket Analysis: From Transactional Data to Strategic Insights

While recommendation systems focus on individualized experiences, market basket analysis broadens the scope by examining associations among products purchased together. The introduction of market basket analysis can be linked to the realization that these associations often reveal deeper customer preferences and behaviors (Brenes, Ciravegna, Ciravegna, & Montoya, 2015). The serendipitous discovery of unexpected product co-purchases, exemplified by the "diapers and beer" phenomenon, highlighted the untapped potential in analyzing transactional data.

2.1.5 Unearthing Hidden Patterns: Market Basket Analysis and Customer Behavior

Market basket analysis, powered by techniques like Apriori and FP-Growth, delves into transactional datasets to uncover associations that may not be apparent on the surface. By identifying products frequently co-purchased, businesses gain insights into customer preferences that transcend individual recommendations (Stanisic & Tomovic, 2015). These discovered patterns provide a strategic advantage in optimizing product placement, devising cross-selling strategies, and enhancing overall marketing initiatives.

2.1.6 Synthesis: Bridging Psychology, Technology, and Retail Strategy

The evolution of customer buying psychology, the invention of recommendation systems, and the introduction of market basket analysis share a common thread: the quest for a deeper understanding of consumer preferences and behaviors (Li, Li, & Hudson, 2013). The fusion of these insights with technological advancements has ushered in an era where businesses can strategically personalize user experiences, optimize sales strategies, and forge stronger customer relationships. As the digital retail landscape continues to evolve, harnessing the symbiotic power of customer psychology, recommendation systems, and market basket analysis becomes not just a competitive advantage, but a strategic imperative for success (Gjino & (Findiku), 2014).

2.2 Market Basket Analysis

Market basket analysis (MBA) is a data mining technique that analyzes customer transactional data to identify patterns of association among items (Dippold & Hruschka, 2013). It is often used in retail and e-commerce organizations to develop association rules that explain the links between various things, as well as to make educated decisions regarding product placement, pricing, and promotions.

MBA is based on the principle that items that are frequently purchased together are likely to be related. For example, a customer who buys bread is more likely to also buy milk, eggs, and butter. By identifying these associations, businesses can target their marketing and promotions more effectively.

2.2.1 History of market basket analysis

The origins of market basket analysis can be traced back to the 1960s, when it was first used by grocery stores to identify products that were frequently purchased together. The technique was further developed in the 1980s and 1990s, with the advent of more powerful computers and data mining algorithms (Gunawardana & Shani, 2015).

The first known use of market basket analysis was in 1967 by a grocery store chain in the United States. The store was using a system called "EPOS" (electronic point-of-sale) to track customer purchases. The EPOS system generated data on which products were purchased together, and this data was used to create association rules.

The first academic paper on market basket analysis was published in 1993 by Rakesh Agrawal and Ramakrishnan Srikant. The paper, entitled "Fast Algorithms for Mining Association Rules," introduced the Apriori algorithm, which is a popular algorithm for generating association rules.

Since the 1990s, market basket analysis has become a widely used technique in retail and e-commerce. It is also used in other industries, such as healthcare, finance, and telecommunications.

2.2.2 How market basket analysis is used today?

Market basket analysis is used today by businesses of all sizes to gain insights into customer behavior and make better business decisions (Borse, Pisal, Chorghe, & Fate, 2017). Some of the specific ways in which market basket analysis is used include:

* Identifying cross-selling opportunities: Businesses can use market basket analysis to identify products that are frequently purchased together. This information can then be used to cross-sell products to customers. For example, a grocery store might recommend milk to customers who have purchased bread.
* Personalizing recommendations: Businesses can use market basket analysis to generate personalized recommendations for customers. This information can be used to recommend products that are likely to be of interest to the customer. For example, an online retailer might recommend shoes to a customer who has purchased clothes in the past.
* Improving product placement: Businesses can use market basket analysis to improve the placement of products in stores and on websites. This information can be used to place products that are frequently purchased together near each other. For example, a grocery store might place milk and bread near each other.
* Optimizing pricing: Businesses can use market basket analysis to optimize pricing for products. This information can be used to price products that are frequently purchased together lower than products that are not frequently purchased together. For example, a grocery store might price milk and bread lower than other products.
* Managing inventory: Businesses can use market basket analysis to manage inventory by identifying products that are not selling well or that are overstocked. This information can be used to reduce waste and improve profits.

2.2.3 Importance and effectiveness of market basket analysis in the online retail market

* Amazon: Amazon uses MBA to recommend products to customers based on their past purchase history and browsing behavior. This is why you might see products that you have previously viewed or purchased being recommended to you on Amazon.
* Walmart: Walmart uses MBA to optimize the placement of products in its stores. For example, Walmart might place products that are frequently purchased together near each other. This can help to increase impulse purchases.
* eBay: eBay uses MBA to target marketing campaigns to specific groups of customers. For example, eBay might send targeted emails to customers who have previously purchased certain products.

These are just a few examples of how MBA is used in the online retail market. As the online retail market continues to grow, MBA is likely to become an even more valuable tool for businesses (Videla-Cavieres & Ríos, 2014).

2.2.4 Challenges and limitations of MBA

Market Basket Analysis (MBA) is a powerful tool for uncovering purchasing patterns, but like any analytical technique, it comes with its own set of challenges and limitations. Here are some key challenges to consider:

* Data Quality and Quantity: The effectiveness of MBA heavily relies on the quality and quantity of data. Inaccurate or incomplete transactional data can lead to erroneous associations and insights. Additionally, as the volume of data grows, the computational resources required for analysis increase, potentially leading to longer processing times (Ahima, 2012).
* Sparse Data: In real-world scenarios, not all products are purchased together frequently. This results in sparse data where many potential item combinations have low occurrence rates. Sparse data can hinder the discovery of meaningful associations and limit the practicality of the results (Trendafilov, Kleinsteuber, & Zou, 2014).
* Association Size: As the number of products in a dataset increases, the number of possible item combinations grows exponentially. This can lead to an explosion in the number of associations to analyze, making it challenging to focus on the most relevant and actionable insights.
* Multiple Testing: In datasets with a large number of associations, the risk of false discoveries increases. When conducting multiple tests to identify significant associations, the likelihood of finding spurious patterns that appear significant by chance also rises.
* Lack of Context: MBA focuses solely on transactional data and does not consider external factors or contextual information that could influence purchasing decisions. Understanding the reasons behind associations and their implications may require additional qualitative analysis.
* Changing Patterns: Consumer preferences and behaviors evolve over time due to factors like trends, seasons, and external events. MBA results might become outdated quickly, making continuous monitoring and updating of associations necessary for accurate insights.
* Implicit vs. Explicit Associations: MBA often captures implicit associations, meaning products that are purchased together but might not have a direct relationship. Explicit relationships, where the items are naturally related, can be more actionable. Distinguishing between the two is a challenge.
* Privacy Concerns: In some cases, transactional data used for MBA might contain sensitive customer information. Ensuring data privacy and complying with regulations becomes crucial, particularly in online retail environments.
* Dimensionality Reduction: As the number of products increases, the dimensionality of the data grows, leading to a phenomenon known as the "curse of dimensionality." This can make it difficult to process and analyze the data efficiently.
* Interpreting Results: While MBA can identify associations, interpreting the implications of these associations requires domain expertise. Not all associations necessarily translate into actionable insights, and determining how to capitalize on them can be challenging.

Despite these challenges, Market Basket Analysis remains a valuable tool in the retail analytics toolkit.

2.2.5 Applications

Market Basket Analysis (MBA) holds a wide range of applications across various industries, primarily in understanding customer behaviors and optimizing business strategies.

Here are some key applications:

* Retail Strategy Optimization: MBA is extensively used in the retail sector to optimize product placement within physical stores. By identifying products frequently purchased together, retailers can strategically position them in proximity, leading to increased visibility and potential cross-selling opportunities.
* Cross-Selling and Upselling: Businesses employ MBA to suggest complementary or related products to customers, enhancing cross-selling and upselling efforts. For instance, when a customer adds a laptop to their cart, MBA can recommend laptop accessories like cases or chargers.
* Inventory Management: MBA helps in effective inventory management by identifying product associations. This enables retailers to optimize stock levels of items that are commonly purchased together, reducing overstocking or stockouts.
* Assortment Planning: Retailers can use MBA insights to plan product assortments more strategically. By understanding which products are frequently purchased together, they can curate assortments that cater to specific customer preferences and boost overall sales.
* Promotional Campaigns: MBA guides targeted marketing campaigns. Retailers can create promotions or discounts that align with product associations, enticing customers to purchase related items and increasing the average transaction value.
* Online Retail Recommendations: In e-commerce, MBA powers recommendation systems. Analyzing user purchase history and behaviors, it offers personalized product suggestions to enhance the online shopping experience and drive sales.
* Website Layout Optimization: For e-commerce platforms, MBA influences website layout. By placing associated products together, websites can guide customers towards relevant items, increasing engagement and conversions.
* Restaurant Menu Engineering: MBA is not limited to retail. In the restaurant industry, it helps optimize menus by identifying popular food and beverage pairings. This enhances menu offerings and profitability.
* Telecom and Cable Service Bundling: MBA assists telecom companies in bundling services like internet, cable, and phone plans. By analyzing which services are frequently purchased together, companies can design attractive packages.
* Healthcare Cross-Selling: In healthcare, MBA can be applied to suggest related medical services or products based on patient history, promoting cross-selling while enhancing patient care.
* Online Streaming Platforms: Platforms like Netflix use MBA to suggest movies or shows based on users' viewing history and preferences, thereby keeping users engaged and subscribed.
* Online Marketplaces: E-commerce platforms like Amazon use MBA to recommend products based on browsing and purchase history, enhancing customer experience and driving sales.

By unraveling hidden patterns in transactional data, MBA empowers businesses to make informed decisions that enhance customer satisfaction, optimize strategies, and drive revenue growth (Musalem, Aburto, & Bosch, 2018).

2.3 Market Basket Analysis- Algorithms

2.3.1 Association Rule

Association rules are a fundamental concept within Market Basket Analysis (MBA), a technique used to uncover relationships and patterns among products frequently purchased together. These rules provide actionable insights by revealing the co-occurrence of items in customers' transactions. The exploration of association rules within the realm of Market Basket Analysis has contributed significantly to understanding customer behavior and optimizing retail strategies (Wu, Zhu, Wu, & Ding, 2014). This literature review delves into the evolution, methodologies, applications, and challenges surrounding association rules, highlighting their pivotal role in modern data-driven decision-making.

2.3.2 Early Insights and Discovery: Pioneering Applications

The inception of association rules traces back to the groundbreaking work of (Agrawal & Srikant 1994), who introduced the Apriori algorithm. This marked a significant leap in the field of retail analytics by offering a systematic approach to uncovering item associations. The algorithm was an innovative solution to the problem of identifying frequent items sets efficiently. Subsequent studies built upon this foundation, leading to the introduction of advanced algorithms like FP-Growth, Eclat, and FPMax.

2.3.3 Methodologies: Unveiling Hidden Connections

Association rule mining techniques are the backbone of MBA. These methods sift through transactional data to discover patterns and relationships between items. The Apriori algorithm, based on support and confidence measures, identifies frequent itemset and generates rules with predefined thresholds (Stanisic & Tomovic, 2015). FP-Growth, on the other hand, employs a tree-based structure to compress the data, speeding up the process of generating association rules (Rathi & Dhote, 2015). These methodologies not only provide insights into item co-occurrences but also enable businesses to predict purchasing behavior.

2.3.4 Terms associated with Association Rule

**Transaction (T)**: A transaction is a record of items purchased together in a single instance . It can be represented as a set of items denoted by T. For example, T = {bread, milk, eggs} represents a transaction where a customer bought bread, milk, and eggs together.

**Itemset (I)**: An itemset is a collection of items purchased together in one or more transactions. It is denoted by I. For example, I = {bread, milk} represents an itemset consisting of bread and milk.

Analyzing the item sets allows you to discover which items tend to be purchased together more frequently. For instance, if you find that {bread, milk} is a frequently occurring itemset, you can strategically place these items closer to each other on your online grocery store website, potentially leading to increased sales and customer satisfaction. Item sets lay the groundwork for more advanced analyses like association rule mining, where you can uncover relationships and patterns among these item sets to make informed decisions for optimizing product recommendations, marketing strategies, and overall retail experiences (Khan & Kumar, 2013).

**Support Count (N)**: The support count of an itemset represents the number of transactions in which that itemset appears. It is denoted by N. Mathematically, for an itemset A

**Support (A)**: Support is a normalized measure that indicates the proportion of transactions containing a specific itemset. It is calculated as the ratio of the support count of the itemset to the total number of transactions. Mathematically:

***Support(A) = Support Count(A) / Total Number of Transactions***

*Equation 1: To find Support of an itemset.*

**Confidence**: Confidence is a crucial metric in Market Basket Analysis that quantifies the likelihood that a certain itemset B is purchased when another itemset A is bought. Understanding confidence helps us identify strong associations between items and make informed decisions. Let us delve into the concept of confidence, including equations and an illustrative example:

Equation Explained:

Confidence (C): Confidence is calculated by dividing the support count of the combined itemset (A ∪ B) by the support count of the antecedent itemset (A). Mathematically, for association rule A → B:

***Confidence(A → B) = Support Count(A ∪ B) / Support Count(A)***

*Equation 2: To find confidence between two itemset.*

**Lift**: is a critical metric in Market Basket Analysis that measures how much the presence of one itemset influences the presence of another itemset beyond random chance. Understanding lift helps us uncover meaningful associations between items and make informed decisions. Let us delve into the concept of lift, including equations and an illustrative example:

Equation Explained:

Lift (l): Lift is calculated by dividing the confidence of an association rule (A → B) by the support of the consequent itemset (B). Mathematically, for association rule A → B:

***Lift(A → B) = Confidence(A → B) / Support(B)***

*Equation 3: To find Lift for the effectiveness of rule.*

Let us understand all the terms with a simple example.

Consider a small coffee shop with the following transactions:

Transaction 1: {espresso, croissant}

Transaction 2: {latte, muffin, croissant}

Transaction 3: {espresso, latte, cappuccino}

Transaction 4: {latte, croissant}

Transaction 5: {espresso, muffin}

Explaining Concepts:

**Transaction**: Each transaction represents a customer's order. For instance, Transaction 1 {espresso, croissant} means a customer ordered an espresso and a croissant together.

**Itemset**: An itemset is a set of items purchased in a transaction. For example, {espresso, croissant} represents the combination of an espresso and a croissant.

**Support Count**: The support count of an itemset is the number of transactions in which the itemset appears. For {espresso, croissant}, the support count is one because it appears in Transaction 1.

**Support**: Support indicates the proportion of transactions containing a specific itemset. The support for {espresso, croissant} is calculated as:

Support({espresso, croissant}) = Support Count({espresso, croissant}) / Total Number of Transactions

Support({espresso, croissant}) = 1 / 5 = 0.2 or 20%

**Confidence**: Confidence measures the likelihood of purchasing itemset B when itemset A is bought. Let us consider the association rule {espresso} → {croissant}. The support count of {espresso} is 3 (Transactions 1, 3, and 5), and the support count of {espresso, croissant} is one. Hence, the confidence is:

Confidence({espresso} → {croissant}) = Support Count({espresso, croissant}) / Support Count({espresso})

Confidence({espresso} → {croissant}) = 1 / 3 ≈ 0.33 or 33%

**Lift**: Lift quantifies how much the presence of itemset A influences the presence of itemset B beyond random chance. For the association rule {espresso} → {croissant}, the confidence is 33%, and the support count of {croissant} is 3. Thus, the lift is calculated as:

Lift({espresso} → {croissant}) = Confidence({espresso} → {croissant}) / Support({croissant})

Lift({espresso} → {croissant}) ≈ 0.33 / 0.6 ≈ 0.55

**Interpreting Insights**:

In this coffee shop example, we have explored transaction, itemset, support count, support, confidence, and lift. These concepts help uncover relationships between items and provide insights for decision-making. For instance, the lift value of approximately 0.55 for the association rule {espresso} → {croissant} suggests that customers who order espresso are about 0.55 times as likely to also order a croissant compared to random chance.

2.3.5 Frequent Itemset Generation

Frequent Itemset Generation is a pivotal process within Market Basket Analysis, focusing on identifying combinations of items that frequently co-occur in transactions. This process is essential for unraveling hidden patterns and associations within extensive transaction datasets (Rajalakshmi, Purusothaman, & Nedunchezhian, 2011). There are several algorithms that facilitate this task, including the Apriori algorithm, the FP-Growth algorithm, and the Eclat algorithm.

Significance of Frequent Itemset Generation:

The primary objective of Frequent Itemset Generation is to pinpoint sets of items that appear together frequently in transactions. These frequent item sets hold the key to understanding customer behavior, preferences, and buying patterns (Long, Bakar, & Hamdan, 2011). Such insights are invaluable for crafting informed decisions related to product recommendations, cross-selling strategies, marketing campaigns, and inventory management.

2.3.6 Algorithms for Frequent Itemset Generation

2.3.6.1 Apriori Algorithm

* The Apriori algorithm is a well-known approach that follows an iterative strategy for generating candidate item sets and eliminating those that do not meet specified minimum support thresholds (Fang, 2013) .
* It operates based on the "Apriori principle," which posits that if an itemset is frequent, then all of its subsets must also be frequent.
* The algorithm commences by identifying frequent individual items (singleton item sets) and progressively extends its scope to larger item sets.
* It generates candidate item sets by combining frequent item sets from the previous iteration and discards those that fail to meet the minimum support criterion.
* This iterative process continues until no further frequent item sets can be derived.

2.3.6.2 FP-Growth Algorithm

* The FP-Growth (Frequent Pattern-Growth) algorithm presents an innovative approach that employs a compact data structure called an FP-Tree to efficiently mine frequent item sets (Rong, Xia, & Zhang, 2013).
* It circumvents the need to generate and store an extensive list of candidate item sets, as observed in the Apriori algorithm.
* The algorithm constructs the FP-Tree by scanning the transaction dataset and subsequently mines frequent item sets from this structured representation via a divide-and-conquer methodology.

2.3.6.3 Eclat Algorithm

* The Eclat (Equivalence Class Clustering and Bottom-Up Lattice Traversal) algorithm constitutes another vital player in the realm of Frequent Itemset Generation (Man & Jalil, 2019).
* Eclat employs a vertical approach, focusing on individual items and establishing connections between items by intersecting their transaction lists.
* This algorithm is particularly effective when dealing with datasets characterized by high dimensionality and sparsity.
* Eclat classifies items sharing the same transactions into equivalence classes, simplifying the process of locating intersections.
* It explores item sets through a bottom-up lattice traversal method, efficiently constructing larger item sets from smaller, frequent ones.

2.3.6.4 Steps in Frequent Itemset Generation

1. Data Preprocessing: Begin by cleaning and preprocessing the transaction dataset, eliminating duplicates and irrelevant data while transforming it into an appropriate format for analysis (Mohamed & Darwieesh, 2014).
2. Minimum Support Threshold: Define a minimum support threshold, which serves as the minimum occurrence frequency required for an itemset to be deemed "frequent."
3. Algorithm Selection and Execution

* Choose an algorithm such as Apriori, FP-Growth, or Eclat, based on the characteristics of your dataset.
* Execute the chosen algorithm to discover frequent item sets according to the specified minimum support threshold.

2.3.7 Interpreting Results

The frequent item sets generated through these algorithms provide valuable insights into prevalent item combinations. These insights can then be transformed into association rules, with metrics such as confidence and lift calculated to facilitate decision-making across multiple facets of business operations (Zhixin, Yusheng, & Dillon, 2013).

2.4 Relevant Research Work

The success of MBA hinges on the selection of an appropriate algorithm for generating frequent item sets—a task accomplished by algorithms such as the Apriori algorithm, FP-Growth algorithm, and Eclat algorithm. Extensive research efforts have been devoted to these algorithms, exploring their efficiencies, strengths, and limitations, and culminating in discussions surrounding the optimal choice for Market Basket Analysis.

2.4.1 Importance of Algorithm Selection in Market Basket Analysis

Choosing the optimal algorithm for Market Basket Analysis holds immense significance as it directly impacts the accuracy, efficiency, and insights derived from the analysis. The chosen algorithm influences both the runtime and memory requirements, key factors for scalability and feasibility (Nandagopal, Arunachalam, & Karthik, 2012). The insights gained from item associations directly inform business strategies, including product recommendations, cross-selling, and targeted marketing campaigns. Thus, selecting the most appropriate algorithm has far-reaching implications for data-driven decision-making.

2.4.2 Risks of Incorrect Algorithm Selection

Failing to choose the correct algorithm can lead to suboptimal results and potentially erroneous conclusions. Utilizing an algorithm ill-suited for the dataset's characteristics may result in excessive computational time, inefficient memory usage, or even incomplete identification of crucial item associations (Mahmood, Shahbaz, & Guergachi, 2014). Incorrect choices could lead to missed business opportunities, ineffective marketing strategies, and erroneous predictions. Therefore, a careful evaluation of dataset characteristics and algorithm attributes is paramount to ensure the credibility and applicability of the generated insights.

The selection of the most suitable algorithm for Market Basket Analysis (MBA) has been a topic of rigorous debate among researchers, each championing different algorithms—Apriori, FP-Growth, and Eclat—as the best choice. This ongoing discourse revolves around several critical factors, including computational efficiency, memory utilization, scalability, and adaptability to varying dataset characteristics. The arguments for and against each algorithm are vital to comprehending the nuances of their applicability.

2.4.3 Apriori Algorithm: Balancing Simplicity and Efficiency

Argument in Favor:

According to (Zein, 2016), Apriori algorithm highlight its simplicity and ease of implementation. Its intuitive nature makes it an excellent educational tool, aiding newcomers in grasping the fundamentals of association rule mini. Furthermore, the algorithm's variants, such as AprioriTid, address its inefficiencies by reducing the number of database scans and candidate itemset generation. These enhancements highlight its adaptability to manage diverse datasets efficiently (Mahmoodi, Mirzaie, & Mahmoudi, 2016).

Counterargument:

(Joshi, Khanna, Sabale, & Tathawade, 2017) contend that the Apriori algorithm's need for multiple database scans and candidate generation may lead to inefficiency, particularly when dealing with large datasets. The sheer computational cost of these operations can be a bottleneck, hindering its application on extensive transaction databases. This inefficiency has sparked the development of more streamlined algorithms to better address the demands of contemporary datasets.

2.4.4 FP-Growth Algorithm: Optimizing Memory Usage and Performance

Argument in Favor:

According to (Chang, Lin, Cheng, & Huang, 2016), FP-Growth algorithm emphasize its ability to tackle the challenges posed by massive datasets. The FP-Tree data structure significantly reduces memory requirements and speeds up frequent itemset generation, particularly in cases where items have high cardinality. Moreover, the algorithm's potential for parallelization and memory optimization further enhances its efficiency, rendering it a prime choice for data-intensive applications.

Counterargument:

(A, M, J, & N, 2018) contend that while the FP-Growth algorithm excels in terms of memory usage and runtime efficiency, its initial construction of the FP-Tree can be memory-intensive. This constraint necessitates careful consideration of available memory resources before deploying the algorithm. The memory-intensive nature of constructing the tree structure may limit its effectiveness on memory-constrained environments or datasets characterized by extreme sparsity.

2.4.5 Eclat Algorithm: Vertical Approach for Sparse Datasets

Argument in Favor:

According to (Trieu & Kunieda, 2012), the Eclat algorithm emphasize its memory efficiency and suitability for sparse datasets. The vertical approach based on transaction lists enables efficient generation of frequent item sets without the need for candidate generation. This approach aligns well with datasets containing numerous items and relatively few transactions. Eclat's focus on vertical intersections allows it to excel where other algorithms may falter.

Counterargument:

(Gajera, Limbad, & Badheka, 2015) contend that while Eclat excels in scenarios with sparse datasets, its performance may diminish on denser datasets. The reliance on vertical intersections and transaction list intersections may not be as effective when dealing with datasets characterized by a high density of transactions, potentially leading to suboptimal outcomes in such cases.

2.4.6 Convergence and Considerations

While each algorithm has its merits, the ideal choice hinges on a comprehensive understanding of the dataset characteristics and desired outcomes. Arguments and counterarguments provide valuable insights into the strengths and limitations of each algorithm. The convergence of perspectives suggests that no single algorithm universally excels in all scenarios. Therefore, algorithmic selection should be driven by a thorough evaluation of dataset density, cardinality, memory resources, and computational efficiency (Ishita & Rathod, 2016).

2.5 Conclusion: Navigating the Algorithmic Landscape

The contentious discourse surrounding the best algorithm for MBA underscores the complexities inherent in algorithmic selection. Researchers' arguments illuminate the nuanced trade-offs between simplicity, efficiency, and memory utilization. In the end, the best algorithm for a specific scenario hinge on a delicate balance between these factors and the unique attributes of the dataset (Divya & Neeru, 2011). As research advances and algorithms evolve, this ongoing dialogue will continue to inform businesses and analysts in making informed algorithmic choices that best suit their objectives and dataset characteristics.

The selection of the appropriate algorithm for generating frequent item sets in Market Basket Analysis is a pivotal decision that influences the accuracy, efficiency, and applicability of the insights extracted. Extensive research efforts have enhanced our understanding of the Apriori algorithm, FP-Growth algorithm, and Eclat algorithm, shedding light on their respective strengths and limitations. While each algorithm presents unique advantages, the final choice hinges on dataset characteristics, computational resources, and desired outcomes. Failing to make the right algorithmic choice can lead to suboptimal results and hinder data-driven decision-making processes (Joshi, Jadon, & Jain, 2010). Thus, a thoughtful evaluation of algorithms is crucial to unlock the full potential of Market Basket Analysis.

In the intricate landscape of Market Basket Analysis, the contention over the optimal algorithm underscores the dynamic nature of data mining. Throughout this discourse, the merits, and limitations of the Apriori algorithm, FP-Growth algorithm, and Eclat algorithm have been fervently debated, illuminating the multifaceted considerations in algorithmic selection.

In the realm of this project, our endeavor is to transcend the theoretical deliberations and embark on an empirical journey. By meticulously evaluating each algorithm's performance, efficiency, and adaptability to varying dataset scenarios, we aim to navigate the practical implications of algorithmic choice. Through rigorous testing, we will subject the algorithms to real-world datasets, unveiling their strengths and vulnerabilities in the context of Market Basket Analysis.

Methodology

The project follows the following approach for the methodology:

Planning

Data Collection

Data Validation & Cleaning

Data Preparation

Implementation

Implementation of Project

Building Model & Comparing Algorithms

Modelling

Conclusion

Figure 1: Project Methodology

3.1 Data Collection

The datasets is actually sourced from Kaggle which was indeed taken from Instacart Technology Company to perform the analysis. Instacart's databases contained comprehensive information on over 3 million grocery orders from over 200,000 Instacart members. Instacart's product and customer data include 50,000 distinct goods, the week and time of purchase, different product aisles and departments.

3.2 Project Setup

After gathering data from the Kaggle site, we started a project to analyze the data for various processes. Python is the major language used in the project. It was chosen largely because of Python's extensive use in data science applications and its ease of grasp. Python is a major application used to do data cleansing, preprocessing, and modeling. Python works particularly well for data analysis. It has various built-in data structures for storing data, such as lists and dictionaries. It also includes numerous powerful libraries for dealing with data, such as NumPy and pandas.

3.3 Technology Stack

This section provides an insight into the technological framework underpinning the development of this project, specifically tailored for market basket analysis. The project relies on a versatile array of libraries, including Pandas, NumPy, Seaborn, Matplotlib, among others. Moreover, it is executed within the Jupyter Notebook environment, offering several advantages and functionalities:

**Jupyter Notebook and Jupyter Lab**: Jupyter Notebook, renowned for its capabilities in creating computational notebooks, proves invaluable in this project. The utilization of Jupyter Lab enhances the project further by providing features like terminals, file viewers, and custom components, thereby fostering a more robust and streamlined development environment.

**Pandas**: The project harnesses the power of Pandas for proficient data analysis and manipulation, especially when dealing with Instacart data. Its extensive functionality facilitates the exploration and transformation of data, a fundamental aspect of market basket analysis.

**NumPy**: NumPy plays a pivotal role in this project, primarily as the engine for numerous data operations and calculations. Its numerical computing capabilities are leveraged to ensure efficient and accurate computations, essential for deriving meaningful insights from the data.

**Matplotlib & Seaborn**: In the realm of data visualization, Matplotlib and Seaborn emerge as indispensable tools. These libraries encourage the project to create informative and visually appealing graphs, charts, and plots, allowing for a comprehensive representation of the analyzed data. Visualizations not only aid in data exploration but also facilitate the effective communication of findings.

This comprehensive technology stack forms the backbone of the project, providing the necessary tools and capabilities to conduct robust market basket analysis and derive actionable insights from Instacart data. It enables efficient data handling, analysis, visualization, and, if required, machine learning-driven predictions, all contributing to the project's success and its ability to address complex business challenges effectively.

3.4 Dataset Description

A total of five datasets are extracted for the analysis. Datasets were derived from an open Kaggle competition.

A diagram of a product

Description automatically generated

Figure 2: Entity Relationship Diagram of Dataset

Dataset 1: Aisles

Data provided here includes information about aisles, such as aisle names and aisle IDs, and how products were organized within them.

|  |  |
| --- | --- |
| Variable | Description |
| aisle\_id | Aisle Identification Number |
| aisle | The name of the aisle in the retail store |

*Table 3 : Description of Aisles data set*

A black text on a white background

Description automatically generated

Figure 3: Shape of Aisles dataset

Dataset 2: Department

This data set provides information on the department such as department names and department ID.

|  |  |
| --- | --- |
| Variable | Description |
| department\_id | department Identification Number |
| department | Name of the department in retail store |

*Table 4 : Description of Department data set*

A black text on a white background

Description automatically generated

Figure 4: Shape of Department dataset

Dataset 3: Order\_Products\_prior/Order\_Products\_train:

In this dataset, all order details for any prior orders are included, including information on orders, products, and reordered items.

Order\_Products\_train is the same as order\_products\_prior and it is trained dataset.

|  |  |
| --- | --- |
| Variable | Description |
| order\_id | Order Identification Number |
| product\_id | Product Identification Number |
| add\_to\_cart\_order | The Order of the product added to cart |
| reordered | 1 ordered by user in past else 0 |

*Table 5 : Description of Order\_Products\_prior/Order\_Products\_train data set*

A black text on a white background

Description automatically generated

Figure 5: Shape of Order\_Products\_prior/Order\_Products\_train dataset

Dataset 4: Orders:

This dataset has information about customer orders like order ID, order number, weekday of the order, an hour of the order, user ID, and days since the prior order.

|  |  |
| --- | --- |
| Variable | Description |
| order\_id | Order Identification Number |
| user\_id | Customer Identification Number |
| eval\_set | Which evaluation set this order belongs in(prior/train/test) |
| order\_number | Sequence of the order placed by user |
| order\_dow | The day of week the order was placed |
| orders\_hours\_of\_day | The hour of the day the order was placed |
| day\_since\_prior\_order | Number of days since last order(NAs for order\_number = 1) |

*Table 6 : Description of Orders data set*

A black text on a white background

Description automatically generated

Figure 6: Shape of Orders dataset

Dataset 5: Products:

This dataset gives information on the products which were sold to customers such as product name, product ID, aisle, and departments.

|  |  |
| --- | --- |
| Variable | Description |
| product\_id | Product Identification Number |
| product\_name | Name of the product purchased by customer |
| aisle\_id | aisle Identification Number |
| department\_id | department Identification Number |

*Table 7 : Description of products data set*

A black text on a white background

Description automatically generated

Figure 7: Shape of products dataset

3.5 Data Pre-Processing

3.5.1 Handling Missing Values:

Except for in orders, there are no missing values in the dataset.csv. As we can see, just 6% of the entries in the days\_since\_prior\_order column are missing. In addition, there are 6% more unique users (206209) than total data in orders.csv.A screenshot of a number

Description automatically generated

Figure 8: Missing values details in dataset

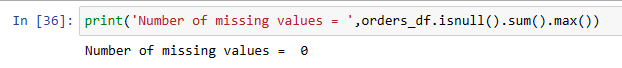
It can be observed that the days\_since\_prior\_order for every user's first order (order\_number = 1) is Nan, which makes sense. We may substitute 0 here.

Figure 9: Removal of Missing values

3.5.2 Merge Data

To generate a primary dataset for EDA, we will integrate all order\_products\_\_prior.csv, order\_products\_\_train.csv, products.csv, orders.csv (just train and prior set), aisle.csv, and departments.csv.

The final dataset includes the following columns and is considered for exploratory data analysis:

A screenshot of a computer

Description automatically generated

Figure 10: Columns of Final Dataset

3.6 Exploratory Data Analysis:

Peak hours of the day

A colorful bars of different colors

Description automatically generated with medium confidence

Figure 11: Peak Hours of the day

The most common time that customers like to place orders is shown in the above figure. The figure displays the hours in military time for analysis. The research shows that during regular business hours, from 9:00 am to 5:00 pm, are the busiest times for shopping. Before 9:00 am and after 5:00 pm, there are not many shopping hours where we may presume people who need certain things are buying them.

Peak days of the Week

A graph of different colored rectangles

Description automatically generated

Figure 12: Peak days of the Week

Sunday had the most sales, followed by Monday. Customers might buy for their weekly shopping on weekends. It makes it logical that the majority of orders are taken on Sunday and Monday because it is the end/beginning of the week and your food may have run out. The other days of the week will not alter significantly. For example, the difference between Wednesdays and Thursdays is not particularly important.

A colorful squares with white and pink squares

Description automatically generated with medium confidence

Figure 13: Hours of the Day vs Day of the Week Heatmap

The above figure shows the correlation between days of the week and hours of day using a heat map. From the heatmap, we can see that Sunday afternoon and Monday morning are prime times to order.

A graph of different colored bars

Description automatically generated

Figure 14: Days Since Prior Instacart Order

* It is interesting to see that there were quite a number of orders within the first 3 days.
* The 7-day mark makes the most sense since people usually order my groceries every 7 days.
* Also, we can see mini weekly peaks on days 14, 21,and 28 indicating pattern of people’s weekly buying habit.
* Day 30 has a spike, and this is probably a limitation of the data clumping day 30 as all prior orders greater than 30 days.

How Time affects the purchasing behavior of customers?

* Most orders are ordered on Sunday and Monday.
* Most orders are placed between 9:00 a.m. and 4:00 p.m.
* The busiest times for orders are Sunday afternoon (1:00PM) and Monday morning (10:00AM).
* By more than 65%, People usually buy previously ordered products from 6:00AM to 8:00AM
* Most users make orders after a week from their last order. or from a month after their last order.
* After a week from the last order, the probability of reordering within the same month is small.
* Send reminders of the most likely ordered products within a week, to catch the high prob of a customer to make their next order.
* The Next order has higher probability to be during 10 days from the current order.

A blue and white rectangle with black border

Description automatically generated

Figure 15: Top 15 most sold products

The above figure shows the top 15 most popular products ordered by the majority of customers. Fruits and vegetables are among the top 15 selling products, according to this list. Additionally, we can see that the most popular among them are organic goods.

5 Most Ordered Products

* Bag of Organic Bananas
* Banana
* Organic Baby Spinach
* Organic Avocado
* Organic Hass Avocado

14% of all orders contained Bag of Organic bananas.

Organic products are frequently ordered.

*Analyzing Organic Products*

* 10% of Instacart’s products are organic products.
* 31.5% of bought products are organic products.
* 67% probability of reordering an organic product.
* 61% probability of reordering a non-organic product.
* No significant pattern of when organic products are bought most, then when products in general are bought most.

A blue and black bar graph

Description automatically generated

Figure 16: Top 15 most ordered products from Aisles

The top 15 most popular aisles are fresh fruits, fresh vegetables, packaged vegetables, and fruit, followed by yogurt and Milk.

A white and blue rectangle with black border

Description automatically generated

Figure 17: Top 15 most ordered products from Departments

The above figure shows from which department customers ordered the products. According to the above figure, the biggest rate of ordering is for Produce. The second-place category, which likewise shows the same order ratio, is dairy eggs. The department with the lowest reorder ratio is babies.

A circular chart with different colored circles

Description automatically generated

Figure 18: Pie Chart – Percentage of products in each Department

The Pie-chart shows the percentage of products in each department. And we can see that personal care and snacks are the top departments with the most percentage of products both being equal to 13% and both constituting more than a quarter percentage of products. And pantry follows it by 11%. With Bulk department with 0% products.

Strategy for Basket Analysis

4.1 Apriori Algorithm

The Apriori technique, introduced in 1994 by Agrawal and Srikant, constitutes a foundational approach for identifying frequent item sets in Boolean association rules. At the core of Apriori lies the principle that "if an itemset is frequent, then all of its subset items will be frequent." An itemset attains the status of "frequent" when its support, surpassing a predefined threshold, demonstrates its occurrence within the dataset. The technique derives its name from the concept of "prior," emphasizing the anticipation of items that transition from one stage to the next, thus reflecting its historical context. Within the realm of data mining, the Apriori algorithm is instrumental in the discovery of association rules, particularly within the domain of affinity analysis or market basket analysis, which elucidates relationships among diverse attributes.

To illustrate this principle, consider an example: if the itemset {b, d, e} from the dataset qualifies as a frequent itemset, denoted by a support measure of 0.35 exceeding the minimum support threshold of 0.25, it implies that all its subsets, such as {b, d, e}, {b, e}, and {d, e}, must also be frequent item sets. Therefore, if {b, d, e} is deemed frequent, its constituent subsets must necessarily be frequent as well. Conversely, if certain item sets, say {a, b}, prove to be infrequent, it logically follows that their supersets must exhibit substantially higher support. Consequently, the entire subset encompassing {a, b} can be promptly eliminated. This pruning process, rooted in support-based pruning, adheres to a linear direction pruning mechanism hinged upon the support measure. The primary objective of this support measure is to facilitate such pruning, embodying the antimonotone property of the support measure.

The Apriori algorithm comprises two essential steps:

Join and Prune:

* Generate all frequent item sets: A frequent itemset is characterized by a transaction support exceeding the minimum threshold.
* Generate confident association rules from frequent item sets: A confident association rule is one in which the confidence level surpasses the minimal confidence threshold.

The Apriori library's Apriori class is employed to implement the Apriori algorithm on the Instacart dataset. In this context:

* A k-itemset is defined as an itemset containing k elements.
* Lk signifies recurring sets of k elements.
* Ck represents frequently occurring collections of candidate items with k members.

The Apriori function is adept at minimizing the number of items to be scrutinized when searching for frequent item groupings, thereby enhancing the efficiency of the process. This approach plays a pivotal role in uncovering valuable insights within datasets.

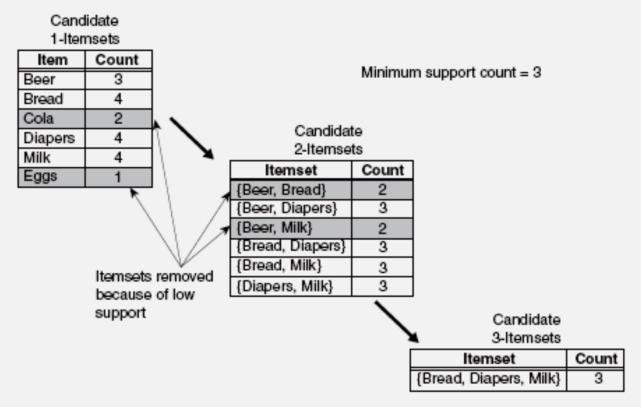


Figure 19: Apriori Algorithm example

This algorithm employs an iterative approach to discover 2-item sets from 1-item sets and 3-item sets from 2-item sets. This process can be extended to identify frequent item candidates for k elements by leveraging frequent item sets with (k-1) elements.

In the initial phase, each item is considered as a 1-point contender. Subsequently, their support is tallied, and any itemset with a support count below the minimum threshold is removed from consideration. Consequently, items like Cola and Eggs no longer meet the support criteria. This aligns with the Apriori principle, which necessitates that all supersets of infrequent 1-item sets be considered infrequent. In the next iteration, candidate 2-item sets are formed with the assistance of frequent 1-item sets.

Given that there are only four frequent 1-item sets, the process generates (4C2) = 6 candidate 2-item sets. The efficacy of the pruning strategy can be demonstrated by counting the candidate item sets generated. Employing a brute-force approach to list all item sets (up to length 3) as contenders would result in 41 outcomes.

(6C1) + (6C2) + (6C3) = 6 + 15 + 20 = 41

However, when applying the Apriori principle, this number reduces to 13.

(6C1) + (4C2) + 1 = 6 + 6 + 1 = 13

Even in a basic scenario, this leads to a significant 68 percent reduction in the number of candidate item sets.

The pseudo code provided outlines the process for creating frequent item sets using the Apriori algorithm. Let Ck denote the candidate set of k-item sets, and Fk represent the frequent set of k-item sets.

4.2 Frequent Pattern Growth Algorithm

The FP-Growth technique introduces an innovative approach to quantify frequent item sets by employing an FP-Tree, a condensed representation of transaction records resembling a graph structure. In essence, FP-Tree transforms datasets into a graphical format. Unlike the Apriori algorithm, which relies on a generate-and-check methodology, FP-Growth takes a different route by first constructing the FP-Tree and then utilizing this streamlined tree structure to derive regular item sets. The effectiveness of the FP-Growth method hinges on its ability to compress data efficiently during the FP-Tree construction phase.

The FP-Growth approach streamlines the process of repeatedly searching for minors and subsequently combining suffixes to generate specific, broad models. By leveraging partially recurring elements as suffixes, this approach achieves a notably prominent level of efficiency, substantially reducing search costs.

FP-Tree Representation

An FP-Tree represents a compact data structure that encapsulates a set of tree-like records. Each transaction is traversed and organized along an FP-Tree path, and this process continues until all transactions have been processed. As the paths overlap through different transactions sharing similar subsets, the resulting tree structure remains concise.

The fundamental execution steps of the FP-Growth algorithm are as follows:

Database Scanning: Initially, the database is scanned to identify items that meet or exceed a predefined threshold value.

Descending Support Values: These identified items are then ranked in descending order based on their support values, starting from the most frequent (largest) and progressing to the least frequent (smallest).

Tree Construction: The FP-Tree begins with just a root node. As each item is processed, the tree expands accordingly.

Database Rescanning: For each transaction in the database, a secondary scan is performed to further populate the FP-Tree.

In summary, the FP-Growth technique revolutionizes the way frequent item sets are extracted, optimizing the process through efficient FP-Tree construction. This method minimizes search complexity and delivers valuable insights by harnessing the power of data compression and graphical representation.

A diagram of lines and dots

Description automatically generated

Figure 20: FP Growth Algorithm example

The illustration illustrates a dataset consisting of 5 transactions and 5 distinct items. As the first three transactions are processed, the figure gradually unveils the emergence of the FP tree structures. Within this tree, each node bears an item label and a counter denoting the number of transactions that have followed the defined path. The subsequent explanation elucidates the stepwise generation of the FP-tree:

Initial Item Analysis: The initial step involves a meticulous examination of the data to ascertain the support value for each item. Items that are infrequently utilized are removed from consideration, while those with high usage frequency are arranged in descending order. In the depicted graph, item 'a' emerges as the most frequently employed, followed by 'c,' 'd,' and 'e.'

First Data Traversal: Subsequently, the program embarks on a second pass through the data to construct the FP-tree structure. Upon encountering the initial transaction 'a, b,' nodes 'a' and 'b' are created. These nodes are then connected in a hierarchical tree structure, forming the path 'root -> a -> b.' Each node in the tree is now associated with a count value.

Node Expansion: As the second transaction traverses 'b, c, d,' fresh nodes representing 'b,' 'c,' and 'd' are generated. These nodes are then interconnected to form a path 'root -> b -> c -> d.' It's important to note that the first two transactions include 'b,' while subsequent transactions with distinct antecedents do not.

Common Ancestor Handling: In the third transaction 'a, c, d, e,' a situation may arise where a common ancestor had been transacted previously. The path continues to overlap as long as the item antecedents match, resulting in a path like 'root -> a -> c -> d' for this specific transaction.

Repeat Data Processing: This process is reiterated for each transaction in the dataset, diligently constructing the FP-tree while considering the shared item antecedents and expanding the tree accordingly.

In summary, the FP-tree construction method involves a multi-step process, starting with initial item analysis, followed by data traversal to construct the tree structure. This process iterates through all the transactions in the dataset, creating a structured representation that can be used for efficient association rule mining.

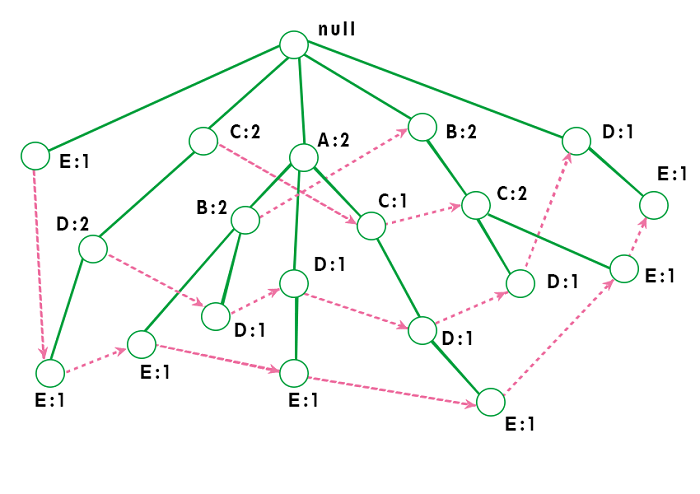


Figure 21 : Frequent Pattern Tree Structure

4.3 Eclat Algorithm

The Eclat Algorithm, akin to the Apriori Algorithm, is a fundamental technique in the realm of association rule mining. Proposed as a powerful tool for discovering frequent item sets, Eclat stands as an abbreviation for "Equivalence Class Clustering and bottom-up Lattice Traversal." It was developed as an alternative approach to identifying item sets with support values exceeding a defined threshold. Much like Apriori, Eclat's core principle is that if an itemset is frequent, all of its subsets are also frequent.

Eclat's journey into frequent itemset discovery begins by employing a vertical approach, primarily based on a compact data structure known as the "tidset." Each item is associated with a tidset, which stores the transaction identifiers where that item appears. Through the intersection of tidsets, Eclat efficiently explores and identifies frequent item sets.

The primary advantage of Eclat over Apriori lies in its ability to save memory and computational resources by avoiding candidate generation. This is particularly beneficial when dealing with large datasets, as the elimination of candidate generation significantly reduces the algorithm's time and space complexity.

A flowchart of a program

Description automatically generated

Figure 22: ECLAT Algorithm Flow chart

Eclat's core steps encompass:

Vertical Transaction Database: Eclat starts with a vertical representation of the transaction database, where each column represents an item, and rows contain transaction identifiers where these items occur.

Finding Frequent 1-Item sets: Eclat initiates by identifying frequent 1-item sets by counting the occurrences of each item's tidset and comparing them against the minimum support threshold.

Building Equivalence Classes: Equivalence classes are formed by merging tidsets of items that share common transaction identifiers. This step is essential for efficient frequent itemset generation.

Generating Frequent Item sets: Eclat generates frequent item sets by recursively combining equivalence classes and applying the minimum support threshold to prune infrequent item sets.

The elegance of Eclat lies in its ability to perform these operations efficiently, ultimately unveiling a concise set of frequent item sets. This makes it particularly valuable for association rule mining, market basket analysis, and other data-driven tasks where identifying frequent patterns is crucial.

In summary, the Eclat Algorithm, with its innovative tidset-based approach, offers a potent alternative to the Apriori Algorithm for discovering frequent item sets in large datasets, streamlining the process and conserving valuable computing resources.

4.4 Recommender System

Association rule mining plays a pivotal role in the context of this thesis research. The development of the Association Rules for Collaborative Recommenders Framework algorithm draws inspiration from the Apriori algorithm. To ensure the generation of a sufficient number of key rules for product recommendation, the algorithm necessitates customization by setting an appropriate minimum support threshold during the mining process.

Our methodology primarily focuses on mining rules pertaining to individual target customers or specific articles. The rationale behind this approach lies in our goal of suggesting items that align with the preferences of a target customer. Consequently, we exclusively consider rules with a [antecedent: consequent] structure in the head rule and conduct online mining of user associations. As a result, the system's performance becomes of paramount importance.

The guiding principles governing association rule mining for recommendation systems are as follows:

Given a dataset of transaction records, a designated target item, a predefined minimum confidence threshold, and the desired number of rules, the objective is to identify association rules with the highest possible support in the head, while still adhering to the minimum confidence criterion.

Instead of imposing a minimum support requirement, our approach seeks to establish a high minimum confidence threshold and a range for the number of rules prior to commencing the mining process. This decision is informed by several factors:

Balancing Confidence and Support: Confidence quantifies the degree of similarity between items, while support reflects the frequency of those similarities. Both factors are crucial for making meaningful recommendations. Greater confidence and support lead to more reliable suggestions. However, setting excessively high thresholds risks generating an insufficient number of rules for accurate recommendations, whereas overly lenient thresholds may result in impractically long runtimes due to the considerable number of rules from frequent items.

Variability in User Tastes: User preferences and product popularity exhibit significant variability. Consequently, determining universally applicable minimum confidence and support levels prior to mining is challenging. The ideal approach involves identifying a high minimum confidence level and a flexible set of criteria that allow the system to establish an acceptable minimum requirement tailored to individual users or content.

In our proposed recommendation system, we limit the construction of association rules to those with a single item in the consequent. This decision is motivated by the following considerations:

Rule Abundance: Association rule mining often yields an excessive number of rules, even when restricting them to having only one item in the consequent. This abundance increases computational complexity due to the expanding item set. Moreover, more complex rules tend to provide marginal insights into the dataset.

Simplicity and Redundancy: Simplifying rules by focusing on those with a single item in the consequent allows us to avoid redundancy. Specifically, for rules with multiple items in the consequent, simpler rules with the same antecedent but a single item from the complex rule's consequent should also be considered. All these rules must be present in the findings, as their support and confidence cannot be lower than that of the more intricate rule. In essence, if a rule X Y -> S T is considered, rules like X Y -> S and X Y -> T should also be part of the results.

This approach ensures that the recommendation system remains effective while managing the complexity of association rule mining, ultimately contributing to more meaningful insights and improved system performance.

Implementation

5.1 Apriori Algorithm

The Apriori function from the mlxtend.frequent patterns library is used to apply the Apriori algorithm to the Instacart dataset. The apriori algorithm requires data to be in one hot encoded format before it can determine which items in a particular dataset are most popular.

A table with black text

Description automatically generated with medium confidence

Figure 23: One-Hot Encoded dataset

Apriori Algorithm Python Generator: A Python generator is an iterator that generates values on-the-fly, allowing you to iterate over a sequence without storing all the values in memory at once. It is memory-efficient and useful for handling large datasets or when accessing values incrementally.

Image mlxtend.apriori: The apriori function from mlxtend.frequent\_patterns is a powerful tool for market basket analysis. It implements the Apriori algorithm and provides a simple interface for discovering frequent item sets in transactional datasets. With customizable parameters and integration with the MLxtend library, apriori offers ease of use, flexibility, and reliable performance for market basket analysis tasks.

How to choose the min support?

Minimum Support: refers to the minimum frequency or occurrence threshold that an itemset or item must meet to be considered significant in a dataset. It is a parameter used in frequent itemset mining and association rule learning algorithms, determining the minimum level of popularity required for an itemset to be considered frequent. min\_support = (support\_count of itemset) / (total number of transactions)

A screenshot of a computer screen

Description automatically generated

Figure 24: Minimum Support Items of Apriori data

This project used the association rules function from the mlxtend.frequent patterns library to determine the correlation between the various products using a frequent itemset generated via the apriori technique. The rules generation based on the lift metrics with a minimum threshold value of 1.

A screenshot of a computer

Description automatically generated

Figure 25: Correlation between Items of apriori data

The table produced by the association rule mining contain various output parameter as follows:

▪ Antecedents: This phrase refers to the first product that is supposed to be sold first.

▪ Consequents: The next product that is anticipated to be sold after the first product is referenced in this term.

▪ Antecedent support: The probability of observing the first product referenced by this term.

▪ Consequent support: The probability of observing the second product referenced by this term.

▪ Support: The probability of observing the first and second product together.

▪ Confidence: The probability of observing the next product when first product sold.

▪ Lift: This term refers that when the first product sold the probability of selling the next product increase by a factor of lift.

5.2 FP-Growth Algorithm

The fpgrowth function from the mlxtend.frequent patterns library is used to apply the fpgrowth algorithm to data set which transformed into one hot encoder in previous section.

A screenshot of a computer

Description automatically generated

Figure 26: Minimum Support Items of FP Growth data

A screenshot of a computer

Description automatically generated

Figure 27: Correlation between Items of FP Growth data

 Results are different than apriori.

* The output of FP-Growth and Apriori algorithms can differ due to their underlying principles and techniques:
  + FP-Growth uses an efficient tree-based data structure called FP-Tree and does not generate candidate item sets explicitly.
  + Apriori generates candidate item sets and requires multiple passes over the transaction data.
* FP-Growth is generally faster and more suitable for large datasets compared to Apriori.
* The differences in approach can lead to variations in the frequent item sets discovered and their corresponding support values.
* Consider the specific requirements, dataset size, and computational resources when choosing between FP-Growth and Apriori for market basket analysis.

5.3 Eclat Algorithm

ECLAT (Equivalence Class Clustering and Bottom-up Lattice Traversal) is an efficient algorithm for frequent itemset mining in transactional datasets. It works by exploiting the vertical data format, where transactions are represented as sets of items.

A screenshot of a menu

Description automatically generated

Figure 28: One-Hot Encoding for ECLAT data

The steps involved in the ECLAT algorithm are as follows:

Vertical Representation:

Convert the transactional dataset into a vertical representation, where each item is associated with the set of transactions in which it appears.

Initialization:

Initialize an empty set called the "prefix" to store the current frequent item sets.

Recursive Process:

* Start with a single item as the prefix and count its support by scanning the database.
* If the support of the prefix is above a specified minimum support threshold, it is considered a frequent itemset.
* Add the frequent item set to the list of frequent item sets.
* Generate a new prefix by combining the frequent itemset with the remaining items.
* Recursively repeat the process with the new prefix until no more frequent item sets can be generated.

Combining Frequent Item sets:

Combine the frequent item sets obtained in the previous step to form larger item sets. This is done by finding the intersections of their transactions.

A screenshot of a menu

Description automatically generated

Figure 29: Combining frequent item sets.

Repeat:

Repeat the recursive process with the combined frequent item sets to find larger item sets until no more frequent item sets can be generated.

Termination:

Stop the process when no more frequent item sets can be found.

Output:

Return the list of all frequent item sets that meet the minimum support threshold.

However, it may perform poorly on sparse datasets or datasets with long transactions.

5.4 Association Results:

A graph with a blue line

Description automatically generated

Figure 30: Confidence threshold chart

Firstly, to find the association rules between products a reasonable confidence needed to be chosen. For this as shown in the above figure, a graph was plotted between the association rule length versus the confidence value. To be able to compare without having a lot of association rules a confidence threshold of 0.1was chosen.

5.5 Comparing performance and Accuracy of Association rules.

5.1 Analysis on Apriori algorithm

In this section, we will initiate our exploration by delving into the Apriori technique, a cornerstone of performance analysis in our comparative study. Apriori stands out as the preeminent and phenomenally successful algorithm in the realm of item set mining. Its fundamental premise revolves around conducting multiple iterations over datasets or databases containing transactions or data.

At the core of the Apriori algorithm lies the Apriori property, which stipulates the necessity of a consistent accumulation of non-empty item sets. This inherent property underpins the algorithm's functionality. Additionally, our investigation has unveiled a distinct characteristic elucidating why any superset fails to pass the minimal support test when the system itself does not meet this crucial criterion.

The Apriori algorithm strategically employs the Breadth First Search (BFS) approach, leveraging the downstream locking trait. This trait dictates that any superset of an item set failing to meet the minimum support requirement is inherently deemed unfit.

In practice, the transaction database is often structured horizontally, with each transaction recording the frequency of item sets. This layout facilitates efficient processing and analysis, allowing Apriori to uncover valuable patterns and associations within the data.

5.5.2 Analysis on FP-Growth algorithm

Regarding FP-Growth, this algorithm employs a strategy rooted in divide and conquer, utilizing the FP data structure to streamline the representation of transactional databases. Unlike traditional approaches that rely on standard item sets, FP-Growth focuses on mining frequent patterns using the FP tree.

In the initial stages of FP-Growth, a list is meticulously arranged in descending order of support. Each item in this list is associated with a structure referred to as a "node." These nodes, apart from the root node, contain the item name, the corresponding support count, and a link to a tree node featuring a similar item name. These nodes play a crucial role in facilitating the growth of the FP tree.

Throughout the construction process of the FP tree, common prefixes can be rearranged, ensuring that the root-to-leaf node paths consistently maintain their sequence. Once the FP tree is fully constructed, it becomes a valuable resource for extracting frequent patterns, with the retrieval process commencing from the leaf nodes.

One of the key advantages of FP-Growth lies in its efficient memory utilization and storage due to its deliberate structural layout. It accomplishes this through two pivotal operations:

Identification of Common FP Tree Elements: FP-Growth excels in identifying common elements within the FP tree, a critical step in pattern mining. This capability allows it to efficiently discover recurring patterns within the dataset.

Compact FP Tree Data Structure: By employing a compact FP tree data structure, FP-Growth optimizes storage and memory utilization. This structural design ensures that the algorithm operates resourcefully while delivering accurate results.

In summary, FP-Growth distinguishes itself through its innovative approach to pattern mining, leveraging the FP tree structure and memory-efficient strategies. This algorithm excels in identifying common patterns within datasets, making it a valuable tool for data analysis and knowledge discovery.

5.5.3 Analysis on ECLAT Algorithm

The ECLAT algorithm, short for Equivalence Class Clustering and Bottom-Up Lattice Traversal, stands as a robust contender in the realm of frequent item set mining. Unlike its counterparts, Apriori and FP-Growth, ECLAT employs a distinctive approach that hinges on efficiency and unique data structuring. Central to its methodology is the Equivalence Class Clustering technique, which groups data into equivalence classes based on shared items within transactions. This clustering strategy drastically reduces computational overhead by allowing ECLAT to concentrate on specific subsets of transactions characterized by common items.

ECLAT adopts a vertical representation of the transactional database, storing data as vertical bit vectors for each distinct item. This streamlined format enhances data retrieval and manipulation, making ECLAT a preferred choice for datasets with numerous transactions and items. Its mining process involves a bottom-up traversal of the itemset lattice, systematically exploring combinations of items, starting with individual elements, and progressing to larger sets. Throughout this traversal, ECLAT employs depth-first search (DFS) to optimize the identification of frequent item sets efficiently. Renowned for its efficiency and scalability, ECLAT excels in analyzing vast transactional datasets and those with high dimensionality. Its applications span diverse domains, including market basket analysis, recommendation systems, and bioinformatics, where its capability to efficiently mine frequent item sets proves invaluable for revealing meaningful data patterns and associations.

5.6 Comparing the Time Complexity between algorithms:

To run these algorithms on a dataset of 131,210 transactions with a minimum support of 0.01. Figure below shows how many seconds each algorithm took to run on this dataset. Notice that Eclat took 1,688 seconds, which is 48 times longer than Apriori (34 seconds) and 168 times longer than FP-growth (10 seconds). Therefore, we had reduced the size of the dataset to 500 transactions for Eclat. This result was unexpected, especially for Eclat. One probable reason is that the dataset has a high-density transaction length, which makes the set intersection operation more costly for Eclat algorithm.

A graph with blue and white bars

Description automatically generated

Figure 31: Bar chart- Time consumptions by different algorithm

1. FP - Growth is the fastest.
2. FP-growth and apriori lead to the same average lift, support, confidence.
3. The eclat algorithm is extremely slow. 10 transactions take 10 times the time of all the transactions compared to apriori or fp-growth.

5.7 Accuracy by comparing the support, lift, confidence, conviction, and leverage.

5.7.1 Support

Support measures the frequency of occurrence of an itemset in the dataset. It indicates how popular or common an itemset is within the transactions. (Market Analysis, n.d.)

A graph of blue and orange lines

Description automatically generated

Figure 32: Support comparison between Apriori and FP-Growth

The line graph illustrates how the accuracy or value changes as you adjust the minimum support threshold. Each line on the graph represents a different algorithm (in this case, Apriori and FP-Growth).

If you notice that one line consistently stays above the other across different support thresholds, it suggests that the algorithm associated with that line is generally more accurate or valuable for the given dataset.

Conversely, if the lines intersect or cross at various support thresholds, it indicates that the algorithm's performance depends on the choice of support threshold. In such cases, you may need to choose the algorithm that aligns better with your specific goals and constraints.

Interpretation: To draw meaningful conclusions, observe how the lines fluctuate with changes in the minimum support threshold. This comparison helps you decide which algorithm is more suitable for your dataset and objectives.

If one algorithm consistently outperforms the other (i.e., it has higher accuracy or value) across different support thresholds, you may prefer that algorithm.

If the lines cross or have different patterns at different support thresholds, consider your specific goals and constraints. For example, if you need faster results, you might choose the algorithm that performs better at lower support thresholds, even if its accuracy is slightly lower.

5.7.2 Lift

Lift measures the strength of association between the antecedent and consequent of a rule, taking into account the support of the rule and the individual supports of the antecedent and consequent. (Yoder, Vandenberg, & Breimer, 2011)

A graph of a graph with blue and orange lines

Description automatically generated

Figure 33: Lift comparison between Apriori and FP-Growth

Lift helps Apriori filter out rules that are not statistically significant. Higher lift values in Apriori indicate stronger and more interesting associations between items in terms of their co-occurrence.

Apriori tends to generate more rules, and lift is often used to prioritize and select the most meaningful rules among them.

Similar to Apriori, higher lift values in FP-Growth indicate stronger associations between items in the discovered rules.

FP-Growth is known for its speed and ability to manage large datasets efficiently, making it a valuable choice for association rule mining.

A high lift value suggests that the two items (antecedent and consequent) tend to appear together more frequently than expected by chance, indicating a strong association between them.

5.7.3 Confidence

Confidence measures the reliability or certainty of a rule. It represents the conditional probability of finding the consequent given the antecedent. (Cavique, 2007)

A graph of a graph with blue and orange lines

Description automatically generated

Figure 34: Confidence comparison between Apriori and FP-Growth

Confidence values range from 0 to 1, with higher values indicating stronger confidence. A confidence value of 1 means that whenever item A is present in a transaction, item B is also present (perfect association).

Confidence helps Apriori filter and rank rules based on how likely the consequent (item B) is to occur when the antecedent (item A) is present in a transaction.

Apriori can generate a large number of rules, and confidence is often used to select and prioritize the most confident rules for further analysis or recommendation.

Confidence in FP-Growth evaluates the conditional probability of the consequent (item B) given the antecedent (item A).

Higher confidence values in FP-Growth indicate that the presence of item A is a strong indicator of the presence of item B, leading to more confident rules.

Confidence measures the conditional probability of observing the consequent when the antecedent is present, making it a valuable metric for understanding the strength of item associations in transactional data. High confidence values signify strong associations, while low confidence values may suggest weaker or less reliable relationships between items.

5.7.4 Conviction

Conviction measures the implication strength of a rule by comparing the expected and observed support of the consequent. It indicates the extent to which the rule is dependent on the antecedent. (Kamakura, 2012)

A graph of a comparison of convicted criminals

Description automatically generated with medium confidence

Figure 35: Conviction comparison between Apriori and FP-Growth

Conviction values can range from 0 to infinity. A conviction value of 1 indicates that the antecedent and consequent are independent (no predictive power), values greater than one suggest a positive association, and values less than 1 suggest a negative association.

Conviction is less commonly used in the context of the Apriori algorithm. While it is a valuable metric for assessing rule strength, it is not as frequently employed as support and confidence.

You can calculate conviction for rules generated by Apriori, but it may not be the first metric considered when evaluating rule quality.

Conviction can be used in FP-Growth to assess rule strength and interestingness. It helps identify rules that are statistically significant and have practical relevance.

FP-Growth is known for its efficiency, and when you want to evaluate a large number of rules, conviction can be a valuable metric for selecting the most interesting ones.

Conviction is a metric that complements support and confidence in assessing association rules. It provides additional insights into the independence of antecedent and consequent, helping to identify rules that are genuinely interesting and not merely a result of random chance. The choice of whether to use conviction alongside support and confidence depends on your specific analysis goals and the algorithms you employ.

5.7.5 Leverage

Leverage measures the difference between the observed and expected support of the rule. It indicates how much the occurrence of the antecedent and consequent together deviates from what would be expected if they were independent. (Bramer, 2013)

A graph with blue and orange lines

Description automatically generated

Figure 36: Leverage comparison between Apriori and FP-Growth

Leverage values can range from -1 to 1. A leverage value of 0 indicates that the antecedent (item A) and consequent (item B) are independent, while positive values suggest a positive association (more co-occurrence than expected), and negative values suggest a negative association (less co-occurrence than expected).

Leverage can be used in Apriori to assess the co-occurrence of items and identify rules with interesting patterns of co-occurrence.

Leverage helps in the identification of rules that represent meaningful associations beyond what would be expected by chance.

It can be used alongside support and confidence to evaluate the significance of association rules generated by Apriori.

Leverage in FP-Growth is used to assess the co-occurrence patterns between items and identify rules with strong associations.

FP-Growth is known for its efficiency, and leveraging this metric can help prioritize interesting rules when dealing with large datasets.

Leverage complements support and confidence in assessing the strength of association rules discovered by FP-Growth.

Leverage is a metric that provides insights into the co-occurrence patterns of items and helps identify rules with interesting and non-random associations. It can be a valuable tool for assessing rule quality and identifying meaningful patterns in transactional data. Whether you use leverage alongside support and confidence depends on your specific analysis goals and the algorithms you employ.

5.8 Comparing Apriori, FPGrowth and ECLAT algorithm:

The choice of algorithm depends on the characteristics of your dataset and the trade-offs between memory usage and runtime. FP-Growth and ECLAT often outperform Apriori in terms of efficiency and speed, especially when dealing with large or sparse datasets. However, Apriori remains a valuable option when interpretability and ease of implementation are essential considerations.

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Apriori** | **FP-Growth** | **ECLAT** |
| **Algorithm Type** | Generate and test (Brute-force) | Divide and conquer (Efficient) | Vertical database representation (Efficient) |
| **Core Idea** | Candidate generation and pruning based on support | Tree-based structure and pattern growth | Vertical database representation and intersection |
| **Candidate Generation** | Generates candidates for each itemset size | Generates conditional FP-trees | Generates item sets using vertical format |
| **Pruning Strategy** | Uses Apriori property to prune infrequent item sets | Uses pattern growth and avoids candidate generation | Uses vertical representation for pruning |
| **Memory Usage** | High (Stores candidate item sets) | Low (Compact tree structure) | Low (Compact vertical format) |
| **Scalability** | Suffers from the "combinatorial explosion" problem as dataset size grows | Efficient for large datasets and high-dimensional data | Efficient for large datasets and sparse data |
| **Computation Speed** | Slower for large datasets and many items | Faster due to FP-tree structure | Faster due to vertical format |
| **Parallel Processing** | Typically, not designed for parallel processing | Parallelism can be leveraged for performance | Parallelism can be leveraged for performance |
| **Suitable for Sparse Data** | Less efficient for sparse datasets | Efficient for sparse datasets | Efficient for sparse datasets |
| **Implementation Availability** | Available in various libraries (e.g., Python's mlxtend, scikit-learn) | Available in various libraries (e.g., FP-Growth in PySpark) | Available in various libraries (e.g., Python's PyECLAT) |
| **Use Cases** | Market basket analysis, recommendation systems, and more | Market basket analysis, anomaly detection, and more | Market basket analysis, recommendation systems, and more |

*Tabel 8 : Comparison between Algorithms*

Conclusion

In conclusion, our meticulous examination of market basket data through various association rule mining algorithms has illuminated an intricate tapestry of item relationships and purchasing patterns within the dataset. This pursuit of knowledge has not only enhanced our understanding of consumer behavior but has also opened up possibilities for optimizing business strategies. Among the algorithms meticulously evaluated, FP-growth, standing tall as the most proficient contender, emerged as the beacon guiding us through this labyrinth of data-driven insights.

FP-Growth, an algorithm known for its efficiency and memory economy, proved its mettle in unraveling the intricate web of associations that govern item purchases. Its ability to generate frequent patterns directly from a compact data structure, the FP-Tree, highlighted its prowess in handling large datasets with finesse. By rapidly identifying item sets that met user-defined support thresholds, FP-Growth enabled us to pinpoint the most prevalent purchasing patterns. These patterns are the key to unlocking strategic decisions, from optimizing inventory management to personalized marketing.

On the other hand, while Eclat, another stalwart in the field of association rule mining, has carved a significant niche for itself, our analysis illuminated certain challenges it faced when dealing with the specific characteristics of our dataset. The sparsity inherent in our data seemed to pose an obstacle to Eclat's ability to extract meaningful associations between items. Unlike FP-Growth, which natively adapts to sparsity, Eclat seemed to grapple with the sparse nature of our dataset. This calls for further exploration into potential adaptations or enhancements that could tailor Eclat to perform optimally in similar data environments.

Furthermore, the absence of support for additional metrics such as lift, conviction, and leverage in Eclat limited our ability to compare it comprehensively with other algorithms, including Apriori and FP-Growth. These metrics play a pivotal role in assessing the strength and relevance of association rules, providing a more nuanced perspective on item relationships beyond mere frequency.

In the grand tapestry of association rule mining, FP-Growth has emerged as a robust choice, a dependable navigator in the sea of item sets, and a beacon of insight. Its ability to efficiently traverse through vast transaction datasets has equipped us with invaluable insights into item associations and purchasing patterns. By uncovering hidden connections, FP-Growth has illuminated pathways to improved business strategies, enhanced customer experiences, and optimized operational efficiencies.

Nevertheless, this journey should not signal the end but rather the beginning of a new chapter. The challenges that Eclat faced in our analysis provide fertile ground for further research. Future endeavors can explore ways to enhance Eclat's adaptability to sparse datasets and its compatibility with market basket analysis. Additionally, expanding the repertoire of evaluation metrics for association rule mining algorithms would facilitate more comprehensive and nuanced comparisons.

In summary, our exploration of association rule mining algorithms has not only expanded our horizons but has also posed intriguing questions that beckon further exploration. FP-Growth, our guiding star in this voyage, has illuminated the way forward in market basket analysis. Eclat, though facing its own set of challenges, offers a tantalizing prospect for improvement. As we set sail into uncharted waters of data analysis and algorithmic exploration, the lessons learned in this endeavor shall continue to guide us, providing valuable insights into the world of consumer behavior and data-driven decision-making.

Future Work

In the pursuit of advancing association rule mining and recommendations, our current study has laid the foundation for several promising avenues of future research. These directions are tailored to address specific academic and practical challenges, leveraging the unique characteristics and strengths of Apriori, FP-Growth, and ECLAT.

Algorithmic Diversity and Exploration

Our research predominantly focused on Apriori, FP-Growth, and ECLAT, but the field of association rule mining offers a multitude of algorithms, each designed to tackle distinct challenges and data characteristics. Future academic endeavors should encompass:

Apriori Variants: A deeper investigation into various Apriori variations, such as Apriori Hybrid and Apriori TID, to discern their applicability and performance in diverse scenarios.

Hybrid Approaches: Exploring the potential advantages of hybridizing multiple algorithms, including the fusion of Apriori, FP-Growth, and ECLAT. Hybrid models can harness the complementary strengths of each algorithm to address complex dataset challenges effectively.

Alternative Data Structures: Extending research to explore alternative data structures beyond FP-Tree, such as trie-based structures or vertical databases, to evaluate their suitability for association rule mining.

Content-Based Recommendations

While our research has excelled at uncovering item associations, future academic investigations can expand the realm of recommendations to encompass content-based strategies. In addition to collective information, content-based recommendations consider individual item attributes and customer preferences, thereby enriching the diversity and accuracy of suggestions:

Product Attribute Analysis: In-depth exploration of similarity score calculations between products based on attributes such as category, brand, or price. Recommendations can be generated by identifying products with similar attributes, enabling more nuanced and personalized suggestions.

Combined Hybrid Approach: The integration of association-based and content-based recommendations into a unified hybrid approach. This amalgamation can leverage the collective wisdom of association rules and the intricate knowledge of individual products to deliver more valuable recommendations.

Cross-Domain Application of Algorithms

The principles of association rule mining extend beyond the confines of the retail domain. Future academic research can delve into the adaptability and impact of these algorithms across various application areas:

Healthcare: The application of association rule mining to healthcare datasets for patient treatment recommendations, disease diagnosis, and the optimization of hospital operations.

Finance: Leveraging association rules to unveil patterns in financial transactions for purposes such as fraud detection, investment recommendations, and risk assessment.

Manufacturing: Implementing association rule mining within manufacturing processes to enhance supply chain management, quality control, and production optimization.

Scalability and Memory Optimization

In the era of big data, the scalability and memory efficiency of algorithms are of paramount importance. Future academic research should prioritize techniques to enhance scalability and memory optimization in association rule mining:

Distributed Computing: Exploring the potential of leveraging distributed computing frameworks such as Apache Spark to efficiently process vast datasets in parallel.

Data Pruning: Developing intelligent data pruning strategies aimed at reducing memory usage while preserving crucial information.

Algorithmic Performance Optimization and Parameter Tuning

Continuous algorithmic fine-tuning and optimization are essential for achieving peak efficiency. Future research should encompass:

Algorithmic Enhancements: A comprehensive exploration of avenues for optimizing Apriori, FP-Growth, and ECLAT algorithms to cater to specific use cases, datasets, and hardware architectures.

Parameter Tuning: In-depth parameter tuning experiments to discover the optimal configuration for each algorithm and data type.

Hybrid Algorithm Evaluation

As we amalgamate all three algorithms—Apriori, FP-Growth, and ECLAT—into our toolkit, future academic research should delve into the intricacies of their interaction. Rigorous evaluations of hybrid approaches that combine the strengths of these algorithms will be invaluable:

Algorithm Fusion: The exploration of combining the outputs of Apriori, FP-Growth, and ECLAT to generate association rules offering a more holistic view of item relationships.

Performance Benchmarking: Rigorous benchmarking to identify scenarios where a particular algorithm or hybrid approach excels, enabling the selection of the most appropriate method based on data characteristics.

Scaling to Very Large Databases

Association rule mining often operates on extensive datasets, where memory constraints become a critical concern. Future academic research should consider the scalability of our algorithms to exceptionally large databases:

Efficient Sampling: Development of efficient sampling strategies that maintain data representativeness while reducing computational load.

Parallel Processing: Exploration of techniques for parallelizing association rule mining tasks across multiple processors or distributed clusters.

In conclusion, the horizon of association rule mining is expansive, offering abundant opportunities for future academic exploration. Our journey through Apriori, FP-Growth, and ECLAT has illuminated pathways toward more comprehensive and potent solutions. As we embark on these academic endeavors, we anticipate uncovering deeper insights, delivering enhanced recommendations, and crafting solutions that possess the potential to reshape industries and transform data-driven decision-making. With the trio of algorithms at our disposal, we stand on the precipice of a data-driven revolution, where the fusion of algorithmic ingenuity and domain knowledge will usher in a new era of insights and innovation.

Bibliography

A, S. O., M, A., J, J. J., & N, M. (2018). Optimized low computational algorithm for elderly fall detection based on machine learning techniques. *Biomedical Research-tokyo, 29*(20), 3715-3722. Retrieved 8 24, 2023, from https://alliedacademies.org/articles/optimized-low-computational-algorithm-for-elderly-fall-detection-based-on-machine-learning-techniques-11016.html

Ahima. (2012). Data Quality Management Model (2012 update) - Retired. *Journal of AHIMA, 83*(7), 62-71. Retrieved 8 24, 2023, from http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1\_049664.hcsp?ddocname=bok1\_049664

Applebaum, W. (1951). Studying Customer Behavior in Retail Stores. *Journal of Marketing, 16*(2), 172-178. Retrieved 8 24, 2023, from https://journals.sagepub.com/doi/10.1177/002224295101600204

Areelu, F., & Akinsola, M. K. (2014). INFLUENCE OF TIERED LESSON AND GROUP PERSONALIZATION INSTRUCTIONAL STRATEGIES ON SENIOR SECONDARY SCHOOL STUDENTS’ ACHIEVEMENT IN MATHEMATICS. *European Scientific Journal, ESJ, 10*(16). Retrieved 8 24, 2023, from http://eujournal.org/index.php/esj/article/download/3541/3313

Borse, S., Pisal, S., Chorghe, P., & Fate, P. (2017). Market Basket Analysis with Mining Association Rule(ShoppingBasketSystem). *International Journal of Advance Research and Innovative Ideas in Education, 3*(2), 2129-2136. Retrieved 8 24, 2023, from http://ijariie.com/formdetails.aspx?menuscriptid=3138

Bramer, M. (2013). *Association Rule Mining II*. Retrieved 9 28, 2023, from https://link.springer.com/chapter/10.1007/978-1-4471-7307-6\_17

Brenes, E., Ciravegna, L., Ciravegna, L., & Montoya, D. (2015). Super Selectos: Winning the war against multinational retail chains. *Journal of Business Research, 68*(2), 216-224. Retrieved 8 24, 2023, from https://sciencedirect.com/science/article/abs/pii/s0148296314003130

Cavique, L. (2007). A scalable algorithm for the market basket analysis. *Journal of Retailing and Consumer Services, 14*(6), 400-407. Retrieved 9 28, 2023, from https://sciencedirect.com/science/article/abs/pii/s0969698907000148

Chang, H.-Y., Lin, J.-C., Cheng, M.-L., & Huang, S.-C. (2016). *A Novel Incremental Data Mining Algorithm Based on FP-growth for Big Data*. Retrieved 8 24, 2023, from https://ieeexplore.ieee.org/document/7564172

Dippold, K., & Hruschka, H. (2013). A parsimonious multivariate poisson model for market basket analysis. *Review of Managerial Science, 7*(4), 393-415. Retrieved 8 24, 2023, from https://link.springer.com/article/10.1007/s11846-012-0088-7

Divya, B., & Neeru, A. (2011). Mining Frequent Item sets without Candidate Generation using Optical Neural Network. *International Journal of Computer Applications*(4), 14-18. Retrieved 8 24, 2023, from https://ijcaonline.org/specialissues/ait/number4/2846-227

Fang, X. (2013). *An Improved Apriori Algorithm on the Frequent Itemse*. Retrieved 8 24, 2023, from https://download.atlantis-press.com/article/8042.pdf

Gajera, S., Limbad, N., & Badheka, M. (2015). A Survey on Techniques of Frequent Pattern Mining. *Data mining and knowledge engineering, 7*(1), 26-29. Retrieved 8 24, 2023, from http://ciitresearch.org/dl/index.php/dmke/article/view/: dmke012015007

Gjino, G., & (Findiku), O. I. (2014). Mobile banking: near future of banking. *Review of applied socio-economic research, 7*(1), 43-51. Retrieved 8 24, 2023, from http://reaser.eu/repec/rse/wpaper/r7\_5\_gjino\_ilollari\_p43-51.pdf

Glanz, K., Bader, M. D., & Iyer, S. (2012). Retail Grocery Store Marketing Strategies and Obesity An Integrative Review. *American Journal of Preventive Medicine, 42*(5), 503-512. Retrieved 8 24, 2023, from https://ncbi.nlm.nih.gov/pubmed/22516491

Gunawardana, A., & Shani, G. (2015). *Evaluating Recommender Systems*. Retrieved 8 24, 2023, from https://microsoft.com/en-us/research/publication/evaluating-recommender-systems

Ishita, R., & Rathod, A. (2016). Frequent Itemset Mining in Data Mining: A Survey. *International Journal of Computer Applications, 139*(9), 15-18. Retrieved 8 24, 2023, from https://ijcaonline.org/archives/volume139/number9/24517-2016909219

Joshi, I. D., Khanna, P. N., Sabale, M. R., & Tathawade, N. B. (2017). A Comparative Study of Association Mining Algorithms for Market Basket Analysis. *International Journal of Advance Research and Innovative Ideas in Education, 3*(1), 702-707. Retrieved 8 24, 2023, from http://ijariie.com/adminuploadpdf/a\_comparative\_study\_of\_association\_mining\_\_algorithms\_for\_market\_basket\_analysis\_ijariie3740.pdf

Joshi, S., Jadon, R. S., & Jain, R. C. (2010). An Implementation of Frequent Pattern Mining Algorithm using Dynamic Function. *International Journal of Computer Applications, 9*(9), 24-28. Retrieved 8 24, 2023, from https://ijcaonline.org/archives/volume9/number9/1410-1904

Kamakura, W. A. (2012). Sequential market basket analysis. *Marketing Letters, 23*(3), 505-516. Retrieved 9 28, 2023, from http://wak2.web.rice.edu/bio/my reprints/sequentialmarketbasketanalysist.pdf

Khan, S. A., & Kumar, K. P. (2013). Market Basket Analysis Using Horizontal Aggregations In SQL. *International journal of engineering research and technology, 2*(4). Retrieved 8 24, 2023, from https://ijert.org/research/market-basket-analysis-using-horizontal-aggregations-in-sql-ijertv2is4256.pdf

Kim, H., Fiore, A. M., Niehm, L. S., & Jeong, M. (2010). Psychographic characteristics affecting behavioral intentions towards pop‐up retail. *International Journal of Retail & Distribution Management, 38*(2), 133-154. Retrieved 8 24, 2023, from https://emerald.com/insight/content/doi/10.1108/09590551011020138/full/pdf

Kumaran, V. S., & Sankar, A. (2013). Recommendation System for Adaptive E-learning using Semantic Net. *International Journal of Computer Applications, 63*(7), 19-24. Retrieved 8 24, 2023, from https://ijcaonline.org/archives/volume63/number7/10478-5210

Li, X., Li, X., & Hudson, S. (2013). The application of generational theory to tourism consumer behavior: An American perspective. *Tourism Management, 37*, 147-164. Retrieved 8 24, 2023, from https://sciencedirect.com/science/article/pii/s0261517713000435

Long, Z. A., Bakar, A. A., & Hamdan, A. R. (2011). *Frequent pattern using Multiple Attribute Value for itemset generation*. Retrieved 8 24, 2023, from http://yadda.icm.edu.pl/yadda/element/bwmeta1.element.ieee-000005976503

Mahmood, S., Shahbaz, M., & Guergachi, A. (2014). Negative and Positive Association Rules Mining from Text Using Frequent and Infrequent Item sets. *The Scientific World Journal, 2014*(2014), 973750-973750. Retrieved 8 24, 2023, from https://hindawi.com/journals/tswj/2014/973750

Mahmoodi, S. A., Mirzaie, K., & Mahmoudi, S. M. (2016). A new algorithm to extract hidden rules of gastric cancer data based on ontology. *SpringerPlus, 5*(1), 312-312. Retrieved 8 24, 2023, from https://ncbi.nlm.nih.gov/pmc/articles/pmc4786510

Man, M., & Jalil, M. A. (2019). Frequent itemset mining: technique to improve eclat based algorithm. *International Journal of Electrical and Computer Engineering, 9*(6), 5471-5478. Retrieved 8 24, 2023, from http://ijece.iaescore.com/index.php/ijece/article/view/20538

*Market analysis*. (n.d.). Retrieved 8 24, 2023, from http://fyi.uwex.edu/downtown-market-analysis/analysis-of-opportunities-by-sector/retail-service-businesses/

*Market Analysis*. (n.d.). Retrieved 9 28, 2023, from NeMBA: http://www.netmba.com/marketing/market/analysis/

Masthoff, J., & Vassileva, J. (2015). *Tutorial on Personalization for Behaviour Change*. Retrieved 8 24, 2023, from https://dl.acm.org/citation.cfm?id=2716264

Mende, M., Bolton, R. N., & Bitner, M. J. (2013). Decoding Customer–Firm Relationships: How Attachment Styles Help Explain Customers' Preferences for Closeness, Repurchase Intentions, and Changes in Relationship Breadth. *Journal of Marketing Research, 50*(1), 125-142. Retrieved 8 24, 2023, from https://journals.sagepub.com/doi/full/10.1509/jmr.10.0072

Mohamed, M. H., & Darwieesh, M. M. (2014). Efficient mining frequent item sets algorithms. *International Journal of Machine Learning and Cybernetics, 5*(6), 823-833. Retrieved 8 24, 2023, from https://link.springer.com/article/10.1007/s13042-013-0172-6

Musalem, A., Aburto, L., & Bosch, M. (2018). Market basket analysis insights to support category management. *European Journal of Marketing, 52*, 1550-1573. Retrieved 8 24, 2023, from https://emerald.com/insight/content/doi/10.1108/ejm-06-2017-0367/full/html

Nandagopal, S., Arunachalam, V. P., & Karthik, S. (2012). A NOVEL APPROACH FOR MINING INTER-TRANSACTION ITEM SETS. *European Scientific Journal, ESJ, 8*(14). Retrieved 8 24, 2023, from http://eujournal.org/index.php/esj/article/viewfile/215/231

Rajalakshmi, M., Purusothaman, T., & Nedunchezhian, R. (2011). MAXIMAL FREQUENT ITEMSET GENERATION USING SEGMENTATION APPROACH. *International Journal of Database Management Systems, 3*(3), 19-32. Retrieved 8 24, 2023, from https://arxiv.org/pdf/1109.2427

Rathi, S., & Dhote, C. A. (2015). *Parallel Implementation of FP Growth Algorithm on XML Data Using Multiple GPU*. Retrieved 8 24, 2023, from https://link.springer.com/chapter/10.1007/978-81-322-2250-7\_58

Rong, Z., Xia, D., & Zhang, Z. (2013). *Complex statistical analysis of big data: Implementation and application of Apriori and FP-Growth algorithm based on MapReduce*. Retrieved 8 24, 2023, from http://dro.deakin.edu.au/view/du:30123948

Singh, P., Katiyar, N., & Verma, G. (2014). Retail Shoppability: The Impact Of Store Atmospherics & Store Layout On Consumer Buying Patterns. *International Journal of Scientific & Technology Research, 3*(8), 15-23. Retrieved 8 24, 2023, from http://ijstr.org/final-print/aug2014/retail-shoppability-the-impact-of-store-atmospherics-store-layout-on-consumer-buying-patterns.pdf

Stanisic, P., & Tomovic, S. (2015). APRIORI MULTIPLE ALGORITHM FOR MINING ASSOCIATION RULES. *Information Technology and Control, 37*(4). Retrieved 8 24, 2023, from http://itc.ktu.lt/index.php/itc/article/view/11985/6660

Surjandari, I., & Seruni, A. C. (2010). Design of Product Placement Layout In Retail Shop Using Market Basket Analysis. *Theory of Computing Systems \/ Mathematical Systems Theory, 9*(2), 43-47. Retrieved 8 24, 2023, from http://journal.ui.ac.id/technology/journal/article/view/379

Trendafilov, N. T., Kleinsteuber, M., & Zou, H. (2014). Sparse matrices in data analysis. *Computational Statistics, 29*(3), 403-405. Retrieved 8 24, 2023, from https://link.springer.com/article/10.1007/s00180-013-0468-8

Trieu, T., & Kunieda, Y. (2012). *An improvement for dEclat algorithm*. Retrieved 8 24, 2023, from https://dl.acm.org/citation.cfm?doid=2184751.2184818

Videla-Cavieres, I. F., & Ríos, S. A. (2014). Extending market basket analysis with graph mining techniques: A real case. *Expert Systems With Applications, 41*(4), 1928-1936. Retrieved 8 24, 2023, from https://sciencedirect.com/science/article/pii/s0957417413007094

Wu, X., Zhu, X., Wu, G., & Ding, W. (2014). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering, 26*(1), 97-107. Retrieved 8 24, 2023, from https://ieeexplore.ieee.org/document/6547630

Yoder, R., Vandenberg, S., & Breimer, E. (2011). Market Basket Analysis for Non-Programmers. *Information Systems Education Journal, 10*(6), 38. Retrieved 9 28, 2023, from http://isedj.org/2012-10/n6/isedjv10n6p38.html

Zein, M. F. (2016). *APPLICATION OF DATA MINING ANALYSIS USING APRIORI ALGORITHM FOR BUYING HABITS OF CUSTOMER FROM INVENTORY DATABASE OF SINAR KARYA FURNITURE*. Retrieved 8 24, 2023, from http://eprints.dinus.ac.id/18227

Zhixin, M., Yusheng, X., & Dillon, T. S. (2013). *Mining Frequent Sequences Using Itemset-Based Extension*. Retrieved 8 24, 2023, from http://iaeng.org/publication/imecs2008/imecs2008\_pp591-596.pdf