Tab 1

Final Paper

Ethan Cooper, Rituparna Ghosh, Ming Murray, Sammy Nouadir,

Stela Opingari, Trevor Small, David Williams

COSC 411 - Algorithm Design & Analysis

**Abstract**

This research paper explores how the Apriori Algorithm can be used to complete a series of tasks related to item recommendations, specifically on a grocery store dataset. Firstly, it will analyze the method in which the algorithm is used to create a list of recommended items based on the user’s previous selections. It will then go into detail about how the same algorithm can be used in order to recommend items based on the user's current aisle. Lastly, it will explain the implementation of the concept in instances, as well as the history of the algorithm over time. In addition to how the Apriori Algorithm has been analyzed in multiple articles, it will also cover other algorithms that have been used for a similar purpose, and how they differ from the algorithm chosen for this project. In this analysis, there will also be a discussion of how the algorithm could be improved or has been enhanced for our implementation. We have utilized various methods to increase efficiency in the use of this algorithm, as well as used Apriori, which has not been implemented by previous web applications, such as Amazon.

**Introduction**

With artificial intelligence continuing to dominate the market, we as computer scientists have a mission to learn as much as we can so as to not become obsolete. One of the most utilized concepts within the current market is predictive analytics in artificial intelligence. This topic is heavily studied within this field, as it is used to understand sales behaviors. It is also used in the industry to recommend items to consumers based on their previous purchases. While the use of predictive AI is an important concept in big data, this paper will focus on the latter half of the previous statement. This concept is seen in almost every instance of online advertising, to the point where people wonder if their computer is “spying” on them, or storing their information in an unsecure way. However, even if the algorithms require stored user information in order to recommend, there is seldom a human looking at that data. To prove this concept, this paper will focus on an algorithm and how it can be used to predict items for everyday shopping purposes. This algorithm is called the Apriori Algorithm.

The concept of predictive algorithms and their use in recommendation systems is broad, and the reason that we decided to implement the Apriori Algorithm is because it is a relatively simple version of many other algorithms. This is because it works specifically on known data, which in the scope of our implementation, is sufficient. Specifically, the algorithm must be used on information that is contained in the dataset that it is pulling from. For example, if a user selects ‘*apples*’ then the dataset must contain a value or column for *‘apples’* otherwise it will not be able to recommend an item. In the corporate scope, a large company such as Amazon would not be able to utilize this algorithm to recommend items, as the store contains so many items, and new items are added constantly, that it would not be realistically possible. However, this also means that our algorithm is more efficient relative to an algorithm that a large company might be using, such as deep learning. This is because we do not have the CPU power to run a deep learning algorithm quick enough to recommend an item almost immediately, Apriori seemed like the best choice.

Throughout this paper, a few different algorithms will be analyzed, as they are all related to predictive artificial intelligence, however, they are not utilized in the final implementation. One of the concepts we will explore is how generative unsupervised learning is used to increase optimization across machine learning models. This concept is analyzed in order to better understand how current algorithms might be improved. However, it is not within the scope of our actual project since the paper focuses on language models. Another concept that is brought up multiple times is *data mining* and *market basket analysis*. These concepts are important to better understand the Apriori Algorithm, as it is an example of how data mining is implemented. One of the more broad conceptual analyzes that will be gone over later on in this paper is explaining the difference between different types of AI. It will explain up to ten categories of AI, and the general idea behind each type that is discussed. This is important in understanding different techniques that can be used to create predictive AI models. One algorithm that is not used in our project implementation, but is discussed in this paper, is called the Customer Churn Prediction. This algorithm is slightly more advanced than the Apriori Algorithm, where it combines many different AI concepts to significantly improve predictive accuracy. Other more advanced concepts, such as deep learning, are also discussed shortly without being used within our implementation. These are important concepts as they could be used to better improve the project, but at this time, we decided not to implement them.

The majority of this paper will focus on key concepts related to the Apriori Algorithm. Most notably, association rules and how they affect the overall algorithm, as well as data mining. It will also mainly discuss ideas pertaining to increasing the efficiency of this specific algorithm. The key concepts that will be discussed related to this are: firstly, adding transaction reduction in order to better the time that it takes to complete. Secondly, through association rules mining by reducing the size of the candidate item sets, and not going through the database multiple times. Lastly, through the difference between supervised and unsupervised machine learning, and how they affect the different approaches to predictive AI research.

The first and possibly most lengthy discussion in this paper is how our project is set up, as well as how the research done lines up with the final product created. Firstly, our goal for the application was to create a website that allows users to have real-time predictions of items that they might want to buy next. From the list of “recommended items” the shopper will have the ability to find that item quickly either from the same aisle that they are in or by switching aisles through a series of buttons. Each aisle is displayed through an HTML button, and the aisles are based on a column that is contained in the dataset. The majority of this code is done through the UI, where a JavaScript function will turn each item from the dataset into a JSON object and then display a picture that can be clicked. Underneath each picture also has the name of the object from the dataset (taken from the JSON objects list). As an item gets clicked, it is added to the “shopping cart.” The “cart” is essentially a list of concatenated strings that get sent to the back end. The back end is a Python script that is based on the Apriori Algorithm. This utilizes the dataset to predict items based on the string that is in the “cart.” Once the list is made, it is returned to the front end to be displayed through the Flask library structure. In the shopping cart user interface, there are three things: the list of items in the cart, which has a picture associated with each item and the price of that object underneath, and two recommendation lists. One list is the recommendations based on the items that the user currently has in the shopping cart, which is updated every time the user adds a new item. The other list is based on the aisle the user is currently looking at. For example, if the user is in an aisle named “Cleaning Supplies” then the algorithm will return items that are not currently in the shopper’s cart and are cleaning supplies. Or if the shopper has something in the cart like “dish soap,” the recommended item might be “sponge” since those items are similar.

Another topic that is important to the discussion of how the final project is set up is the dataset. Throughout this paper, the dataset will be referenced multiple times, as it is important to understand what the algorithm is using in order to produce accurate predictions. As a general concept, the dataset is simply a .csv file that is being utilized by both the back end and front end code. During the discussion of the code section of the paper, there will be a more detailed analysis of what is contained within this dataset, as well as how it is set up in order to be accurately represented within the web application.

Overall, the goal of this research paper is to analyze the well-documented methods of predictive artificial intelligence, as well as understand how those concepts could be iterated and improved on. It also aims to explain how the web application we created relates to these concepts, and how we improved these methods to create a final working application that efficiently utilizes predictive AI. Creating an application using the concepts we researched, we will show how necessary predictive AI is in the current market, as well as how many different methods can be used to achieve a similar outcome. Even if some algorithms are better than others, it can be shown that in different circumstances, most of them would be beneficial to be used despite their shortcomings in comparison to a more advanced one. For this reason, the Apriori Algorithm was the best option out of the many different predictive algorithms for our project goals and our final implementation.

**Existing Works**

This article provides a broad overview on the development and application of artificial intelligence (AI) across various fundamental sciences, including information science, mathematics, medical science, materials science, geoscience, life science, physics, and chemistry. AI frameworks and platforms have reduced the complexity of developing models in information science, enabling researchers to focus on designing neural networks. In mathematics, AI techniques like reinforcement learning, generative adversarial networks and Bayesian learning are addressing challenges related to optimization, interpretability and generalization. In medical science, AI is revolutionizing healthcare through applications like AI doctors, outbreak detection, biomarker discovery, medical imaging, wearable devices, and drug discovery. AI tools are improving diagnosis, treatment, as well as patient management. Materials sciences benefit from AI's ability to accelerate material discovery by integrating data-driven methods with traditional approaches. In geoscience, AI is aiding climate change studies, disaster prediction and resource management. Another application of artificial intelligence worth mentioning is how AI and neuroscience mutually inspire each other, with AI aiding in omics data analysis, smart agriculture, and understanding brain functions. Physics leverages AI to improve simulations and data analysis in fields like particle physics and astronomy, and in chemistry, AI is automating processes in computational chemistry and aiding in the discovery of catalysts. The article emphasizes AI's growing role in scientific research, its challenges, and the importance of developing secure, lifelong learning models.

Another article significant to this project is ‘Machine Learning(ML) Algorithms’. In Arthur Samuel’s words, ML is a discipline of algorithms or statistical models that enable computers to learn from data without explicit programming. Some ML approaches are supervised, unsupervised, semi-supervised, reinforcement, and ensemble learning. Supervised learning relies on labeled datasets to train models, with popular methods such as decision trees, Naive Bayes, and support vector machines. On the other hand, Unsupervised learning uncovers patterns in unlabeled data using techniques like k-means clustering and principal component analysis. Mentioning semi-supervised has a mixture of labeled and unlabeled data for improving efficiency and optimization, leveraging methods like transductive SVM and generative models. Optimal decisions are made by agents through a reinforcement learning process through environmental interactions to maximize cumulative rewards. Ensemble learning, which enhances accuracy by combining multiple models, employs techniques like Boosting and Bagging. Overall, this paper dives into neural networks, which simulate brain activity, and instance-based methods like k-Nearest Neighbors. It takes into account that supervised learning works well for smaller datasets, while unsupervised learning is better suited for larger ones. About ML growth and importance, the discussion emphasizes its applications in areas like online recommendations and advanced data analysis, showcasing its transformative impact across industries.

The next article that we found was ‘Implementation of Apriori Algorithm for Analysis of Consumer Purchase Patterns’ beginning with introducing Cafe Bojack Shop which acts as a source of dataset analysis providing real-world insights for utilizing Apriori Algorithms. Apriori Algorithms are highlighted as a powerful tool for store owners to uncover patterns in consumer purchase behavior. This algorithm means the discovery of meaningful patterns in data through specific methods, and association rules, which are derived using this algorithm. The process can be started with the calculation of support value representing the frequency of item combinations in the dataset. Next, it determines the confidence value, which measures the reliability of these patterns. The article explains that the Apriori Algorithm operates on candidate item sets of size k, in an iterative manner pruning to subsets of size k-1. This algorithm works by supporting the value for each candidate item set by scanning the dataset, continuing this process until reaching the longest item set. High-frequency patterns with support values above a defined threshold are identified. If no such patterns are found, the algorithm halts; otherwise, it increments k and repeats the process. Such an example of a transaction dataset is provided to show the algorithm in action, accompanied by the use of case diagrams and analytical forms to visualize the problem-solving approach. The article concludes by emphasizing the benefits of the Apriori Algorithm in analyzing consumer purchase patterns, showcasing its utility in data-driven decision-making for retail businesses.

The other article that is useful for this group project was found to be ‘Improving Efficiency of the Apriori Algorithm Using Transaction Reduction’. This article focuses on enhancements to the Apriori Algorithm, beginning with foundational concepts such as data mining—the extraction of significant patterns from datasets—and association rules. Four key definitions of association rules are Transactional Database D which contains finite item sets I = {i1, i2,...,in }, where each item ik has a unique transaction ID (Tid). Association rules take the form X ⇒ Y, where X and Y are subsets of I, and X∩Y=∅. Support (S) measures the percentage of transactions containing X ∪ Y, while confidence (C) measures the percentage containing both X and Y. And lastly, Item sets with support values above the minimum threshold are called frequent item sets, while those below are infrequent cases.

The article later brings up reduction techniques to improve its efficiency. A new attribute, Size of Transaction (SOT), is added to the dataset, representing the number of items in each transaction. During execution, the algorithm assigns SOT = K, searches for other transactions of size K, and removes redundant nodes, effectively reducing the size of the dataset being processed. An example case can be a dataset illustrating the enhanced algorithm in action, showing how the addition of the SOT attribute streamlines operations. The article concludes by emphasizing the increased efficiency of this modified approach, reducing computational overhead while maintaining the Apriori Algorithm’s core functionality.

The subsequent analysis that we found was how ‘An Improved Apriori Algorithm for Mining Association Rules’ reviewed the evolution and applications of mining that uncover relationships and frequent patterns within datasets. The Apriori Algorithm, which is a cornerstone of this field, has been widely used in commercial and medical data analysis. However, the paper critiques the traditional Apriori approach for its inefficiency and complexity. The research aims to build on the strength of the original Apriori Algorithm while addressing its limitations by eliminating the need to repeatedly scan the database, reducing the size of candidate item sets, or enhancing the joining and pruning processes for greater speed.

This improved algorithm employs a novel database mapping technique to minimize rescanning, aggressively prunes frequent and candidate item sets to streamline operations, and incorporates overlap-based support counting to boost efficiency. Detailed examples illustrate these enhancements, showing how the optimized algorithm accelerates the mining process while maintaining accuracy. The research results indicate that the proposed implementation is significantly faster than the original Apriori Algorithm and outperforms other improved versions. This improvement demonstrates a practical solution to the efficiency challenges in association rule mining.

The following article ‘An Overview of Supervised Machine Learning Methods’ focuses on an introduction to machine learning that categorizes its algorithms based on the problem's goals. The primary groups include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, transduction, and learning to learn. However, the focus is on the two core categories only. Supervised and unsupervised learning. In supervised learning, input data is labeled, meaning that the classes are predetermined by human input. These algorithms identify patterns and create mathematical models based on the labeled data. In contrast, unsupervised learning deals with unlabeled data, aiming to classify inputs into distinct categories. Also, the supervised learning process is divided into two key steps. Such as training, which means, sample data is processed by the algorithm to build the learning model.

The model uses an execution engine to make predictions, producing tagged data or final predictions. The goal of this publication was to construct an estimator table that predicts the label of an object based on its features. The algorithm refines itself by comparing its predictions with correct outputs, adjusting to minimize errors. This paper then further highlights differences among supervised learning algorithms, including Decision Trees, Linear Regression, Naive Bayes, and Logistic Regression, emphasizing their unique approaches to handling data and making predictions.

The relevant paper that we found is ‘A State-of-the-Art Review on Machine Learning Algorithms: Supervised Learning in Data Classification’ which is based on ML fundamentals categorizing algorithms into Supervised, Unsupervised, Semi-Supervised, and Reinforcement learning. However, it narrows its focus to the two primary groups: Supervised and Unsupervised learning. The paper explores supervised learning algorithms in depth, dividing them into three main categories Probabilistic Classifiers, Linear Classifiers and Other Classifiers. In Probabilistic Classifiers, they use mixture models for data classification. Popular examples include Naive Bayes Classifier (NBC), Bayesian Network (BN), and Maximum Entropy Classifier (ME). Linear Classifiers include categories that encompass algorithms like Support Vector Machine (SVM), Multilayer Perceptron (MLP), Logistic Regression (LR), Rule-based Classifiers, Decision Trees (DT), and Random Forests (RF). The rest, Other Classifiers, includes unique techniques like Case-Based Reasoning, Boosting, and Quadratic Classifiers.

This paper mainly focuses on supervised learning involving human-labeled input and feedback, enabling algorithms to refine predictions during the training phase. It concludes by reiterating the goal of reviewing widely used supervised learning approaches in data classification. The review evaluates these methods based on their objectives, methodologies, advantages, and limitations.

The article “Painless Unsupervised Learning with Features" looks over unsupervised learning methods for machine learning, and more specifically, the effects of added features within a standard generative unsupervised learning model which could provide optimizations and improvements across many different models. This was a large inspiration for our project. The article begins by looking briefly at the difference in the presence of supervised and unsupervised learning within machine learning models. They look at the recent improvements of unsupervised learning thanks to increased data, a better understanding of modeling methodology, and improved optimization algorithms. These improvements have been helpful, but in these researchers' case the most important work being done has been incorporating “richer domain knowledge” into their structures and models. The researchers note that this type of knowledge has typically been encoded in a way that makes injecting it difficult and time-consuming. The paper goes into great detail on how they were able to add richer features declaratively, allowing them to do so without complex machinery and focusing less on structure, instead customizing the features for the model being used. They begin by discussing their part-of-speech (POS) induction that has been modified using the features modeling discussed previously. This POS induction is done by labeling words with tags, and the researchers discovered that their models, despite the relative simplicity, were able to outperform their normalized counterparts and show improvements to the algorithm being utilized. Two optimization methods they look at are the EM algorithm and the Direct Marginal Likelihood method. They found that the Marginal Likelihood method led to higher accuracy for certain models, but the EM algorithm was found to be more efficient in situations where computing costs increase. The researchers then go through different linguistic challenges of speech learning models including grammar induction, part-of-speech induction, word alignment, and word segmentation. The researchers found improvements utilizing their methodology of enriching simple, normalized models to a point where improvements can be made that outperform more expensive or elaborate models. This methodology of specific and deliberate changes on a per-model basis has led to improvements in unsupervised learning that could lead to more benefits in the subject as research continues.

We found the next paper named “Analysis of Data Mining Algorithms in Market Basket Analysis" which delves into the use of data mining techniques, particularly Market Basket Analysis (MBA), to uncover purchasing patterns within product datasets. This technique generally assists with finding patterns specifically within similar product purchasing. The technique can not only assist in the process of sorting through large amounts of data but can show a significant time reduction within the examining of said data. The article begins by looking at MBA and how machine learning in conjunction with this analysis can lead to increased revenue for the retail industry. The article also goes over the many different types of data mining techniques including scientific evaluation, intrusion detection, business transactions, education, and healthcare/insurance. Although there are many practical uses of data mining, this article clarifies why they will specifically be looking at the MBA for the use of analyzing consumer behavior in a retail setting. MBA allows for the forecasting of consumer behaviors. This method uses association rule mining, which finds correlations based on the frequency of the event. The results of this type of mining provides important information that can be used on extremely large data sets. It looks at the antecedents, or the purchases of the buyer, and the consequent, or the correlated items that may be purchased based on the antecedent. These two factors are then computed using support and confidence rules that ensure that item sets are frequent (support) and have minimum levels of correlation, both combined equal the confidence level. The article then looks at utilizing the Apriori algorithm for databases with large amounts of transactions. The article also looks at other algorithms including “sort and merge scan” and “frequent pattern growth” but discovered that Apriori had lower execution times on their datasets when compared to the other algorithms.

The article “AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems” provides an overview of Artificial Intelligence and different methods of creating AI as well as different types of AI and how they can be used to solve and automate real world problems. The article starts by discussing AI-based modeling and how it can be used in many different industries to help solve real world problems by automating intelligent systems. Throughout the article we go into depth on various types of AI techniques and how they can be used in many different fields including but not limited to business, healthcare, and cybersecurity. The article also goes in depth on machine learning and deep learning and how these can be used to allow AI to make better decisions with such a chaotic real world. The article goes into depth on AI techniques and categorizes them into 10 categories, these categories are machine learning, neural networks and deep learning, data mining, rule-based modeling, fuzzy logic-based approach, knowledge representation, case-based reasoning, text mining and natural language processing, visual analytics, computer vision and pattern recognition, hybridization, searching and optimization. With these 10 categories being put into use, the article also goes into some of the different types of AI and what they are all capable of doing. Analytical AI, functional AI, interactive AI, textual AI, and visual AI. Each of these various types of AI hold the power of helping solve real world problems. We take a better look into the various types of reasoning that we can teach our AI to use to solve specific problems. Reasoning is defined as the process of using existing knowledge to conclude and make predictions. 2 of these methods of reasoning are “Case-Based” and Hybrid Approach.

The next article “Estimate and Predict the Foreign Currency Exchanging Rates by Using Adaptive Neuro Fuzzy Inference System and Deep Learning Algorithms” opens discussing some of the different AI algorithms that have been used in the financial field including fuzzy inference systems, and Machine Learning (Ml). There have been many different machine learning algorithms that have been used so far to predict the prices of certain stocks. However, we see a lack of studies working on the prediction of currency exchange and the articles that do only use a single method to make the predictions. This article tries tackling that problem by using Adaptive Neuro Fuzzy Inference System (ANFIS) and Deep Learning (DL). The article talks more about the issues that come up when trying to predict the currency exchange rate such as unforeseen variations on the current market and how it varies significantly over time. The solutions the author brought up are by making the change of the currencies exchange rate a periodic function and setting a certain period to use. This article chose to go with a 3-month period. To get the correct data the author used the NASDAQ data and checked the targets using the ANFIS and DL models. Now that the research on datasets and periods have been discussed it goes into the AI algorithm it will be using to try and get the best prediction on the currency exchange. The (ANFIS) which was first introduced by Jyh-Shing Roger Jang in 1992, is composed of Artificial Neural Network (ANN) as well as the Fuzzy Inference System (FIS). The FIS works like a multilayer feed forward neural network, however the links between nodes have no weight factors. It has a network of neurons that communicate between the input and hidden layers, and the hidden and output layers. The author also spoke on how ANN and deep learning are used in their project and how the neural networks were adjusted based on the comparison of the output and target. There were 3 columns, Open, High, and Low that worked as inputs and close as the output. The results of this project are very promising, as the ANFIS algorithm is about 64% better than the ANN/DL algorithm.

The research paper “Customer Churn Prediction Using Weight Average Ensemble Machine Learning Model” gave us more information on AI in business as it aims to address the critical problem of customer churn, where customers stop using a product or service, by developing a highly accurate predictive model. Customer churn is a major concern for businesses, as it directly affects revenue and long-term growth. The study proposes an ensemble machine learning approach, specifically a weighted average ensemble model, which combines multiple individual machine learning algorithms to improve the overall prediction accuracy. The authors of the study utilized various machine learning algorithms, such as decision trees, support vector machines (SVM), and random forests. Each of these algorithms has strengths and weaknesses when predicting churn. The weighted average ensemble model assigns higher weights to algorithms that perform better, ensuring that their influence on the final prediction is stronger. By doing so, the model leverages the advantages of each algorithm while minimizing their weaknesses. Through experiments and comparative analysis, the research demonstrates that the weighted average ensemble model outperforms traditional single-model approaches, delivering superior accuracy and reliability in predicting customer churn. This model allows businesses to better understand customer behavior, enabling them to take proactive steps to retain valuable customers. By reducing churn rates, companies can prevent revenue losses and improve customer satisfaction. The approach is particularly valuable for industries with high churn rates, such as telecommunications, banking, and e-commerce, where customer retention is a key factor in sustaining profitability.

This article, “A survey for user behavior analysis based on machine learning techniques: current models and applications”, dives into how machine learning (ML) is used to understand user behavior in industries like e-commerce, social media, and healthcare. The paper focuses on different types of ML models and how they’re used to analyze data, predict user actions, and improve customer experiences. Some of the key models covered are supervised learning, where the model learns from labeled data to predict specific outcomes, like which users are likely to buy a product, unsupervised learning, used to find hidden patterns in data, often applied for grouping users based on similarities (user segmentation), and reinforcement learning, which adapts to user behavior over time, helping companies improve real-time interactions, like in chatbots or dynamic pricing. The article also discusses deep learning, which is super helpful for analyzing massive datasets (think social media or streaming services), and helps identify more complex patterns in user behavior. This is the tech behind recommendation systems like Netflix or Spotify, where they suggest new content based on what you've watched or listened to. It doesn't just cover the technical side but also shows how these models are applied in real-world cases. For example, recommendation systems on platforms like Amazon use machine learning to make personalized product suggestions. Sentiment analysis is another big one, where companies analyze social media posts or reviews to figure out customer opinions. The paper also highlights user segmentation, which helps businesses target specific audiences more effectively. On top of all that, the article brings up some key challenges, like privacy concerns and ethical issues. Collecting and using so much personal data raises questions about user consent and how companies should handle data responsibly, especially with concerns around bias or data misuse. Overall, this survey gives a solid overview of how machine learning is changing the way businesses understand and predict user behavior, while also acknowledging the ethical and technical hurdles that come with it.

As computer and communication technologies converge, managing increasingly complex systems for less technically advanced users becomes a challenge. This article, “Learning User models for an intelligent telephone assistant”, provided an overview of intelligent interfaces, such as the Intelligent Personal Assistant (IPA) from British Telecommunications (BT), assist with managing tasks like scheduling, communication, and email or call filtering by monitoring user activity and learning user preferences. The ability of such systems to effectively model user behavior by identifying patterns in user data is critical to their success. The document explores the development and implementation of user models for an intelligent telephone assistant, focusing on the need for adaptive systems that improve user interaction without requiring explicit input. It introduces a novel approach called Flexible Incremental Learning of User models (FILUM), which is based on dynamic support logic algorithms. In contrast to traditional methods, FILUM defines behaviors for the models rather than attribute-value pairs. These prototypes help forecast user actions by dynamically changing probability in response to observed behavior, which allows the system to gradually refine its model. The telephone assistant responds to incoming calls based on user context, caller important, call frequency, and diary entries. To test this approach, the researchers employed simulations using the n-player iterated prisoner’s dilemma as a model to gather data. This test environment allowed the system to predict user behavior without detailed knowledge of decision-making algorithms, similar to how the telephone assistant predicts call-handling preferences. FILUM successfully achieved over 80% accuracy in prediction, showcasing its effectiveness in building user models. The system’s design allows it to adapt and learn from user feedback improving over time, though further validation is needed for widespread application.

This paper, “Data Mining Algorithm Data Model Data Analysis Based on Artificial Intelligence Technology”, examines developments in data mining techniques, with an emphasis on how to improve data analysis by integrating artificial intelligence technologies like the BP neural network. The importance of data mining in scientific research is highlighted, with its growth being fueled by artificial intelligence technologies. The BP neural network handles complex, nonlinear interactions more successfully than traditional data mining techniques, notwithstanding their usefulness. The introduction of a modified HA-BP neural network, which outperforms the extraction speed and has over 95% accuracy, this technology speeds up the process of extracting data with addition to greatly improving accuracy. The paper highlights how the model can be used in different domains such as stock market predictions and customer behavior analysis, where the rapid extraction and interpretation of data provide actionable insights. Combining AI with data mining turns unstructured data into insightful knowledge that can be used to diagnose and make predictions with accuracy. This promotes data-driven innovation in various scientific domains. Furthermore, this paper breaks down the stages involved in data mining, including data preparation, preprocessing, and the construction of a mining model to extract valuable patterns from large datasets. The enhanced BP neural network performs better in identifying hidden patterns and correlations inside big datasets, demonstrating its adaptability to a wide range of applications. It can also handle nonlinear data systems. The study emphasizes how crucial it is to prepare data and how creating reliable, consistent datasets is necessary before mining.

**Our Implementation and Improvements**

As explored previously in the *Existing Works* section, there are various other research papers and implementations that discuss, utilize, and attempt to improve on existing AI algorithms. And more specifically, many others have done this with the Apriori Algorithm which we have chosen for our implementation. This section will aim to go into great depth about exactly how we have implemented the Apriori Algorithm into our back end, the improvements we have made to the algorithm, and how it all comes together in the front end which the user interacts with. As a refresher, here are the technologies we have utilized in the project:

*Front end: HTML/CSS, JavaScript, JSON*

*Back end: Python, MLxtend, Pandas, Flask*

To reiterate what was said in the *Introduction* about how the web application works*,* the front end of our app which showcases each Aisle page selectable by a button, is initially displayed through the back end flask code. It is then dynamically updated based on the current aisle that is selected and can be changed by the user. When the user selects a button, a function call is made to regenerate the *item\_display* container in our HTML with the newly selected aisle’s items. The items and their relevant data: the name, price, and an image of the product; are stored in the JSON file that was generated from the dataset that holds the previous transactions history, which has some modifications that will be discussed later. The JSON file which is used for items to be displayed anywhere on the page holds the name of each aisle (produce, frozen food, canned goods, etc) and all the items that are located in said aisle. Then, for each item in the selected aisle, a predefined HTML block is generated with the item’s name, price, image, and a button to add it to the user’s cart are then appended to the *item\_display* container.

In order to support the dynamic nature of our front end, we had to develop a well-thought-out back end. We utilized Python as the main programming language to develop the back end, along with the Flask framework. We decided to make use of Flask mainly because of its simplicity and efficiency when working with data transfer between the front and back end. Using Flask provided a compelling foundation that enabled rapid development and scalability of our web application. By taking advantage of the Flask framework we were able to handle user interactions efficiently, process data requests and deliver real-time updates to the front end all of which create a smooth and responsive user experience. Integrating Flask in our data processing tools not only supports the dynamic nature of the front end but also optimizes the functionality and overall performance of our web application.

As part of our back end, we have implemented several necessary libraries. The MLxtend library is crucial for implementing the Market Basket Analysis (MBA) along with the Apriori Algorithm. In addition, it also offers a set of essential tools that can be used for processes such as data mining and machine learning, which together simplify the process of association rule mining and make our implementation more efficient overall. Another library that we have included in our back end is the Pandas library, which helped us prepare and manipulate the dataset. Pandas allowed us to clean, transform, and organize our data efficiently. We ensured the clarity of our dataset because it directly affects the efficiency of the Apriori Algorithm.

The collection of data that we used as part of our project consists of 38765 unique transactions stored in a CSV file. These transactions are spread out in over 700 different dates of purchases and there are a total of 166 unique items. Initially each item had only three values: member\_number, data and item description. Due to the fact that we wanted to add the functionality of recommending items based on the customer's location, we had to add another column to our dataset named itemAisle. We categorized all the items into 15 aisles: alcohol, baking, beverages, cleaning items, condiments and spices, dairy, deli, dessert, dry and canned food, frozen foods, household goods, personal care, pets, produce and snacks. Additionally, we used a script to add custom aisle tags to all the items in the data set. This tagging system allows us to analyze purchases in multiple ways, which leads to a more accurate representation of user purchase trends. Another modification we made to the dataset was going through an extensive data cleaning and removal of inconsistencies, duplicates, and irrelevant information. The reason for doing so is to increase the accuracy and reliability of our dataset, since it directly impacts the quality of the recommendations generated by the Apriori Algorithm.

The data cleaning process was later followed by a transformation process, in which we converted our dataset into a suitable format for the Apriori Algorithm. For each individual transaction there is a basket of items which essentially is just a list of items purchased together in one transaction. The basket of items allows the algorithm to identify frequently occurring patterns and derive meaningful association rules. Another component of the transformation is feature selection, a process in which we identify and select relevant features for the items such as name, price and aisle location. These features are instrumental in generating association rules from which we have derived our recommendation system.

We decided to apply one-hot encoding when loading the grocery dataset into our Python script. The rationality behind using one-hot encoding stands in the fact that the Apriori Algorithm operates on binary data structure to calculate the support, confidence and lift all three of which will be explained shortly. The one-hot encoding transformation did precisely that. It transformed our transactional data into a binary format represented by a binary matrix. Each row of the matrix stands for a specific transaction, and each column represents one of 166 unique items. If item y is part of transaction x, then the cell [x,y] would have a value of 1 indicating that the item is present in the transaction. Contrarily, if the item is not part of the transaction then the value of the cell will be 0.

As mentioned previously, the Apriori Algorithm is applied by using the MLxtend library. It first identifies frequently occurring sets and then generates association rules, which are the core of our recommendation system. These rules are the ones responsible for predicting items that the customer might want to purchase based on their current selections. The way the algorithm is implemented directly influences the accuracy and relevance of the recommendations. To achieve performance optimization, so that it better suits our specific dataset and project objectives, we can adjust the algorithm's parameters. Specifically: support, confidence, and lift.

In order for us to understand how adjusting the algorithm’s parameters leads to optimized performance, we first need to know what these parameters are. Support is the parameter determining the frequency threshold, which is responsible for deciding if a set of items should be considered frequent. Confidence measures how likely it is for one item to be purchased when another item is already in the cart. Its purpose is to ensure reliable recommendations that are generated based on strong association. Finally, lift is the parameter that determines the strength of an association rule by comparing the observed frequency of the sets of items to their expected frequency if these items were independent, highlighting particularly strong associations.

When we talk about items that are likely to be purchased together with other items, we are referring to the terms antecedents and consequents. To better understand these terms, we will think of a scenario where an association rule indicates that it is highly likely that when customers have bread and milk in their carts, they will proceed to purchase butter as well. In this case, bread and milk are considered to be the antecedents and butter is considered to be the consequent. There are multiple ways of explaining antecedents and consequents. The previous scenario helps with the idea of antecedents being the if part of an if else statement and the consequents are the else part.

With the intention of optimizing the algorithm's performance and decreasing the computational load, we implemented transaction reduction techniques. As a result, the dataset processing is faster and more efficient. We achieved these effects by eliminating transactions that did not meet the minimum support threshold early in the processing stage, which therefore reduced the overall number of sets of items that required further evaluation.

The back end of our web application is able to seamlessly transfer data between the user's interactions on the user interface and the recommendation logic happening in the back end. Once the user adds one of the many available items to their cart, a POST request containing the updated data in JSON format is sent to the back end. The back end then uses the Apriori Algorithm to process the data received as well as generate new recommendations based on the current contents of the cart or the user’s location aisle wise. After the recommendations are generated, they are sent back to the front end through GET requests. The GET requests dynamically update the user interface to display the new suggestions.

The front end of our web application keeps track of the cart state within the session storage. This storage however, is temporary and once the page refreshes it resets. Currently, we have not implemented a database which would store user-specific information permanently. After analyzing the pros and the cons we decided to take this approach because of the simplicity of initial implementation. If in the future, retaining user data across sessions will be needed it could be implemented using a database system, such as SQLite, PostgreSQL, MySQL, etc. In addition to storing user sessions and cart details persistently, this approach would create possibilities to add additional features such as user authentication, personalized recommendation histories, and retrial of shopping carts across different timely sessions or devices.

As previously stated, apart from implementing the Apriori Algorithm as it has been put into practice before we have improved various aspects of it. One aspect that was improved on our implementation of the Apriori Algorithm is the dynamic adjustment of the thresholds. By dynamically adjusting parameters such as support, confidence and lift we were able to fine-tune the recommendation output based on changing user behaviors and scenarios. In the event of a greater or modified dataset the flexibility achieved through this improvement will keep the algorithm effective, relevant and accurate.

Our back end capabilities include generating predictions tailored to a user’s cart items and offering contextual suggestions based on the aisle they are currently exploring. For example, if a customer is browsing "Personal Care," they may receive suggestions for related products like "make up remover " or "hygiene articles." This dual approach offers both personalized and context-driven recommendations, enhancing user satisfaction and encouraging more purchases.

While designing how the front and back end would be integrated, our main priorities were efficiency and fast responsive interactive experience. When users modify their current state either by adding new items to their cart or changing aisles, JavaScript handles these actions and communicates them to the back end by using asynchronous HTTP requests. After the back end processes this data, it generates new recommendations which are then sent back to the front end through JSON format. While all this is happening, the user experiences no delays and real-time recommendation updates.

We’ve also introduced dynamic item cards that display product details such as name, price, and an "Add to Cart" button. When a product is added to the cart, the back end processes the updated data using the Apriori Algorithm to generate fresh recommendations. These are then sent to the front end to update the user's shopping experience in real time. To improve algorithm performance, we made several adjustments, such as dynamically modifying support, confidence, and lift thresholds. We noticed that when the support threshold was raised, it resulted in reduced sets of items and the focus was on frequently occurring items. On the other hand, when we lowered the confidence threshold, it introduced diverse recommendations tailored to different users.

Another area where we made improvements is in data processing. Our implementation of transaction reduction techniques resulted in the minimization of the high computational load that would be needed to process large amounts of data. We were able to lower the computational load simply by filtering out transactions that did not meet the minimum support threshold. We removed these transactions early in the process, which therefore reduced the number of different items that had to be processed and evaluated. After doing so, our algorithm was able to generate recommendations more quickly, resulting in a faster user experience.

Besides reducing transactions that did not meet the minimum support threshold, we also refined the association rules. This optimization ensures that only the most relevant and meaningful rules are applied when generating recommendations. Therefore, the recommendations generated by our web application are as accurate as possible with the user's purchasing behaviors and preferences. We need a high level of precision if we want to maintain user satisfaction as well as increase the overall sales.

There were some updates to the UI that were vital to improving the user experience. Adding images to the storefront enhances the visual appeal and will improve user engagement. It’s important to align the visual items for better recommendation accuracy and customer satisfaction. There are over 100 items in the storefront and each of them has a name, price, and image. It would be tedious to manually download each item JPG and update the image path in the JSON file. In order to automate the process, we implemented the Unsplash API to retrieve images for each item based on queries. We created a directory where the downloaded images will be saved, and we inserted our API key to authenticate requests to Unsplash. The program imports the necessary libraries: requests for making API calls, JSON for reading and writing to the JSON file, is for handling file paths and directories, and time for implementing delays to manage rate limits.

We start by loading the grocery data JSON file into a Python dictionary. Within the fetch images script, there is a helper function that takes a search item and a file path as inputs. The function sends requests to the Unsplash API to fetch a random image that matches the item name/search term. The script iterates through the grocery items in the JSON data. For each item, it checks if an image path already exists, and if it does not contain the default placeholder, “placeholder.jpg”, the item is skipped. This indicates that there is an image for the item already. Otherwise, the script formats the item name into a URL-friendly search term, defines the local file path, and attempts to download the image. For items successfully downloaded, the script updates the image field in the JSON data with the file name. If the request is unsuccessful, the image field retains the placeholder JPG.

Due to there being a rate limit, the program cannot complete all the image requests in one run. In order to work around this, we implemented the time library. The program processes 50 requests before being put on cooldown for 3600 seconds and then runs again. This repeats until every item in the grocery data JSON file is processed. After processing all the items in the grocery data JSON file, the updated JSON data is saved back into the original file, ensuring the image paths reflect the downloaded images. This script handles errors, enforces API rate limits, dynamically manages directories, and persists changes to the JSON file, making it a solution for associating grocery items with relevant images.

Future improvements to the image grab integration system could focus on data storage enhancements, enhanced image matching, batch processing optimization, and user feedback integration. These enhancements would not only improve the immediate functionality of the system but also future-proof adaptability and user satisfaction.

**Conclusion (½ page)**

This research showed the potential of the Apriori Algorithm to build a more intelligent recommendation system based on grocery store datasets. After finding meaningful patterns in transaction data, we were able to generate personalized and helpful product suggestions to make shopping easier for customers. To optimize the efficiency and accuracy of the algorithm, modifying its parameters and optimizing the input data, it allowed the system to be able to work with bigger datasets and give a more diversified recommendation.

This also means that the system was designed to be user-friendly, having smooth integration from the back end to the front end for real-time interactive updates. While this current version performs very well in a controlled environment, extending the features to track the sessions with the use of machine learning could make the system even more personal and adaptable. These improvements would help it perform better in real-world scenarios and cater to the needs of a larger audience.

Overall, our project's goal was to provide insight into the application of various tools focusing on the Apriori Algorithm. We implemented it in a way that it can be used to solve real-world problems in retail. It combines technical skills with a focus on the enhancement of user experience and would, therefore, contribute to future advancements in AI-powered recommendation systems.

**Works Cited**

A. Sharma and H. Babbar, "Analysis of Data Mining Algorithms in Market Basket Analysis,"

2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, 2023, pp. 275-280, doi: 10.1109/InCACCT57535.2023.10141816. keywords: {Data analysis;Filtering algorithms;Market research;Minimization;Prediction algorithms;Information filters;Data mining;Data Mining;Market Basket Analysis;Apriori;FP-Growth;Architecture},

Berg-Kirkpatrick, Taylor, and Alexandre Boucharde-Cote. “Painless Unsupervised Learning with

Features.” *Aclanthology*, UC Berkeley, June 2010, aclanthology.org/N10-1083.pdf.

D. H. Setiabudi, G. S. Budhi, I. W. J. Purnama and A. Noertjahyana, "Data mining market basket

analysis' using hybrid-dimension association rules, case study in Minimarket X," 2011 International Conference on Uncertainty Reasoning and Knowledge Engineering, Bali, Indonesia, 2011, pp. 196-199, doi: 10.1109/URKE.2011.6007796. keywords: {Association rules;Itemsets;Marketing and sales;Testing;Algorithm design and analysis;Data Mining;Market Basket Analysis;Apriori;Hybrid-Dimension Association Rules}.

D. Yang, "Data Mining Algorithm Data Model Data Analysis Based on Artificial Intelligence

Technology," 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), Tumkur, Karnataka, India, 2022, pp. 1-4, doi: 10.1109/ICMNWC56175.2022.10031763. keywords: {Wireless communication;Analytical models;Data analysis;Neural networks;Big Data;Prediction algorithms;Data models;Data Mining;Neural Network;Abnormal Data Detection;Data Processing},

Mahesh, Batta. “(PDF) Machine Learning Algorithms -A Review.” *Research Gate*, Jan. 2020,

www.researchgate.net/publication/344717762\_Machine\_Learning\_Algorithms\_-A\_Review.

Mart&iacute;n, Alejandro G., et al. “A Survey for User Behavior Analysis Based on Machine

Learning Techniques: Current Models and Applications - Applied Intelligence.” *SpringerLink*, Springer US, 26 Jan. 2021, [link.springer.com/article/10.1007/s10489-020-02160-x](http://link.springer.com/article/10.1007/s10489-020-02160-x).

I. N. M. Adiputra and P. Wanchai, "Customer Churn Prediction Using Weight Average Ensemble

Machine Learning Model," 2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE), Phitsanulok, Thailand, 2023, pp. 90-94, doi: 10.1109/JCSSE58229.2023.10202105. keywords: {Machine learning algorithms;Computational modeling;Insurance;Machine learning;Companies;Predictive models;Prediction algorithms;customer churn prediction;machine learning;ensemble model;XGBoost;random forest},

Nasteski, Vladmir. “An Overview of the Supervised Machine.” *Research Gate*, Faculty of

Information and Communication Technologies, www.researchgate.net/profile/Vladimir-Nasteski/publication/328146111\_An\_overview\_of\_the\_supervised\_machine\_learning\_methods/links/5c1025194585157ac1bba147/An-overview-of-the-supervised-machine-learning-methods.pdf. Accessed 16 Nov. 2024.

R. Saravanan and P. Sujatha, "A State of Art Techniques on Machine Learning Algorithms: A

Perspective of Supervised Learning Approaches in Data Classification," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2018, pp. 945-949, doi: 10.1109/ICCONS.2018.8663155. keywords: {Training;Neurons;Supervised learning;Computational modeling;Data models;Support vector machines;Artificial neural networks;Classification Problem;ML;Reinforcement Learning;Supervised Learning;Training Process},

Sarker I. H. (2022). AI-Based Modeling: Techniques, Applications and Research Issues Towards

Automation, Intelligent and Smart Systems. *SN computer science*, *3*(2), 158. <https://doi.org/10.1007/s42979-022-01043-x>

Singh, Jaishree, et al. “Improving Efficiency of Apriori Algorithm Using Transaction Reduction

.” *Cite SeerX*, PSU, Jan. 2013, [citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2c69a4643c6d686145ad5874d8a09364648e605e](http://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2c69a4643c6d686145ad5874d8a09364648e605e).

Suprianto Panjaitan *et al* 2019 *J. Phys.: Conf. Ser.* 1255 012057

T. P. Martin and B. Azvine, "Learning user models for an intelligent telephone assistant,"

Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569), Vancouver, BC, Canada, 2001, pp. 669-674 vol.2, doi: 10.1109/NAFIPS.2001.944682. keywords: {Telephony;Prototypes;Home appliances;Mobile handsets;Speech recognition;Text recognition;Artificial intelligence;Pervasive computing;Logic;System testing}.

Xu, Yongjun, et al. “Artificial Intelligence: A Powerful Paradigm for Scientific Research.”

*Innovation (Cambridge (Mass.))*, U.S. National Library of Medicine, 28 Oct. 2021, www.ncbi.nlm.nih.gov/pmc/articles/PMC8633405/.

Y. Bai and D. Wang, "Estimate and Predict the Foreign Currency Exchanging Rates by Using

Adaptive Neuro Fuzzy Inference System and Deep Learning Algorithms," 2024 4th International Conference on Information Communication and Software Engineering (ICICSE), Beijing, China, 2024, pp. 65-69, doi: 10.1109/ICICSE61805.2024.10625700. keywords: {Fuzzy logic;Deep learning;Training;Adaptation models;Adaptive systems;Software algorithms;Prediction algorithms;ANFIS algorithm;deep learning;estimate and predict exchanging rates for currencies;AI applications in financial implementations},

Yuan, Xiuli. “An Improved Apriori Algorithm for Mining Association Rules.” *AIP Publishing*,

AIP Publishing, 13 Mar. 2017, pubs.aip.org/aip/acp/article/1820/1/080005/886234/An-improved-Apriori-algorithm-for-mining.