FPT Education
FPT UNIVERSITY

DAP391m

Project

RFMP-based Collaborative Filtering for Product Recommendation System.

Vuong Quynh Anh
quynhanhvuong8050@gmail.com

Lecturer: Do Thai Giang

Abstract:

This project proposes an integration between behavioral segmentation and collaborative filtering to recommend products that best match each customer's purchasing preferences. In addition, the project aims to optimize the economic problem for businesses by adjusting the corresponding weighting factors. We present a hybrid recommendation system built upon two submodules: RFMP clustering, which incorporates the features Recency, Frequency, Monetary and Purchase Periodicity; and a Neighborhood-based Collaborative Filtering (NBCF) model enhanced by AHP weights. The RFMP component captures customers' purchasing behavior, while the NBCF component leverages similarities between customers to generate personalized product suggestions. Using the AdventureWorks2022 database – which contains comprehensive information about transactions, customers and products, we constructed two datasets for our model including RFMP dataset and NBCF dataset. These two datasets were then preprocessed and time-based split into train and test set, 80-20 percent respectively. In final stage, An A/B testing experiment shows a slight improvement from the proposed hybrid model compared to the Global Popularity baseline, measured by whether customers actually purchase the recommended products or not. This result highlights the potential of this research direction and opens opportunities for further enhancement in order to better align with business objectives.

Keywords – Hybrid Recommendation System, RFMP clustering, Neighborhood-based Collaborative Filtering, A/B Testing, Utility Theory.

Table of Contents

Chapter 1: Introduction

1.1. Overview:

Currently, trade is booming across all sectors, ranging from household goods and food to electronics, computers, and mobile phones. As the quality of life improves, customer shopping needs have also become more diverse. At the same time, as shopping demand increases, supply will also grow significantly. This leads to intense competition among businesses. Now they need to solve the problem: How can they retain customers so that customers always choose to shop at their store?

I believe the solution to this problem lies in recommending products to customers. Just as customers tend to buy more when a marketing representative is very good at selling, I consider product recommendation to be the core solution to the problem of retaining customers.

To apply machine learning algorithms to solve the product recommendation problem, we often see approaches such as content-based filtering or collaborative filtering [1]. The common point of these approaches is that they all need to rely on the company's data sets to compile a sufficiently good user profile, from which machine learning techniques can then be applied to predict that customer's future shopping needs.

Regarding the construction of a complete user profile, I believe that if we collect data on online platforms, it will be easier to extract features such as click rate, watch time, search history, or even shopping cart activities [2]. However, if we collect data at traditional stores or direct sales businesses, building user profiles becomes more difficult because we don't have much information that can be used to summarize customer preferences or habits [3]. Given this reality, I would like to research and propose a solution for compiling user profiles in situations where data is scarce, thereby building a product recommendation system suitable for direct sales.

I believe that this goal can achieved with the help of the machine learning techniques [4]. By identifying patterns in customers' buying history, the system can predict which products are the most suitable based on what each customer used to give attention to. The accuracy of the algorithm can be further improved by applying weights developed from economic theories, helping to reach the exact demands from business tactics. In this project I will study and propose a methodology to this research direction.

1.2. Literature Review:

Considering the publications on the product recommendation problem, we often see that their systems are built based on filtering problems (content-based, collaborative, behavior-based, hybrid, etc.).

Delving deeper into these filtering techniques, we see that the biggest weakness of collaborative filtering (CF) – the most widely used technique – is the cold start problem, meaning we cannot recommend products to completely new customers because we cannot classify them into any user group. To overcome this difficulty, we have two approaches: first, using a hybrid filtering model (combining CF with another filtering technique) [1] [5] [6]; second, using clustering or segmentation models before applying appropriate filtering techniques [7].

For the first approach, Xu et al. showed that their content-based x collaborative filtering approach could partially improve cold start and increase accuracy [1]. Xu et al. analyzed mainly based on users' browsing history, including orders, shopping carts, searches, and user reviews. The data was then trained using the BERT language model to extract user features, followed by Next Purchase Prediction (NPP) to predict which product would be the best next choice for the user. Although effective, this model requires significant resources and is highly dependent on user behavior data.

Similarly, Chakraborty also proposed using a hybrid recommender architecture by integrating three models: CF, content-based filtering (CBF), and neural network models [5]. In addition, Chakraborty proposed an optimization scheme using metaheuristic algorithms and hyperparameter tuning, thereby achieving very good results in his experiments. This method showed notable improvements in metrics such as Precision, Recall, F1-Score, RMSE, etc. More importantly, the improvements achieved are considered highly suitable and aligned with the initial marketing objectives. However, this approach also shares the same drawback as Xu et al., which is the requirement for a very large dataset.

Liu et al. also proposed a hybrid recommender system, but they used two types of collaborative filtering: user-user based CF and item-item based CF [6]. Based on ratings and co-purchasing networks, Liu et al. built a graph-based recommendation system with products as nodes and the correlation in the frequency of co-purchasing between two products (two nodes) as edges. Although it is also a hybrid recommender, this system still cannot overcome the cold start problem, as it has not yet come up with a specific solution for completely new customers or new products that have never been purchased before.

For the second approach, Irina proposed using clustering algorithms as a step to preprocess issues such as data sparsity and cold start [7]. After clustering, the data also gains diversity (adding new features). After completing the clustering, we return to the CBF, CF, or hybrid problem. A metaheuristic algorithm was used for optimization. In addition to improving accuracy, Irina's model can also be applied to real-time data, something neither traditional CF nor CBF can do. The data requirements for this algorithm are moderate, yet it can still achieve good accuracy.

1.3. Objectives:

In this project, I propose a research approach to enable personalized product recommendations based on user profiles compiled from limited data source, while still meeting the needs of each business's marketing strategy:

- Compile user profiles based on customer transaction data.

- Build a hybrid model to recommend products to customers based on each business's marketing strategy.

This research may not be complete enough to form a complete pipeline, but it will provide a direction for combining machine learning techniques with economic theories with the desire to improve the actual metrics that businesses aim to improve.

Chapter 2: Related Works

2.0. Expected Utility Theory in Recommendation Systems:

The Utility Theory (UT) is a fundamental concept in economics that describes how individuals make their decisions to maximize their "satisfaction level" (utility) when choosing between options [12]. In the context of a recommendation system, each product can be seen as a "option", and in order to maximize the utility, we have to find the best options that maximize the utility. For each individual/customer, we build the utility function with different weight values:

$$U_{customer}(product) = f(price, category, brand, similarity, \ldots)$$

Each customers' personalized utility function can be modeled to reflect their preferences. Used in this recommender model, the utility function for a customer c to a product p can be modeled as:

$$U_{u(p)} = \alpha \times RFMP_c \times MPR_p + \beta \times CF_{(c,p)}$$

in which:

- RFMP score x Monetary/Price (MPR) captures customers' past purchasing behaviour, which reflects their affordability and spending patterns.

- CF score represents the expected utility, calculated from user-user based similarity through collaborative filtering algorithms.

- alpha and beta stands for the relative importance between past behavioral utility (RFMP) and predicted preference utility (CF).

Based on this theoretical foundation of the Utility Theory, I will experiment from practical implementation. Each of these variables and weights will be calculated and optimized in machine learning algorithms. At the end of this project, I hope to propose a full-functioned framework that gives precise recommendations based on this utility function.

2.1. RFMP clustering.

The RFM (Recency – Frequency – Monetary) analysis technique is a popular method for analyzing customer behavior in marketing and data mining [8]. The goal of this model is to evaluate the value of each customer based on their transaction history through three key indicators:

- Recency (R*): represents the time period from the most recent purchase to the present. A smaller R value indicates that the customer is “newer” and more likely to make a repeat purchase.

$$R^* = \sum_{i=1}^{f}\left(D_i - D_0\right)$$

in which: f represents the total number of transactions of a customer, D_0 represent the first transaction date and D_i represents the i-th transaction date. This Recency formula is changed to utilize the most out of our dataset. [9]

- Frequency (F): represents the number of purchases within a specified time period. A higher F value indicates a higher level of customer engagement with the business.

- Monetary (M): represents the total amount spent by the customer during that period. The higher the M, the more profit the customer generates.

These three factors are typically standardized, scored, and aggregated to classify customers by value and loyalty level, thereby informing marketing or customer care strategies.

However, the traditional RFM model only describes purchasing behavior in three basic aspects and does not reflect the cyclicality and regularity of purchasing behavior. To overcome this limitation, Saedi et al. (2025) expanded this model into RFMP, adding a fourth factor: Periodicity (P) [9].

$$Periodicity = stdev\left(IVT_1, IVT_2, \ldots, IVT_{n-1}, IVT_n\right)$$

Periodicity (P) measures the regularity of customer purchasing frequency, typically calculated using the standard deviation of the time interval between consecutive purchases. If P is small, it means that customers exhibit regular purchasing behavior, indicating high loyalty. If P is large, it means purchasing behavior is irregular and less stable.

By adding the P factor, the RFMP model provides a more comprehensive description of the customer lifecycle and purchasing behavior. While RFM only reflects “how long, how much, and how much money”, RFMP adds the ability to assess “whether customers purchase regularly”, thereby enhancing the ability to identify loyal customers and potential for long-term value retention.

2.2. Neighborhood-based Collaborative Filtering.

The Neighborhood-based Collaborative Filtering (NBCF) method is one of the basic and intuitive approaches in recommender systems. The main idea is: if two users or two items have similar behavior in the past, it is very likely that they will have similar preferences for other items in the future [10]. For example: if person A and person B both give 5 stars to the movie “Detective”, and person A also likes “The Judge”, then it can be predicted that person B will also like “The Judge”.

Technically, NBCF works in two main steps: (1) identifying “neighbors” by finding users or items similar to the user/item to be recommended; (2) using the ratings or interactions of those neighbors to predict the target user's level of interest in an unrated item. In the user-user approach, we find users similar to user u and then look at their ratings of item i to predict how u will rate i. In the item-item approach, instead of comparing users, we compare items: we find items similar to item j and then rely on user u's ratings of similar items to predict whether u will like j.

Chapter 3: Methodology

3.1. Dataset:

3.1.1. AdventureWorks2022 database review.

AdventureWorks is a sample database for Microsoft SQL Server 2008 to 2014, created by Dataedo [11]. This database supports standard online transaction processing scenarios for a fictitious bicycle manufacturer – Adventure Works Cycle. Scenarios include Manufaturing, Sales, Purchasing, Product Management, Contact Management and Human Resources.

Although it is only a fictional database, AdventureWorks includes many modules that closely resemble real-world scenarios, with data on Business Entities, People, Human Resources, Products, Manufacturing, Purchasing, Inventory, Sales, and Admin. The realism of this data is a major plus, because the challenges businesses face in reality are likely to be similar to those we encounter in the database.

3.1.2. Dataset collection.

From a large database (consisting of 9 modules and 71 tables) [11], I selected the necessary data. We decided to use RFMP clustering and CF, so we also needed to prepare two different datasets.

For the RFMP dataset, we need to aggregate transaction data from the Sales tables to calculate the R, F, M, and P scores for each customer. We will calculate these scores as follows:

- Find Recency by calculating the number of days between purchases and finding the average.

- Find Frequency by counting the total number of orders the customer has purchased.

- Find Monetary by calculating the average value of orders over the total number of orders the customer has purchased.

- Find Periodicity by taking the standard deviation of the IVTs, where IVT (inner-visit time) is the interval between purchases in days. [9]

After performing the above calculations, I obtained the RFMP dataset:

	CustomerID	Recency	Frequency	Monetary	Purchase Periodicity
0	11000	1565	3	2114.955300	394.662303
1	11001	1792	3	3029.169700	366.656970
2	11002	1502	3	2351.395833	403.317906
3	11003	1601	3	2314.584600	394.976371
4	11004	1559	3	2988.671433	395.992845
...	...	...	...	...	...
19114	30114	2561	8	650.697225	32.333088
19115	30115	2559	8	65.527900	32.295234
19116	30116	551	4	1272.045625	45.835758
19117	30117	6028	12	1363.044450	26.376068
19118	30118	2552	8	283.247450	32.281961

19119 rows × 5 columns

Figure 1: Example of RFMP dataset.

For the NBCF dataset, we need the utility matrix. As mentioned in the related works section, the NBCF problem usually uses the rating points of each product that customers have purchased as the main parameter of the utility matrix, but in this case we will not use ratings, because the AdventureWorks dataset does not have customer ratings on products.

This is a disadvantage for our analysis, but as I mentioned, the difficulties we encounter in this dataset are very likely to be the same problems we encounter in real-world cases. Indeed, for traditional stores or direct sales businesses, we usually have no way to collect customer reviews after they have purchased a product. This will be a difficulty that we need to overcome. In this project, I propose using the Frequency score for each product to replace the rating score. More specifically, we will take the total number of times that product appears in an orderfrom a customer, then divide it by the number of orders. For example, customer A has purchased for a total of 6 orders. In those 6 orders, there are 3 orders where customer A purchased product B. So we have a frequency score for product B for customer A is 3/6 = 0.5. The higher the frequency score, the greater the likelihood that this customer will repurchase that product, which also means that this customer likes that product. Based on the above reasoning, I decided to replace the Rating score with Frequency. After performingthe above calculations, I obtained the utility matrix:

	CustomerID	ProductID	Frequency
0	11000	707	0.125000
1	11000	771	0.125000
2	11000	779	0.125000
3	11000	878	0.125000
4	11000	881	0.125000
...	...	...	...
79428	30118	989	0.010381
79429	30118	990	0.013841
79430	30118	991	0.006920
79431	30118	992	0.010381
79432	30118	993	0.013841

79433 rows × 3 columns

Figure 2: Example of NBCF utility matrix

And these are the two main datasets that we will be using in this project.

3.1.3. Normalization and removing outliers.

- For the RFMP dataset: We will need to perform a fine normalization for this data, as the R, F, M, and P indices vary greatly. This statement is proved in the following chart:

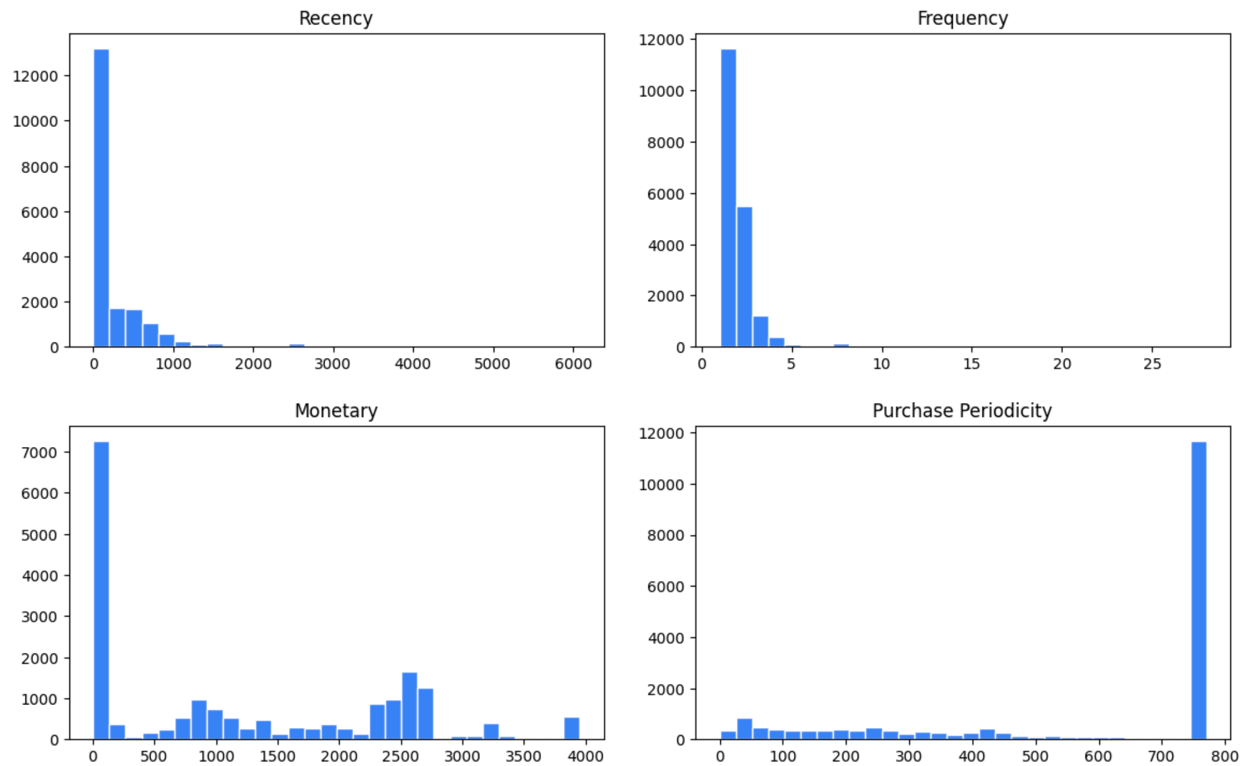

Figure 3: RFMP dataset plot

It can be seen that the values of R, F, M, and P are skewed at a very large margin. These figures show that customers mainly make only one purchase (Frequency = 1) and do not repurchase. The Monetary values also show that a large number of customers only make purchases of low value. Recency and Periodicity are also greatly inflated at both ends, mainly because if customers do not repurchase, there is no way to calculate a sufficiently accurate IVT.

Although the dataset is realistic, it is not suitable for immediate machine learning training; we need to clean this data first. My RFMP dataset cleaning process consists of two parts:

First, we standardize the values R, F, M, and P. It is clear that the values R, F, and P are not too large, so we can apply Z-Score (StandardScaler) to standardize them.

For the value M, because these values are much larger than the values of R, F, and P, we will apply LogScale to M.

Next, we need to remove noise by removing outliers. In this project, I chose to use the Interquartile Range (IQR) method to detect and remove outliers in the data. Considering the lower bound as Q1, the upper bound as Q2, and taking the coefficient as 1.5, we remove the noise data.

After removing the noise, we will also remove the NaN values.

Finally, we get a much better RFMP dataset.

	CustomerID	R	F	M	P
count	19119.000000	17768.000000	1.832600e+04	19119.000000	19119.000000
mean	20559.000000	0.386980	-9.925731e-17	6.021322	0.000536
std	5519.324234	0.788611	1.000000e+00	1.903186	0.999450
min	11000.000000	-0.585632	-7.008367e-01	0.928417	-2.048070
25%	15779.500000	0.000000	-7.008367e-01	4.168146	-0.944793
50%	20559.000000	0.000000	-7.008367e-01	6.820029	0.730554
75%	25338.500000	0.357840	9.288465e-01	7.787642	0.730554
max	30118.000000	3.153313	2.558530e+00	8.282480	0.730554

Figure 4: RFMP dataset after normalization and outliers removal.

- Regarding the NBCF utility matrix: This is also an advantage of replacing Rating with Frequency, because the Frequency score will always be in the range [0,1]. We don't need to further standardize this dataset, as it is already well-prepared.

3.1.4. Splitting dataset into train and validate set.

In this project, I will use A/B testing to compare Hit Rates as a test for the project's results. To simulate real-world scenarios, I decided to split the train and validate sets based on time. Specifically, I will calculate the 80-20 split point for transactions, using 80% of the time as the train set – simulating the “present” time, and the remaining 20% of the time as the validate set – simulating the “future” time.

Additionally, since the testing problem is A/B testing, we also need to split the train and validate sets into two distinct sets, A and B, to apply two different recommendation systems, thereby enabling a comparison of results.

After splitting the data, I obtained a total of 4 datasets: A_train, A_validate, B_train, and B_validate.

3.2. Model:

3.2.0: Utility function.

$$\boldsymbol{final\_score = \alpha \times RFMP\_score \times MPR + \beta \times NBCF\_score}$$

in which:

- final_score represents the final predicted products’ rating that customer A have not bought.
- NBCF_score used the CF method to find and it represents the predicted products’ rating that customer A have not bought. Calculation methodology will be speficied in the next session (Chapter 3.2.3).
- RFMP_score represents the customer’s RFMP score. Calculation methodology will be speficied in the next session (Chapter 3.2.2.)
- MPR is calculated as the customer’s Monetary divided by the products’ Prices. This weight aims to decrease the final_score of products which are more expensive than the average order value of that customer.

As I have mentioned in Chapter 2.0., to prodive recommendations for a specific customer, we will apply this function to calculate the final score. The top k-th products which have the highest final score will be determined as the products that best fit the customer's need and will be recommended to that customer.

3.2.1. Analytic Hierachy Process (AHP) priority.

Before analyzing specific models, we need to walk through the AHP weights first. AHP is a multi-criteria decision-making method developed by Thomas Saaty. In a simple way of understanding, when there are multiple features to consider, AHP helps us systematically calculate the importance level (in the form of weights) of each feature.

The application of AHP weights to the RFMP clustering model was also mentioned by Saedi et al. in their customer segmentation project [9]. In Saedi et al.'s project, AHP weights act as weights to fine-tune the importance of each feature R, F, M, P, thereby developing the Customer Lifetime Value (CLV) for each customer.

In this project, we will also use AHP weights as a weighting factor to refine the influence of features in the model. For example, if we want customers to have a high Frequency score (customers who return to purchase multiple times), we will prioritize the Frequency metric over metrics from other features such as R, M, P. The pairwise claim indices will be determined by business experts, thereby determining the outcome our model will aim for.

We propose calculating AHP weights for each feature R, F, M, P using the following steps:

- Input is the pairwise comparison matrix: Experts will evaluate the relative importance between criteria based on the Saaty scale.
- Calculate priority weights: Calculate the eigenvector to derive the relative weight of each criterion. Weights are normalized to sum to 1, reflecting the influence of each criterion in the overall picture.
- Recheck using the Consistency Ratio (CR): Calculate the CR to ensure that the evaluators' judgments are not contradictory ($CR < 0.1$ means the results are consistent and reliable; if $CF > 0.1$, the pairwise matrix needs to be adjusted).

3.2.2. RFMP clustering.

In their clustering problem, Irina et al. used the K-Means algorithm [7]. Similarly, Saedi et al. also used K-Means for their customer segmentation problem. However, K-Means only works well when the data clusters are roughly spherical (points are equidistant from the center) because this algorithm uses Euclidean distance to calculate similarity between data points. As previously analyzed, our dataset is heavily skewed, so applying K-Means for clustering will yield poor results.

I propose using the Gaussian Mixture Model (GMM) as an alternative. For skewed, flat, or overlapping data, GMM performs better than K-Means. Additionally, GMM uses the Expectation-Maximization (EM) algorithm, so it is less likely to get stuck in local optima.

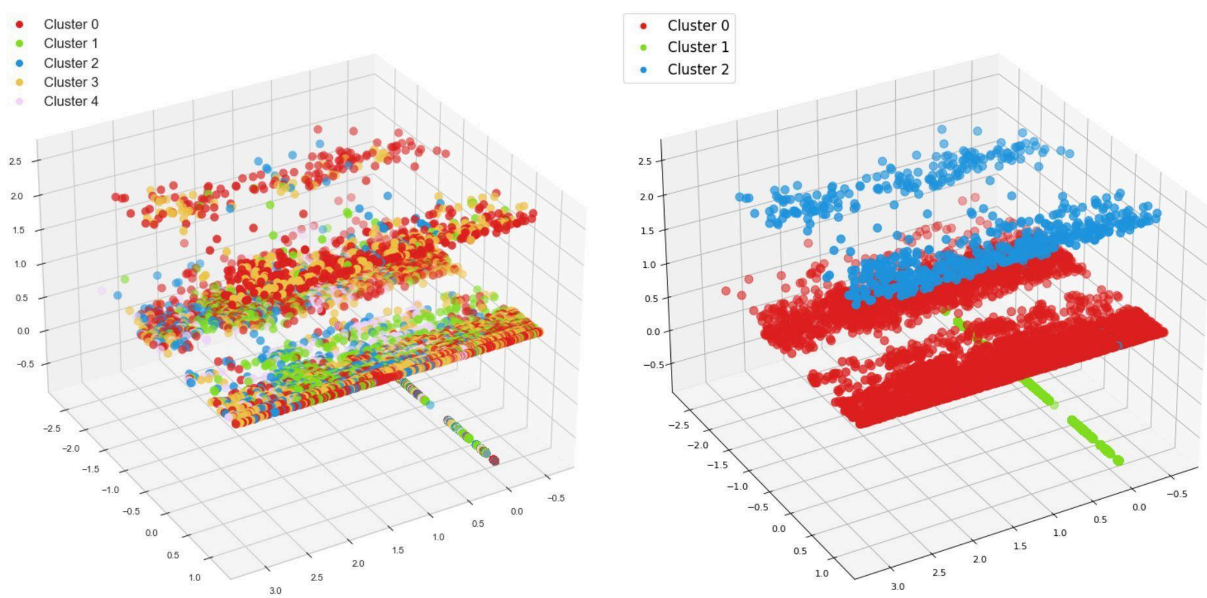

Figure 5: *KMeans k=5 clustering (left),*
GMM k=3 clustering (right)

The left plot shows the results of the KMeans trial run. The optimal k value was found using the Elbow Method, but we can see that the results are not good. The data of the clusters overlap each other and lack the necessary generality for each cluster.

The plot on the right shows the results of the GMM trial run. The optimal k value was found using the Silhouette and David Bouldin indices. The classification results are relatively good, with the data points divided into clusters in a very intuitive manner.

After running GMM to classify, we find the RFMP_score value by calculating the average of the R, F, M, P scores of each customer in each cluster; and then calculate the average RFMP of each individual in the cluster to obtain the final RFMP_score representing that entire user cluster.

This segmentation problem will give us an overview of customers, classifying them into groups with different potentials. Based on the results of this problem, you can make clearer analyses of each group:

Cluster	R	F	M	P	RFMP_Score	Clusters' analysis
0	1.05	0.93	6.20	-1.09	7.09	Recent transactions, frequent purchases, and fairly high average order value.
1	0.00	-0.70	5.86	0.73	5.89	Non-frequent shoppers (R=0; no repurchases), recently inactive, average spending.
2	1.36	2.55	6.45	-1.64	8.71	Frequent purchases, very high average order value.

To visualize this more simply, we can name each cluster based on their characteristics. Cluster 2 is likely the most loyal VIP customers of the business, with characteristics such as high R, very high F (significantly higher than the other two groups), high M (high spending), and low P (regular purchase frequency). Cluster 0 consists of customers with good potential. Most of them have fairly good R scores, average F scores indicating moderate purchasing levels, high M scores indicating high spending levels, and low P scores indicating fairly consistent purchasing frequency. For this cluster of customers, we need to mark them as customers with the potential to become loyal customers if they are cared for and provided with sufficiently good services. Cluster 1 alone is a collection of inactive customers. R = 0 indicates that it has been a long time since they made a purchase, or even that they have only made one purchase, while F has a negative value. M is quite high, indicating that they have spent a large amount of money but have not returned to make further purchases. P is positive, indicating a long purchase frequency. This is the group with the highest risk of churn and needs special attention if the business aims to retain these customers.

The RFMP score represents the customer's behavior and is calculated as described above.

3.2.3. Neighborhood-based collaborative filtering.

When discussing about CF, the main idea is: To predict a user's (U) rating for an item (I) that they have not interacted with, we will find a set of neighbors for user U, and then use the average rating of the neighbor set to infer U's rating for I. [10]

There are two approaches to finding neighbors: user-user based, which finds other users with preferences similar to U; or item-item based, which finds other items with characteristics similar to those that U has purchased/rated.

After constructing the utility matrix (Chapter 3.1. Dataset), we will choose a method to calculate similarity. In this project, I decided to choose user-user based CF. From a mathematical perspective, I set the goal of predicting the frequency (rating) $\hat{y}_{u,i}$ from user u for item i. The steps to calculate this value include:

- Normalization: This is an important step that Ekstrand et al. mentioned to handle bias. However, as I mentioned in the previous sections, since the frequency value only ranges from 0 to 1, we do not need to worry about bias differences between customers.

- Calculate user-user similarity: Use cosine similarity to find the “neighbor” users v that are most similar to user u. We will calculate sim(u,v) for every pair of users (u,v) that we have.

- Neighborhood selection: We can select k customers v with the highest similarity sim(u,v). From there, we collect N(u,i), which is a list of neighbors v of u who have also purchased products similar to u.

- Rating prediction: Rating prediction can be calculated as we find the average of weighted and normalized ratings from N(u,i).

$$\hat{y}_{u,i} = \bar{y}_u + \frac{\sum_{v \in N_{u,i}} sim(u,v)(y_{v,i} - \bar{y}_v)}{\sum_{v \in N_{u,i}} |sim(u,v)|}$$

in which can be simply explained as: Predicted rating = (Average rating of u) + (Total weight of rating differences from neighbors).

+) $y_{v,i} - \bar{y}_v$: Mức độ chênh lệch (chuẩn hóa) mà user v (hàng xóm) đánh giá item i.

+) $sim(u, v)$: Trọng số (weight) - v càng giống u thì ý kiến của v càng quan trọng.

NBCF score ($\hat{y}_{u,i}$) is the predicted rating of user u for each product i in the list of products that u has not purchased, and has been calculated as above.

Chapter 4: Experiments and Results.

In this project, I used A/B testing as the experimentation to evaluate the results of this research approach. A/B test is a randomized controlled trial. In a simple manner, we split our customers' dataset into 2 random groups:

- Group A (Base-line/Benchmark model): We applied on the baseline model (in this project I used Global Popularity as the baseline model).
- Group B (Treatment model): We applied on the new model that uses the new approach that we have walked through in this report (RFMP-based Collaborative Filtering).

Usually we can test in a limited time (example: 1 week, 1 month, 1 year…) and calculate real-world metrics namely click-through rate (CTR), conversion rate (CVR), total revenue… etc. At the end of the test, we answer the question: "Is group B brings the higher revenue? Are customers from group A or B more satisfied with our recommendations (that can be proved if the customer is buying these recommended products or not)?" If the answer is yes, group B does increase some of these metrics, it means that our B model won and should be implemented in real-life situations.

In a course project manner, we could not manage to apply A/B testing on a online environment (the dataset updates continuously), so I proposed a 80-20 time-based split, so that I can mimic the real-world scenarios. As I chose a split point, I will then split them as a train and a test dataset and make sure that the data leakages will not happen. By doing this, I can build a realistic simulation. For example, today is 10/9, I used all of what I had knew (train set), and then I check if the model's prediction is good or not in 11/9 (test set). For more information, please comeback to Chapter 3.1. Dataset where I had speficically explained this splitting method.

Through backtesting on a time-series splitted dataset, the RFMP-based Collaborative Filtering model (treatment model) demonstrated a slightly improved performance compared to the Global Popularity (baseline model) in the CTR metric.

Chapter 5: Conclusion and Future Works.

5.1. Conclusion

This project has researched and proposed a novel approach to the product recommendation system problem, focusing specifically on data-scarce scenarios lacking explicit customer ratings. By integrating RFMP (Recency, Frequency, Monetary, Periodicity) behavioral analysis with Neighborhood-based Collaborative Filtering (NBCF), the project has successfully built a hybrid model capable of delivering personalized recommendations.

The main contributions of this project include:

1. **Handling Scarce Data:** Instead of relying on traditional "ratings," the model successfully proposed and used the "Frequency" score (purchase frequency over total orders) as an implicit metric to build the Utility Matrix. This resolves the core problem faced when deploying systems in traditional stores or direct sales businesses, which lack the infrastructure to collect customer reviews.
2. **Advanced Behavioral Clustering:** By extending the traditional RFM model to RFMP and applying the Gaussian Mixture Model (GMM) algorithm , the project overcame the weaknesses of K-Means on skewed data. It successfully segmented customers into meaningful groups (e.g., "Loyal VIPs," "High Potential," "At-Risk"). The resulting RFMP_score serves as a component that captures past customer behavior.
3. **Hybrid Utility Model Construction:** The project proposed a utility function that combines behavioral scores (RFMP_score and MPR) with preference prediction scores (NBCF_score). This allows the system to recommend products that are not only "similar" but also considerate of the customer's "affordability" and "purchasing patterns".
4. **Performance Validation:** Through an A/B testing (backtesting) experiment on a time-based split dataset to mimic real-world scenarios [11], the hybrid RFMP×NBCF model (Group B) demonstrated a slight but clear improvement in the CTR (0.1078) compared to the Global Popularity baseline model (Group A, 0.0815).

Although the improvement is modest, this result confirms the potential of this research direction, proving that integrating behavioral analytics into collaborative filtering is a valid and feasible approach to enhance the effectiveness of a recommendation system.

5.2. Future Works

Based on these findings, two primary directions are proposed for future enhancement. First, the Analytic Hierarchy Process (AHP), which was introduced theoretically, should be practically implemented. This involves collaborating directly with business experts to define pairwise comparisons that align the model's weights (e.g., for R, F, M, P) with specific strategic goals, such as prioritizing customer retention (weighting F, P) or maximizing revenue (weighting M).

Second, Content-Based Filtering (CBF) must be integrated to solve the critical "Cold Start" problem , which our current NBCF model cannot handle for new users or new products. By building product profiles from attributes like category and price, CBF can provide immediate, relevant recommendations to new users and ensure new items are discoverable. This enhancement will evolve the system into a true hybrid, covering the entire customer and product lifecycle and creating a more robust, strategic, and comprehensive recommendation solution.

References

[1] Kangming Xu, Huiming Zhou, Haotian Zheng, Mingwei Zhu, Qi Xin. Intelligent Classification of E-commerce Products Based on Machine Learning (2024).

[2] Xiaolei Li on International Journal of High Speed Electronics and Systems (3/2025). An Accurate Construction Method of E-commerce user Profile Bsed on Artificial Intelligence Algorithm and Big Data Analysis.

[3] Hamsa Bastani, Pavithra Harsha, Georgia Perakis on Machine Learning & Science Operations Management (2021). Learning Personalized Product Recommendations with Customer Disengagement.

[4] Priyadarsini Patnaik, Parameswar Nayak, Siddharth Misra on Advanced Research and Real-World Applications of Industry 5.0 (2023). Personalized Product Recommendation and User Satisfaction: Reference to Industry 5.0.

[5] Sanjukta Chakraborty on International Journal of Advances in Business and Management Research (2025). A study on hybrid recommender system for effective targeted marketing in e-commerce platforms.

[6] Mark Liu, Catherine Zhao, Nathan Zhou. Building a Recommendation System Using Amazon Product Co-Purchasing Network (2025).

[7] Beregovskaya Irina, Koroteev Mikhail on Financial University under the government of Russian Federation. Review of Clustering-based Recommender System.

[8] Sydul Arefin, Rezwanul Parvez, Tanvir Ahmed, Mostofa Ahsan, Fnu Sumaiya, Fariha Jahin, Munjur Hasan on 2024 IEEE (eIT). Retail Industry Analytics: Unraveling Consumer Behavior through RFM Segmentaion and Machine Learning.

[9] Abdolah Saedi, Meysam Abbasi, Alieh Mehdi Nejad on International Journal of New Political Economy 6 (2025). Customer segmentation using the extended RFMP model based on Customer lifetime value (CLV) and data mining.

[10] Michael D. Ekstrand, John T. Riedl, Joseph A. Konstan Foundation and Trends in Human-Computer Interaction (2010). Collaborative Filtering Recommender Systems.

[11] Dataedo. Repository: AdventureWorks.

[12] Elsevier Ltd. (2001). International Encyclopedia of the Social & Behavioral Sciences: Economics and Ethics.

[13] Amy Gallo (June 2017) on Analytics and Data Sciences. A Refresher on A/B Testing.

[14] Alexandra Roshchina, John Cardiff, Paolo Rosso on SAAIP, IJCNLP (2011). User profile construction in the twin personality-based recommender system.