Personalized Recommendation Using Machine Learning

By Balla Rakesh, Chethan H N, Shubhra saxena, Pavan Yidapalapti

DATE: 23-11-2022

1. Abstract

Personalized recommender systems **provide recommendations according to the user profile and estimate users' preferences or interests in a particular domain**. Since user profiles reflect the users' informational needs, it plays a fundamental role in recommender systems.

Personalized Recommender Systems are software tools and techniques providing suggestions for items to be of use to a user. The suggestions provided are aimed at supporting their users in various decision-making processes, such as what items to buy, what music Development of recommender systems is a multi-disciplinary effort which involves experts from various fields such as Artificial intelligence, Human Computer Interaction, Information Technology, Data Mining, Statistics, Adaptive User Interfaces, Decision Support Systems, Marketing, or Consumer Behavior.

Recent growth in online business has resulted in increased use of different personalized services to develop one to one customer relationships, effective marketing, and to attract users and retain customers and to boost repeat purchase rates, drive sales, and enhance the conversion rates.

Ecommerce websites and Social Media Platforms use personalization as an effective strategy by providing one to one services like product recommendation, information and ratings of the product by satisfying individual users' needs.

Personalized Product Recommender systems are particularly useful when an individual needs to choose an item from a potentially overwhelming number of items that a service may offer, Personalization can be based on a customer's previous purchases, browsing behaviour, geographic location, language and other personal information.

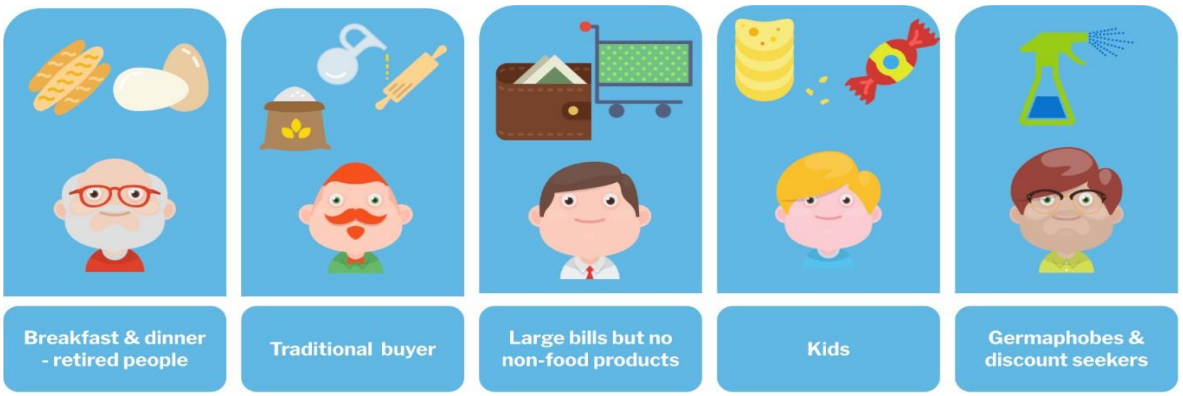
Ecommerce Websites like Flipcart.com and Amazon.in provide product recommendations based on collaborative filtering techniques and also suggest some frequently buy items with people having similar interest in products. User's' navigational behavior and search is analyzed and extracted knowledge is used to target users by showcasing advertisements.

Collaborative filtering, content-based filtering, data mining methods and context-aware methods are used to build Personalized Product recommender systems.

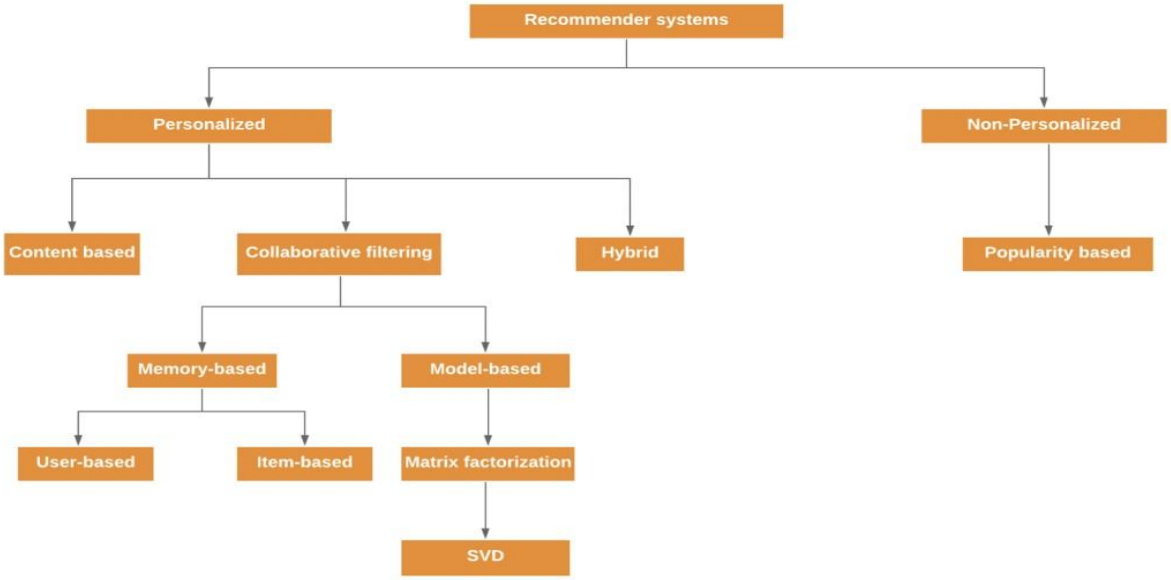
2. Problem Statement

- The objective of recommender systems is to provide recommendations based on recorded information on the users' preferences, previous Purchases, browsing behaviour, geographic location, language and other personal information.
- These systems use information filtering techniques to process information and provide the user with potentially more relevant items, to accomplish that, collaborative filtering and content-based Filtering systems are used For Personalized Recommendation system.
- To develop a recommendation system which enables bidirectional communication between the user and system using a utility range-based product recommendation algorithm in order to provide more dynamic and personalized recommendations.

3. Introduction



A **recommender system**, or a **recommendation system** is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item.
In the last decade companies have invested a lot of money in their development. Netflix awarded a $1 million prize to a developer team in 2009 for an algorithm that increased the accuracy of the company's recommendation engine by 10 percent.



Picture 1 – Types of recommender systems

There are two main types of recommender systems – personalized and non-personalized

3.1 Non-personalized recommendation

Non-personalized recommendation systems like popularity-based recommenders recommend the most popular items to the users, for instance top-10 movies, top selling books, the most frequently purchased products.

What is a good recommendation?

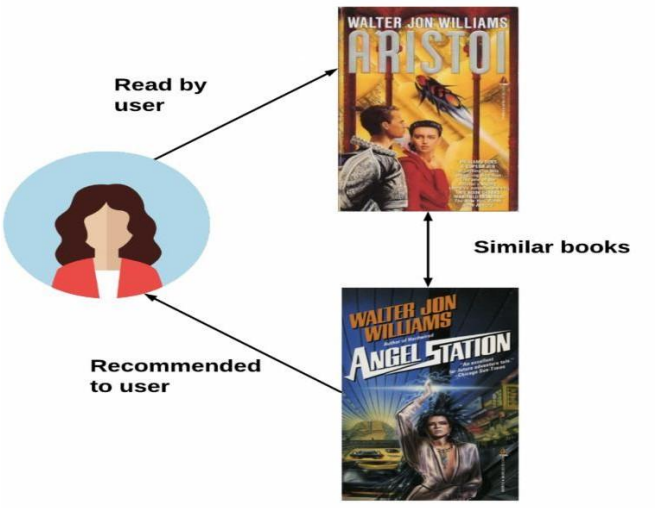
- The one that is personalized (relevant to that user)
- The one that is diverse (includes different user interests)
- The one that doesn't recommend the same items to users for the second time
- The one that recommends available products

3.2 Personalized recommendation systems

Personalized recommender system analyses users data, their purchases, rating and their relationships with other users in more detail. In that way every user will get customized recommendations.

The most popular types of personalized recommendation systems are content based and collaborative filtering.

3.2.1 Content based personalization recommendation system



Picture 2 – Content based recommender system

Content based recommender systems use items or users metadata to create specific recommendations. The user's purchase history is observed. For example, if a user has already

read a book from one author or bought a product of a certain brand it is assumed that the customer has a preference for that author or that brand and there is a probability that user will buy a similar product in the future.

Assume that Jenny loves sci-fi books and her favourite writer is Walter Jon Williams. If she reads the Aristoi book, then her recommended book will be Angel station, also sci-fi book written by Walter Jon Williams.

Collaborative filtering in practice gives better results than content-based approach. Perhaps it is because there is not as much diversity in the results as in collaborative filtering.

Disadvantages of content-based approach:

- There is a so-called phenomenon filter bubble. If a user reads a book about a political ideology and books related to that ideology are recommended to him, he will be in the "bubble of his previous interests".
- lot of data about user and his preferences needs to be collected to get the best recommendation
- In practice there are 20% of items that attract the attention of 70-80% of users and 70-80% of items that attract the attention of 20% of users. Recommender's goal is to introduce other products that are not available to users at first glance. In a content-based approach this goal is not achieved as well as in collaborative filtering.

3.2.2 Collaborative filtering personalization recommendation system

The idea of collaborative filtering is simple: User group behaviour is used to make recommendations to other users. Since the recommendation is based on the preferences of other users it is called collaborative.

There are two types of collaborative filtering: memory-based and model based.

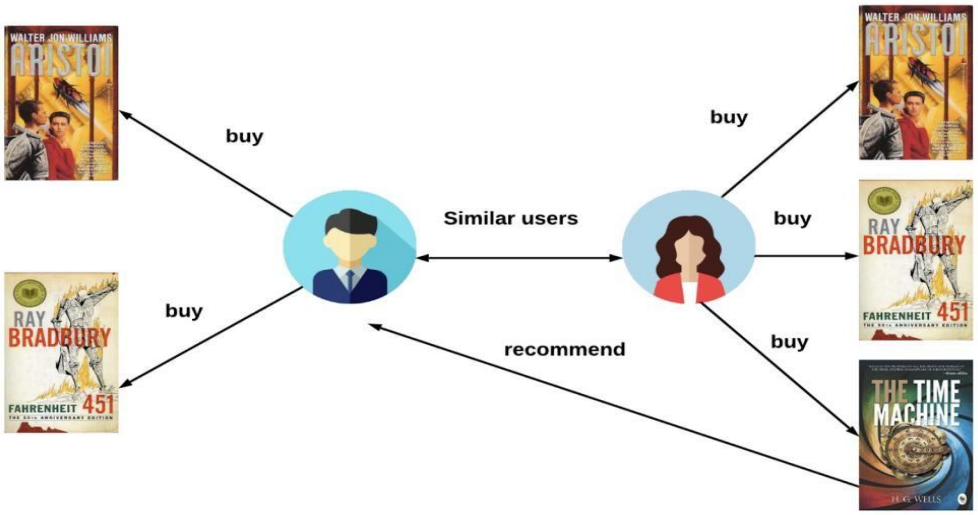
3.2.2.1 Memory based Collaborative filtering

Memory based techniques are applied to raw data without pre-processing. They are easy for implementation and the resulting recommendations are generally easy to explain. Each time it is necessary to make predictions over all the data which slows down the recommender.

There are two types of Memory Based: user based and item based collaborative filtering.

3.2.2.1.1 User based Collaborative filtering

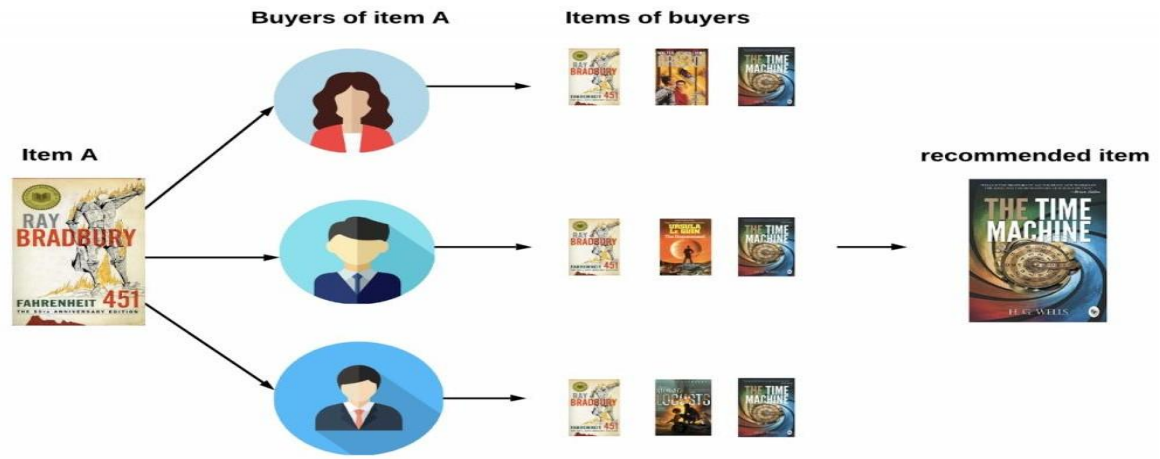
"Users who are similar to you also liked…" Products are recommended to the user based on the fact that they were purchased / liked by users who are similar to the observed user. If we say that users are similar what does that mean? For example, Jenny and Tom love sci-fi books. When a new sci-fi book appears and Jenny buys that book, since Tom also likes sci-fi books then we can recommend the book that Jenny bought.



Picture 3 – User based collaborative filtering recommender system

3.2.2.1.2 Item based Collaborative filtering

"Users who liked this item also liked…" If John, Robert and Jenny highly rated sci-fi books Fahrenheit 451 and The time machine, for example gave 5 stars, then when Tom buys the book Fahrenheit 451 then the book The time machine is also recommended to him because the system identified books as similar based on user ratings.



Picture 4 – Item based collaborative filtering recommender system

How to calculate user-user and item-item similarities?

Unlike the content-based approach where metadata about users or items is used, the collaborative filtering memory-based approach user behavior is observed, e.g. whether the user liked or rated an item or whether the item was liked or rated by a certain user.

For example, the idea is to recommend Robert the new sci-fi book.

Steps:

1. Create user-item-rating matrix
2. Create user-user similarity matrix
3. Cosine similarity is calculated (alternatives: adjusted cosine similarity, pearson similarity, spearman rank correlation) between every two users. In this way a user-user matrix is obtained. This matrix is smaller than the initial user-item-rating matrix.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

4. Look up similar users
5. In the user-user matrix, users that are most similar to Robert are observed
6. Candidate generation
7. When Robert's most similar users are found, then we look at all the books these users read and ratings they gave.
8. Candidate scoring
9. Depending on the ratings, books are ranked from the ones that Robert's most similar users liked the most, to the ones they liked the least.
10. The results are normalized (on a scale from 0 to 1)
11. Candidate filtering
12. It is being checked whether Robert has already bought any of these books. Those books should be eliminated because he has already read it.

The calculation of item-item similarity is done in an identical way and has all the same steps as user-user similarity.

Comparison of user-based and item-based approaches

The similarity between items is more stable than the similarity between the users because the math book will always be a math book, but the user can change his mind, e.g. something he liked last week he might not like next week. Another advantage is that there are fewer products than users. This leads to the conclusion that an item-item matrix with similarity scores will be smaller than a user-user matrix. Also item-based is a better approach if a new user visits the site while the user-based approach is problematic in that case.

3.2.2.2 Model based Collaborative filtering

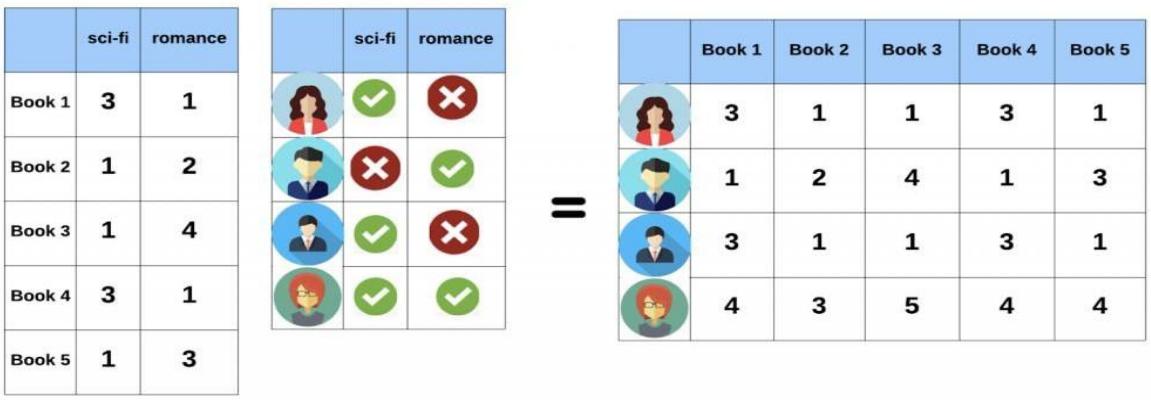
These models were developed using machine learning algorithms. A model is created and based on it, not all data, gives recommendations, which speeds up the work of the system. This approach achieves better scalability. Dimensionality reduction is often used in this approach. The most famous type of this approach is matrix factorization.

Matrix factorization

If there is feedback from the user for example, a user has watched a particular movie or read a particular book and has given a rating, that can be represented in the form of a matrix where each row represents a particular user and each column represents a particular item. Since it is almost impossible that the user will rate every item, this matrix will have many unfilled values. This is called sparsity. Matrix factorization methods are used to find a set of latent factors and determine user preferences using these factors. Latent Information can be reported by analysing user behaviour. The latent factors are otherwise called as features.

Why factorization?

Rating matrix is a product of two smaller matrices – item-feature matrix and user-feature matrix.



Picture 5 – Matrix factorization

Matrix factorization steps:

1. Initialization of random user and item matrix
2. Ratings matrix is obtained by multiplying the user and the transposed item matrix
3. The goal of matrix factorization is to minimize the loss function (the difference in the ratings of the predicted and actual matrices must be minimal). Each rating can be described as a dot product of row in user matrix and column in item matrix.

$$\min_{Q*,P*} \sum_{(u,i)\in K} (r_{ui} - P_u^T Q_i)^2 + \lambda(||Q_i||^2 + ||P_u||^2)$$

Picture 6 – minimization of loss function

Where K is a set of (u,i) pairs, r(u,i) is the rating for item i by user u and λ is a regularization term (used to avoid overfitting).

1. In order to minimize loss function we can apply Stochastic Gradient Descent (SGD) or Alternating Least Squares (ALS). Both methods can be used to incrementally update the model as new rating comes in. SGD is faster and more accurate than ALS.

Hybrid recommenders

They represent a combination of different recommenders. The assumption is that a combination of several different recommenders will give better results than a single algorithm.

RECOMMENDER SYSTEMS METRICS

Which metric will be used depends on the business problem being solved. If we think that we have made the best possible recommender and the metric is great, but in practice it is bad, then our recommender is not good. Netflix recommender was never used in practice because it did not meet customer needs. The most important thing is that the user gains confidence in the recommender system. If we recommend him the top 10 products, and only 2 or 3 are relevant to him, he will consider that the recommender system is bad. For this reason, the idea is not to always recommend top 10 items, but to recommend items above a certain threshold.

3.3 Evaluation techniques

Now, we will introduce different techniques that evaluate whether the model overfits or underfits. The ultimate goal for any model is to perform well for any future data. So, how do we go about this? The dataset that is used is divided into two sections: training and test. The training data is used to train the model while test data is used to evaluate it. In an ideal situation, we segregate the dataset in the ratio 8:2 with 80% of training and 20% is used for test. Letting the data to be distributed non-linearly and fitting it with a linear model can lead the data to be underfitting and this model does not work well with training data. Meanwhile, overfitting performs well with training data but performs badly with test data. Here model fits

well over the data distribution area. Some of the evaluation methods we have used are as follows:

3.3.1 Cross Validation

Cross-validation is a statistical method used to estimate the skill of machine learning models. Two types of cross-validation can be distinguished: exhaustive and no exhaustive cross-validation. Exhaustive cross-validation includes leave-one-out and leave-out cross-validation. On the other hand, non-exhaustive cross-validation includes k-fold cross-validation, Holdout method and Repeated random sub-sampling validation. In this thesis, we have used k-fold cross-validation. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k =10 becoming 10-fold cross-validation.

Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. That is, to use a limited sample in order to estimate how the model is expected to perform in general when used to make predictions on data not used during the training of the model. It is a popular method because it is simple to understand and because it generally results in a less biased or less optimistic estimate of the model skill than other methods, such as a simple train/test split.

The general procedure is as follows:

- Shuffle the dataset randomly.
- Split the dataset into k groups
- For each unique group: Take the group as a hold out or test data set Take the remaining groups as a training data set Fit a model on the training set and evaluate it on the test set Retain the evaluation score and discard the model
- Summarize the skill of the model using the sample of model evaluation scores Importantly, each observation in the data sample is assigned to an individual group and stays in that group for the duration of the procedure.

This means that each sample is given the opportunity to be used in the hold out set 1 time and used to train the model k-1 times.

Root Mean Square Error (RMSE) is a standard way to measure the error of a model in predicting quantitative data. Formally it is defined as follows:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\vartheta_i - y_i)^2}{n}}$$

Here, $\hat{y}1, \hat{y}2, \ldots, \hat{y}n$ are predicted values. $y1, y2, \ldots, yn$ are observed values. n is number of observations The division by n under the square root in RMSE allows us to estimate the standard deviation σ of the error for a typical single observation rather than some kind of "total error". By dividing by n, we keep this measure of error consistent as we move from a small collection of observations to a larger collection (it just becomes more accurate as we increase the number of observations).

To phrase it another way, RMSE is a good way to answer the question: "How far off should we expect our model to be on its next prediction?" RMSE is a good measure to use if we want to estimate the standard deviation σ of a typical observed value from our model's prediction, assuming that our observed data can be decomposed as:

$observed\ value = predicted\ value + predictably\ distributed\ random\ noise\ with\ mean\ zero$

The random noise here could be anything that our model does not capture (e.g., unknown variables that might influence the observed values). If the noise is small, as estimated by RMSE, this generally means our model is good at predicting our observed data, and if RMSE is large, this generally means our model is failing to account for important features underlying our data.

a. Mean Absolute Error (MAE)
b. MAE is one of the many metrics for summarizing and assessing the quality of a machine learning model. Here, error refers to the subtraction of Predicted value from Actual Value as below.

$$Prediction\ Error = Actual\ Value - Predicted\ Value \quad (15)$$

c. This prediction error is taking for each record after which we convert all error to positive. This is achieved by taking Absolute value for each error as below:

$$Absolute\ Error \rightarrow |Prediction\ Error|$$

Finally, we calculate the mean for all recorded absolute errors (Average sum of all absolute errors).

$$MAE = \sum |yi - xi|\ n\ i{=}1\ n \quad (17)$$

Here, yi is the predicted value, xi is the actual value and n is the number of observations.

3.4 Qualitative and Quantitative Analysis

The comparison among the different systems in this study has two distinct aspects. The quantitative aspect relies on metrics like RMSE and MAE that were described in the previous subsections. But the qualitative aspect relies on the quality of the recommendation and we evaluate it by eyeballing the generated recommendation.

Metrics:

- **Accuracy** (MAE, RMSE)
- **Measure top -N recommenders**:
- **Hit rate** – First find all items in this user's history in the training data; remove one of these items (leave-one-out cross-validation); use all other items for recommender and find top 10 recommendations; If the removed item appear in the top 10 recommendations, it is a hit. If not, it's not a hit.
- **average reciprocal hit rate (ARHR)** – we get more credit for recommending an item in which user rated on the top of the rank than on the bottom of the rank.
- **cumulative hit rate** – those ratings that are less than a certain threshold are rejected, e.g. ratings less than 4
- **rating hit rate** – rating score for each rating is calculated in order to find which type of rating is getting more hits. Sum the number of hits for each type of rating in top-N list and divide by the total number of items of each rating in top-N list.
- **Online A/B testing** – A/B testing is the best way to do online evaluation of your recommender system.

4.Market/Customer/Business need Assessment

- Marketers see an average increase of 20% in sales when using personalized experiences.
- 80% of shoppers are more likely to buy from a company that offers personalized experiences.
- customers might not quickly find what interests them or what they are looking for, and ultimately, they might not make a purchase. To enhance the shopping experience, product recommendations are key.
- More importantly, helping customers make the right choices also has a positive implications for sustainability, as it reduces returns, and thereby minimizes emissions from transportation
- The customer buying preferences have been significantly changed due to the pandemic. Therefore, by using this technique, we aim to provide Online businesses with useful insights from the available data and ways to generate more revenue.

5. Target Specification

- The proposed system/service will provide the Online vendors and Ecommerce Websites with some techniques so that their sales boost up. It will suggest them to group certain items together, based on the analysis performed by the algorithm, so that the customer buys these grouped items together.

  Also, applying certain discount strategies on such grouped items will also increase the sales as required.
- Recommender systems help the users to get personalized recommendations, helps users to take correct decisions in their online transactions, increase sales and redefine the users web browsing experience, retain the customers, enhance their shopping experience
- Recommendations can provide key insights and the opportunity to better understand who a customer is in order to delight hem, add value, and improve the overall relationship with a brand.

6. External Search

The sources I have used as reference for analysing the need of such a system for local businesses and how E-commerce giants and social platforms have been using the technique to boost up online sales, have mentioned below:

- Recommendation systems: Principles, methods and evaluation
- Introduction to recommender systems
- Use Cases of Recommendation Systems in Business
- Recommender Systems: Behind the Scenes of Machine Learning-Based Personalization
- Basic Concepts and Architecture of a Recommender System
- System Architectures for Personalization and Recommendation
- A Comparative Study of Recommendation Systems

7.Benchmarking

Recommend Approach	Source of Knowledge	Type of knowledge	K. extraction method	Drawbacks
Knowledge based	psychographic, demographic, personal attributes of users	Decision Rules	Machine-learning, K. engineer interaction	Bottleneck in K. engineering, subjective user profile, static user profile
Content based	contents of web pages	description of items in the user profile (a set of attributes identifying the items), item-item relationship	document modeling, information filtering, information extraction	overspecialized problem, dependent on the availability of content, syntax-based recommendation (losing semantic meanings)
Collaborative based	other user's profiles (interesting list of other users in the community)	similarity matrix (shared features of other users' preferences in the community)	K-Nearest Neighbor (kNN), Cosine or Correlation based similarity	spare coverage problem, latency state problem, sparsity problem, new item rating problem, new user problem, cold-start problem, violate user privacy
Demographic based	Users' demographic data such as gender, age, date of birth, education, etc	Category membership,	Classification methods, locating group interests	dependent to availability of demographic data, less accurate (poor quality of demographic data)

8. Applicable Patents

- Patent 1 - Personalized recommendations of items represented within a database

It Describes the Enhanced Model for Personalized recommended system using Customer Purchase Behavior, previously purchased, viewed, or placed in an electronic shopping cart by the user.

To generate the table, historical data indicative of users affinities for particular items is processed periodically to identify correlations between item interests of users (e.g., items A and B are similar because a large portion of those who selected A also selected B).

Personal recommendations are generated by accessing the table to identify items similar to those selected by the user. In one embodiment, items are recommended based on the current contents of a user's shopping Cart.

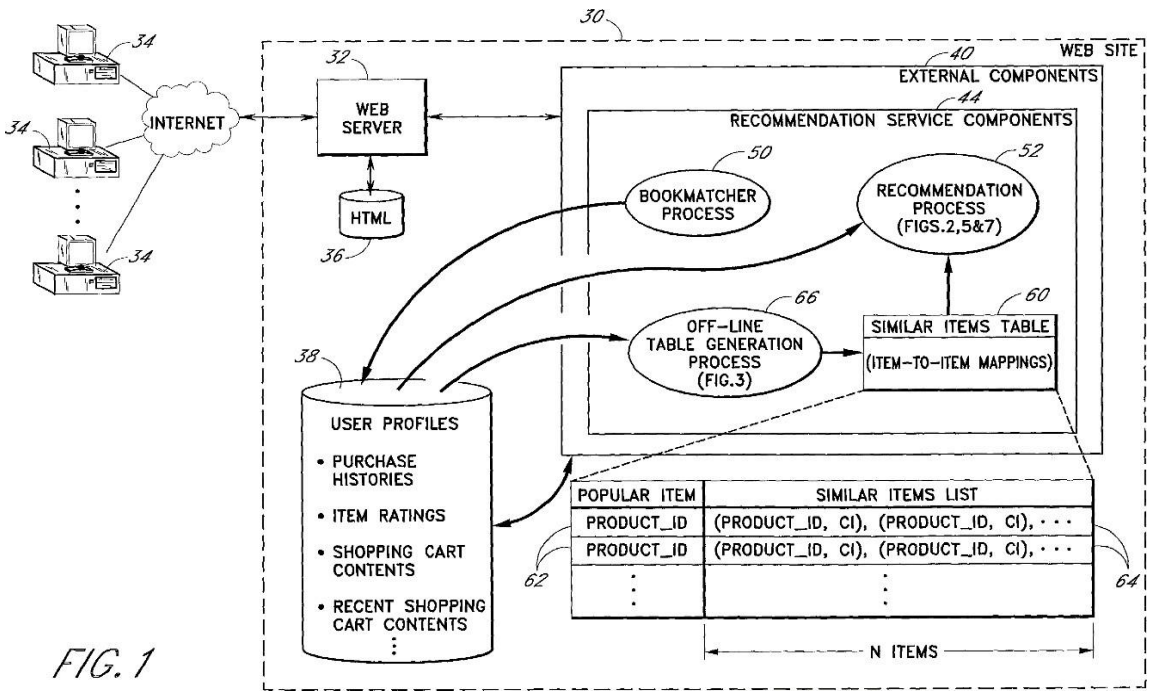
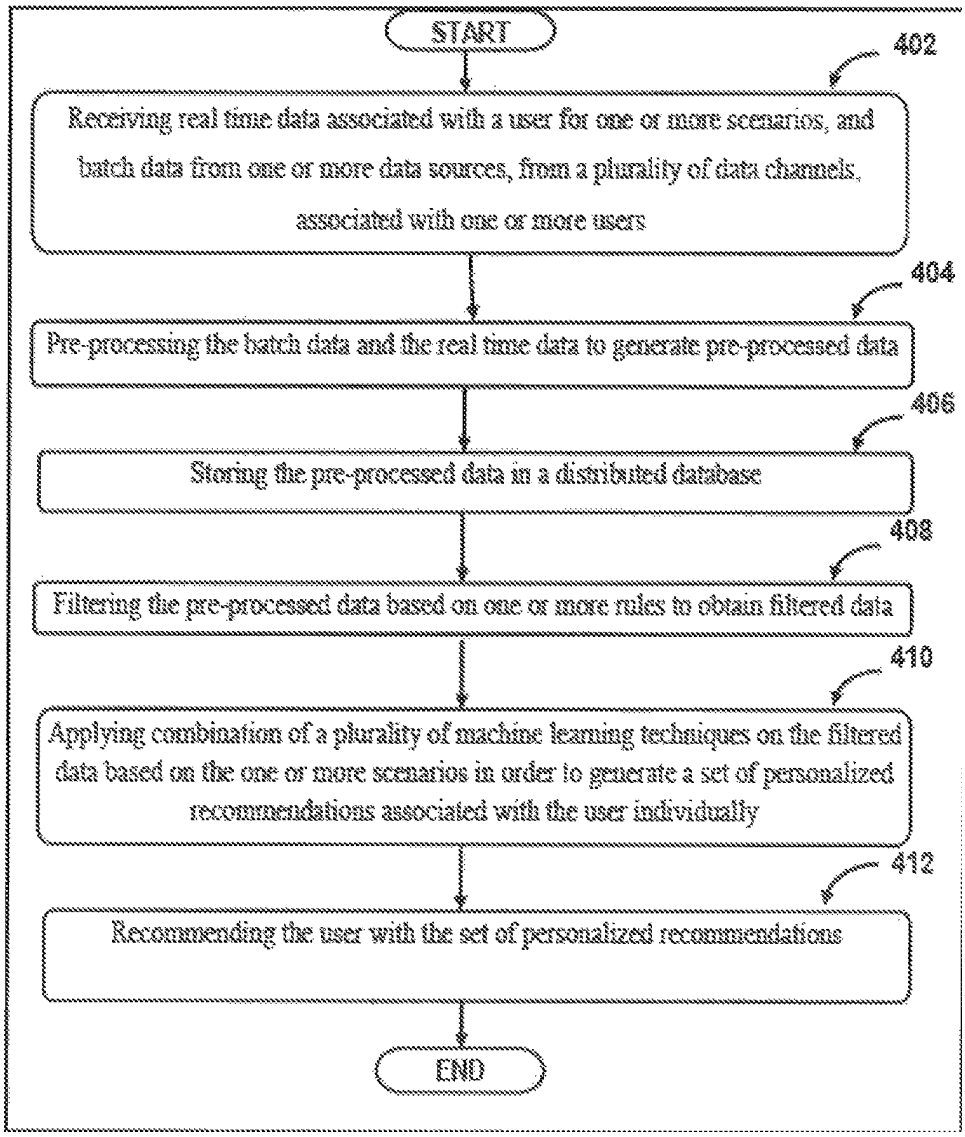


FIG. 1

❖ System and method for generating recommendations

In this Patent it Describes Enhanced Model for generating recommendations to a user. Based on user real time data and Batch data associated with multiple user data, system filter the pre-process this data. Then System applies combination of machine learning techniques on filtered data, based on the scenarios associated with the user, leveraging inter-play between machine learning techniques, to generate personalized recommendations for individual user and storing the personalized recommendations in distributed database.

Machine learning techniques are customized to work in distributed processing mode to reduce overall processing time. System recommends user with the personalized recommendations comprising products or services.



9. Applicable Regulations

- The **General Data Protection Regulation** (**GDPR**) is legislation that updated and unified data privacy laws across the European Union (EU) to protect the User Personal data. The GDPR applies to all 27 member countries of the European Union (EU).
- EU legislation, the **E-Privacy Regulation**, aims to address the legal concerns around user identification technologies, most notably third-party cookies. While GDPR only applies to the processing of personal data, ePrivacy regulates electronic communication even if it concerns non-personal data. Also, in the case of cookies, the ePrivacy generally takes precedence.
- **Privacy legislation** is being is in use in more than a dozen countries around the world, including Brazil, India, Japan, Australia, and, within the US, California to protect the handling of personal information about individuals. This includes the collection, use, storage and disclosure of personal information in the federal public sector and in the private sector.
- **Data Protection Bill** (**DPB**) is introduced in India to promote the protection of personal data; to regulate the manner in which personal data may be processed; to provide persons with rights and remedies to protect their personal data; and to regulate the flow of personal information.

10. Applicable Constraints

- Significant investments required.
- Evaluating different solutions can be enormously time consuming, as you need to evaluate their case studies, the technology, how the solution will be integrated into your current company setup, and so on.
- Lack of understanding of recommendation models and insufficient knowledge of business domain.
- Lack of data analytics capability.
- Inability to capture changes in user behavior.
- Privacy concerns.
- Continuous data collection and maintenance

11. Business Model (Monetization Idea)

- Most websites like Amazon, YouTube, and Netflix use collaborative filtering as a part of their sophisticated recommendation system, Recommender systems help accomplish the marketing goals by presenting items to the users on the basis of personal interests as well as correlations between products. It stimulates more consumption due to the variety of products it can show. It is estimated that around 35% of E-commerce like Amazon and other Social Platforms revenue is derived from its recommendation system.

- Recommender systems are used in a variety of areas, with commonly recognized examples taking the form of **playlist generators for video and music services, product recommenders for online stores, or content recommenders for social media platforms and open web content recommenders**

- The new recommendation system has helped **Spotify** increase its number of monthly users from **75 million to 100 million**.
- **Netflix** makes **75%** of its sales from recommending movies and shows to its customers
- Between 15% to 45% increase in conversion rate from recommendation system for online stores.
- 25% increase in average purchase price by using recommendation in E-Commerce and online stores.

12. Concept Generation

The primary objective of a Recommendation System is to build an objective function or a mathematical model for the end-users and the specific items. This objective function should be able to measure the usefulness of the item for the user.

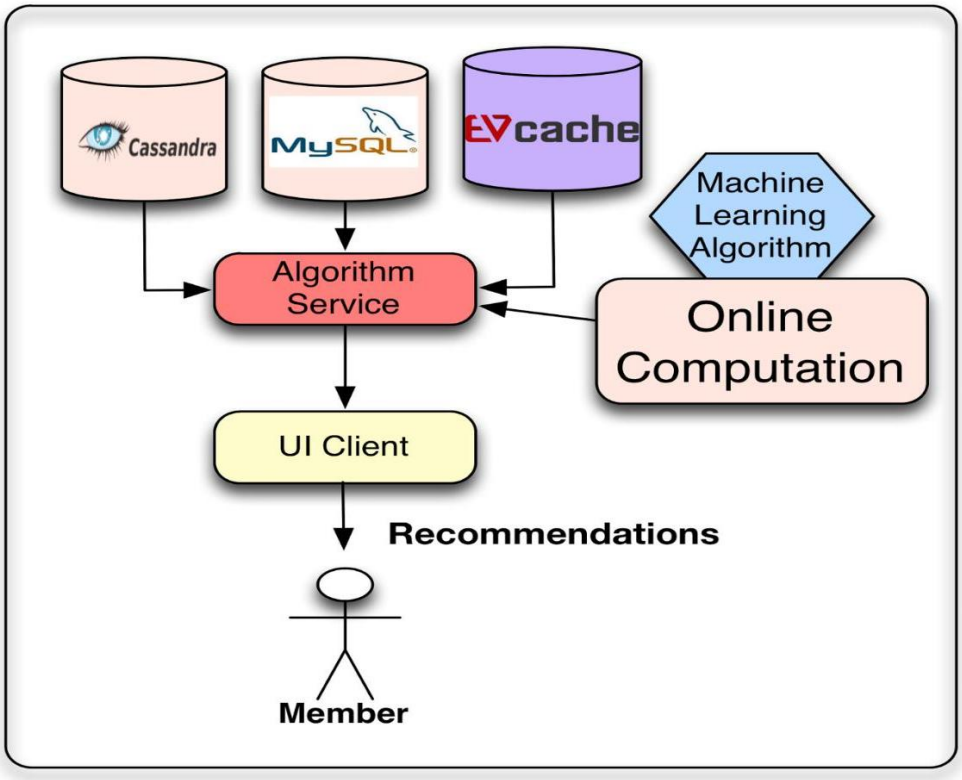
To Build a recommendation system, which can learn users' interests and hobbies according to their profile or historical behaviors, and then predict their ratings or preferences for a given item. **To changes the way businesses, communicate with users and strengthens the interactivity between them**.

Personalized recommendations **drive revenue by positively impacting a customer's total cart amount**. They offer relevant cross-sell and up-sell opportunities that pique a customer's interest, resulting in them purchasing more than just the original item they came in for

13. Concept Development

The Concept Development in building a Personalized Recommendation System are: loading and formatting the data, calculating the similarity between the users or between the items and predicting the unknown ratings for the users. The data can be collected explicitly or implicitly, and the recommendation system turns out to be more optimized with an increase in the available data.

This data can be stored either in **NoSQL** format for unstructured data or as **SQL** tables for structured data. In the latest technology, the cloud is used to store the data and can be easily



retrieved. Data will –be formatted: in most cases, the user-item sets are converted into a matrix known as ratings matrix. Here the end-users are represented by rows while the products are represented by columns, each cell value points to the ratings given to the product by a particular user.

The primary data stores we use are Cassandra, EVCache, and MySQL. Each solution has advantages and disadvantages over the others. MySQL allows for storage of structured relational data that might be required for some future process through general-purpose querying. However, the generality comes at the cost of scalability issues in distributed environments. Cassandra and EVCache both offer the advantages of key-value

stores. Cassandra is a well-known and standard solution when in need of a distributed and scalable no-SQL store. Cassandra works well in some situations, however in cases where we need intensive and constant write operations we find EVCache to be a better fit. The key issue, however, is not so much where to store them as to how to handle the requirements in a way that conflicting goals such as query complexity, read/write latency, and transactional consistency meet at an optimal point for each use case.

14. Product Details- How Does it Work?

Four key phases through which a recommender system processes data. They are information collection, storing, analysis, and filtering. Let's have a closer look at each phase.

Data collection

The initial phase involves gathering relevant data to create a user profile or model for prediction tasks. The data may include such points as the user's attributes, behaviors, or content of the user accesses' resources. Recommendation engines mostly rely on two types of data such as:

- **explicit data** or user input data (e.g., ratings on a scale of 1 to 5 stars, likes or dislikes, reviews, and product comments) and
- **implicit data** or behavior data (e.g., viewing an item, adding it to a wish list, and the time spent on an article, etc.).

Implicit data is easier to collect as it doesn't require any effort from users: You can just keep user activity logs. Though such data is more difficult to analyze. On the other hand, explicit data requires more effort from users, and they aren't always ready to provide enough information. But such data is more accurate.

Also, recommender systems may utilize user attribute data such as demographics (age, gender, nationality) and psychographics (interests) as well as item attribute data (genre, type, category).

Data storing

We have to Provide enough data to train a model. The more quality data there is to feed algorithms with, the more effective and relevant recommendations they will provide.

The next step involves selecting fitting storage that is scalable enough to manage all the collected data. The choice of storage depends on the type of data you're going to use for

recommendations in the first place. This can be a standard SQL database for structured data, a NoSQL database for unstructured data, a cloud data warehouse for both, or even a data lake for Big Data projects. Or you may use a mix of different data repositories depending on the purposes. You can learn more in our dedicated article about a machine learning pipeline.

Data analysis

Data is only useful when it's thoroughly analyzed. There are different types of data analysis based on how quickly the system needs to produce recommendations.

- **Batch analysis** means that data is processed and analyzed in batches – periodically. This can be the analysis of daily sales data.
- **Near real-time analysis** means that data is processed and analyzed every few minutes or seconds but not in real time. These can be recommendations generated during one browsing session.
- **Real-time analysis** means that data comes in streams and gets processed and analyzed as it is created. As a result, a system makes real-time recommendations.

Data filtering — applying algorithms to make recommendations

When building a recommender system, an important aspect is to pick the most appropriate filtering approach (we described them above) and implement the right algorithm to train a model. The algorithms can be simple like the ones to measure the distance between similar items or more complex and resource-heavy ones.

15. Final Product Prototype (Abstract) with Schematic Diagram

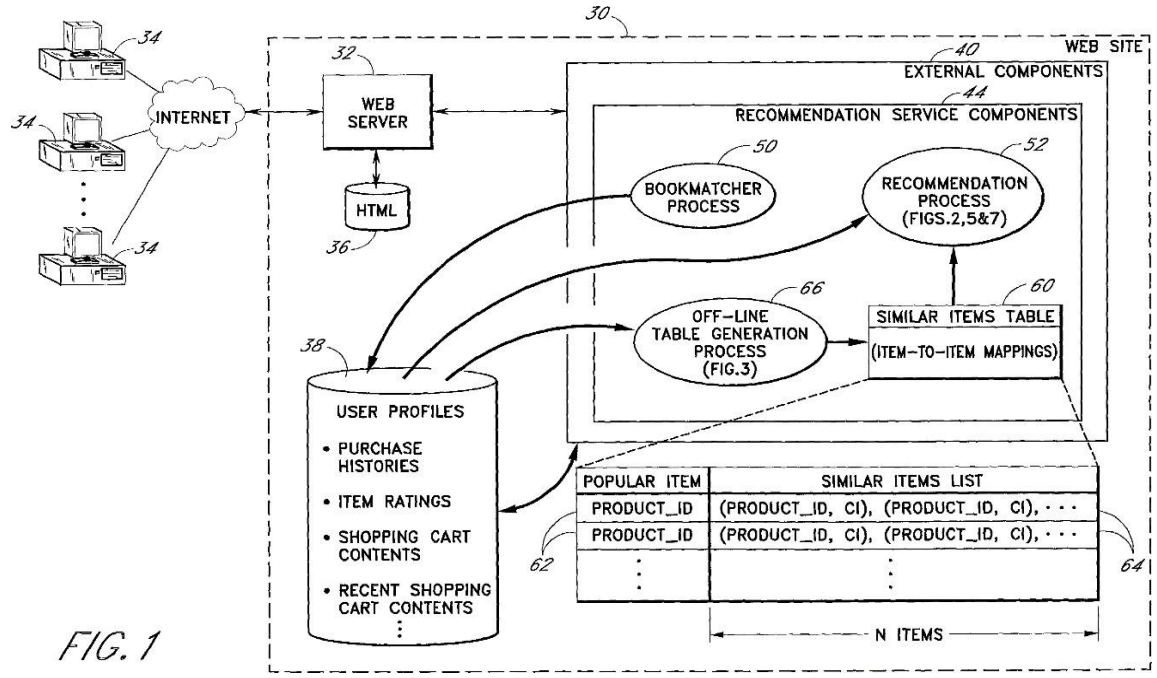
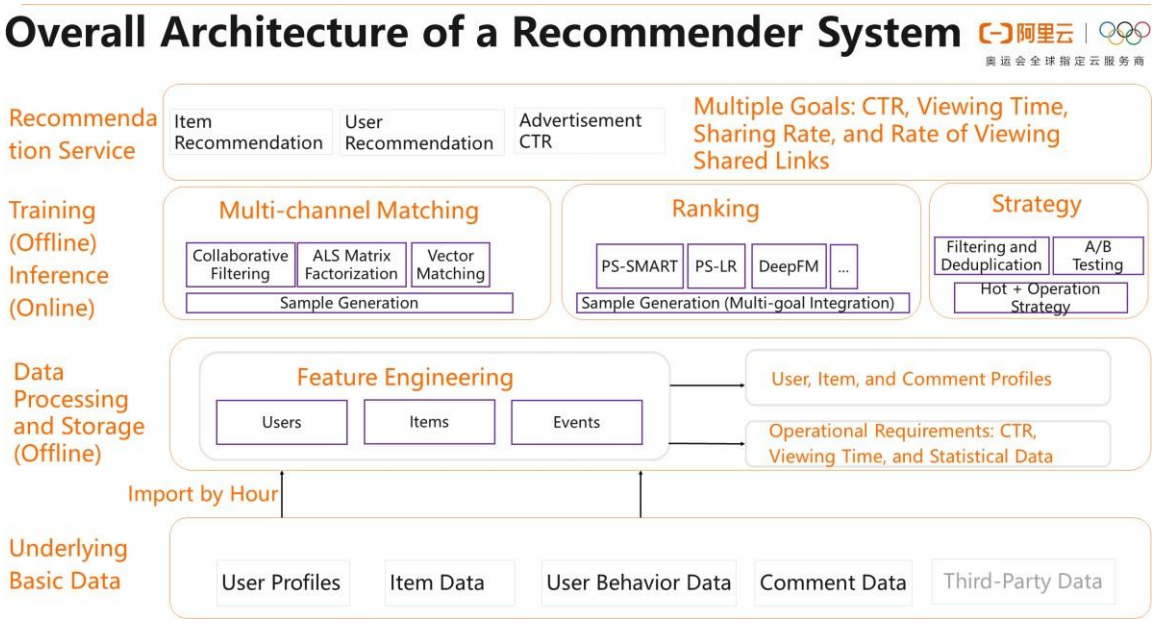


FIG. 1

A computer-implemented service recommends items to a user based on items previously selected by the user, such as items previously purchased, viewed, or placed in an electronic shopping cart by the user. The items may, for example, be products represented within a database of an online merchant. In one embodiment, the service generates the recommendations using a previously generated table that maps items to respective lists of "similar items. To generate the table, historical data indicative of user's affinities for particular items is processed periodically to identify correlations between item interests of users (e.g., items A and B are similar because a large portion of those who selected A also selected B). Personal recommendations are generated by accessing the table to identify items similar to those selected by the user. In one embodiment, items are recommended based on the current contents of a user's shopping Cart.

This Below Picture describes the overall architecture of a recommender system. The following figure shows the underlying basic data layer. This layer contains user profile data, item data, behaviour data, and comment data. The user profile data may be users' heights and weights, items they purchased, their purchase preferences, or their education background. The item data is the prices, colours, and origins of items. If the item is a video, the item data is the information of the video such as the video content and tags. The behaviour data refers to the interaction between users and items.



For example, when a user watches a video, the user may add a like to the video, add the video to favourites, or pay for the video. These actions are all the user's behavior data. The comment data may involve third-party data, and may not be available for every item on every platform.

However, the user data, the item data, and the behavior data are essential. With the three types of data ready, we can move on to the data processing and storage layer. In this layer, we can perform data processing, such as identifying user features, material features, and event features. Going forward is modelling based on these features.

As we have aforementioned in the preceding section, the entire recommendation process contains two important modules: matching and ranking. Multiple algorithms can run in parallel in the matching module.

Matching is followed by ranking. Many ranking algorithms are also available. We will go into details about which ranking algorithm to use in the third article of the series. Next, you need to develop a new policy. You must filter and deduplicate the recommendation results, perform A/B tests on the results, and try the operational strategies before you push the recommendations online. The top layer is the recommendation service, which can recommend an advertisement, a product, or a user.

For example, a social networking app can recommend users to let them follow each other. When you have such a recommendation architecture, some cloud services will be needed to make the architecture meet the four basic requirements on an enterprise-level recommender system. The most common practice is to build these modules based on cloud services and cloud ecosystems.

16. Code Implementation

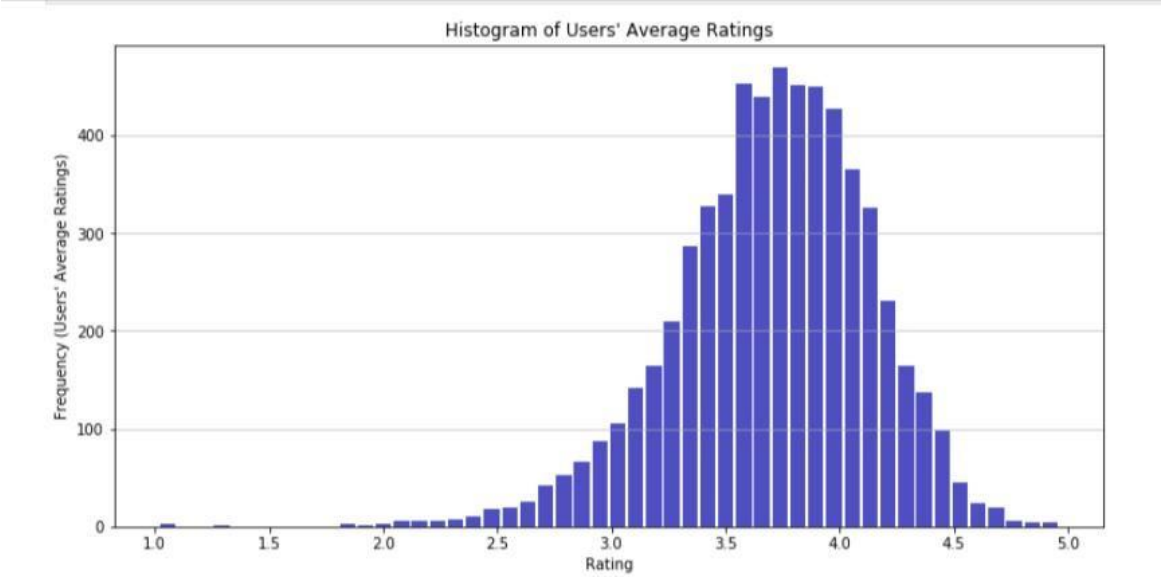
We used ‘MovieLens 1M Dataset’ [15] for our analysis. The dataset contains 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users who joined MovieLens in 2000. We used two files in particular, namely ratings and movies. The ratings file contained 4 fields. They are UserID, MovieID, Rating and Timestamp.

- UserIDs range between 1 and 6040
- MovieIDs range between 1 and 3952
- Ratings are made on a 5-star scale (whole-star ratings only)
- Timestamp is represented in seconds since the epoch
- Each user has at least 20 ratings.

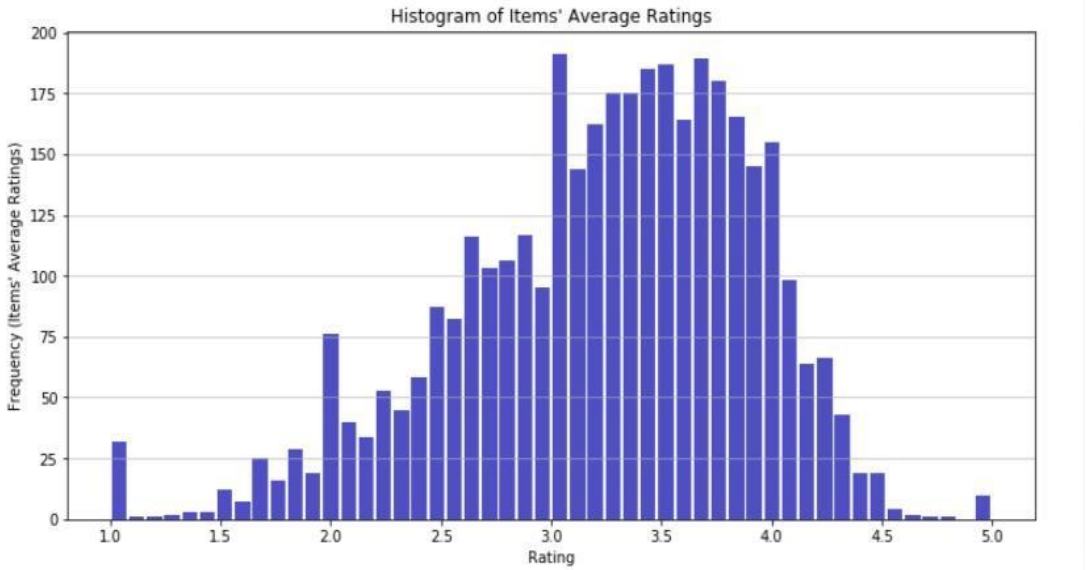
The movies file contained 3 fields. They are MovieID, Title and Genres

- Titles are identical to titles provided by the IMDB (including year of release)
- Genres are pipe-separated and are selected from the following genres: Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western We performed some initial exploratory analysis on the datasets.

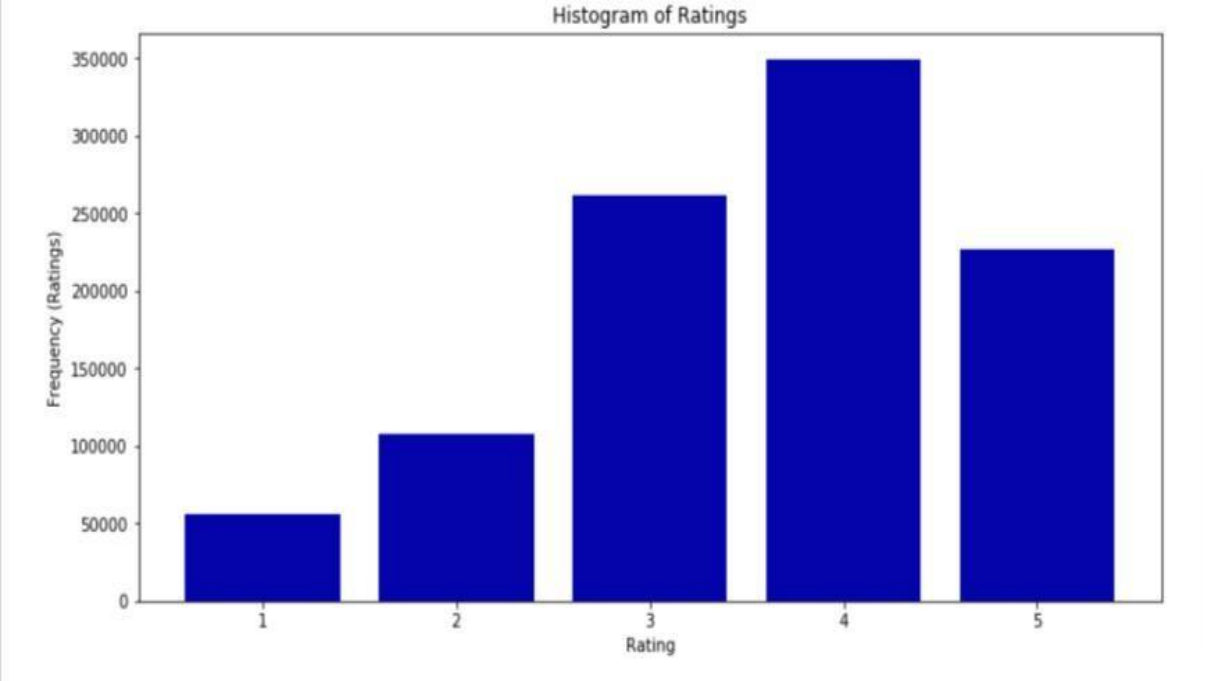
  Below Figure illustrates the histogram of average ratings given by the users. We can see this plot approximates a normal distribution with a left heavy tail. Most users' average ratings fall between 3.5 and 4.



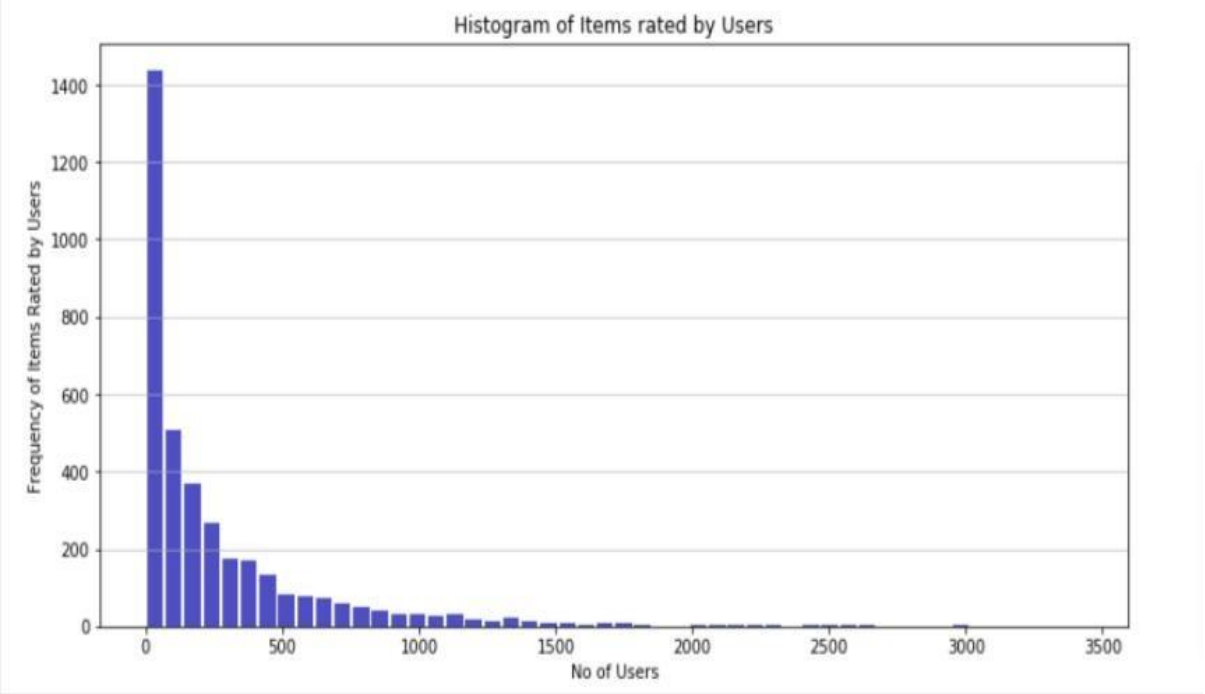
Below Figure illustrates the histogram of average ratings that items got. This plot also approximates a normal distribution with a left heavy tail. However, in this case, the values are more spread out. Most items have been rated between 3 to 4.

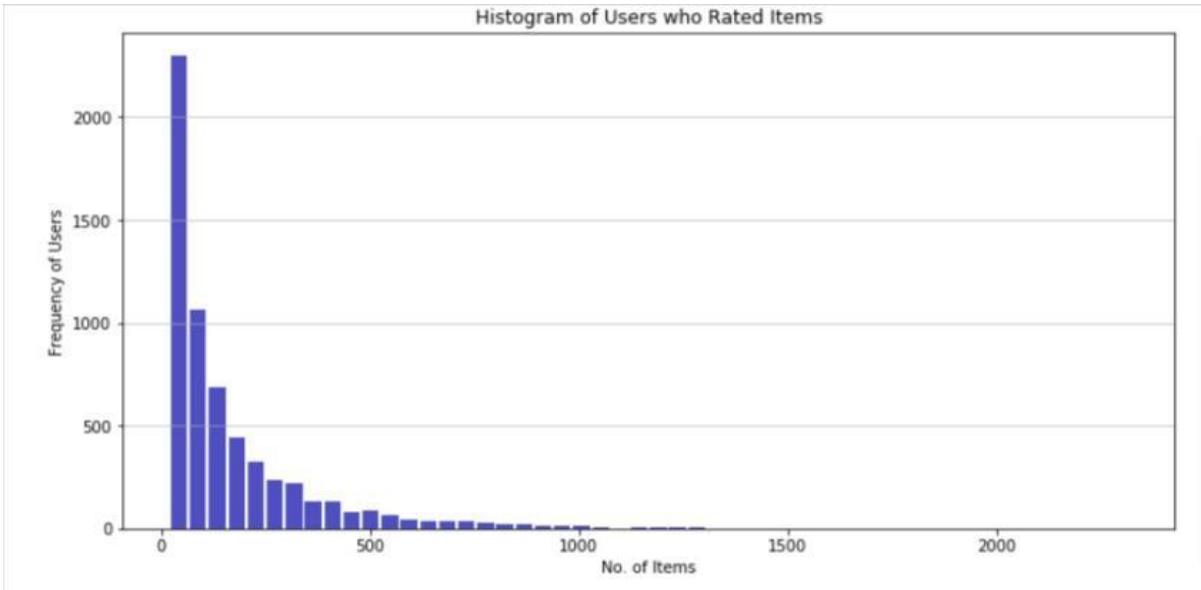


Below Figure shows the histogram of ratings. It is consistent with the previous two plots as we see that the most frequent ratings are 4 and 3 respectively.

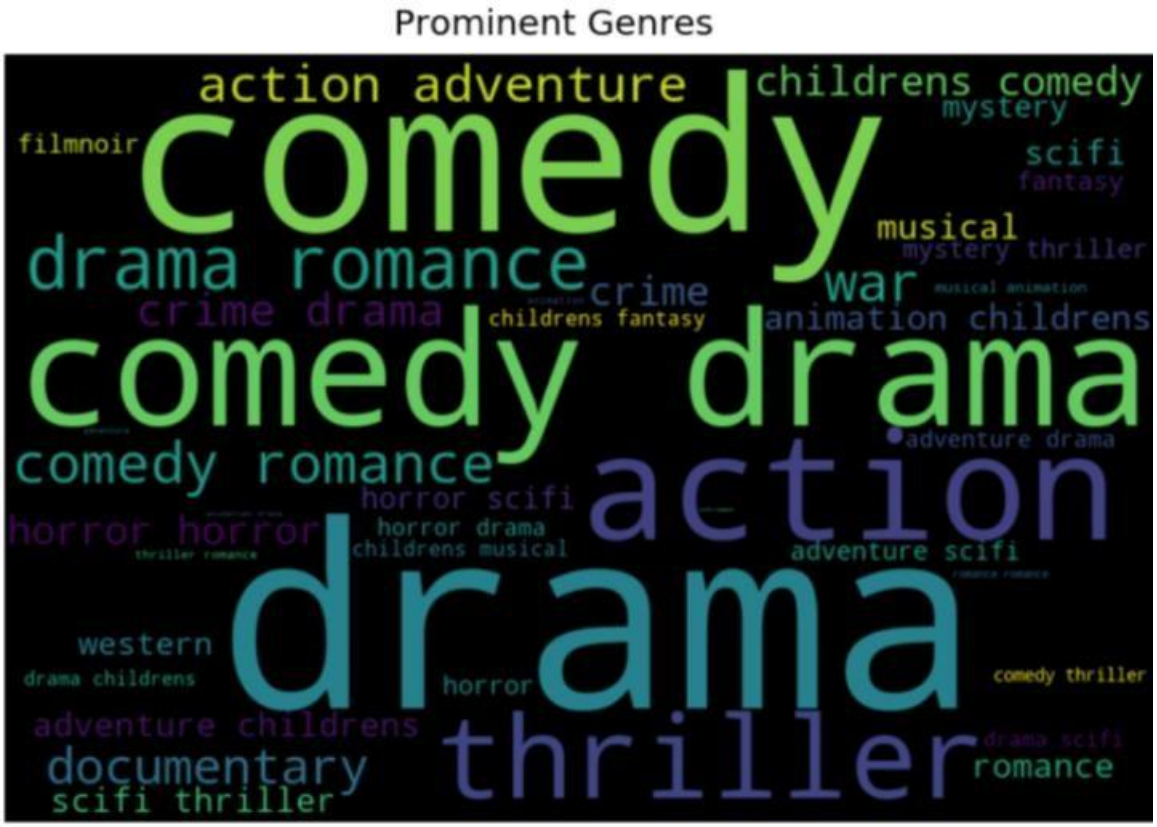


Above and Below Figure illustrates the histogram of items rated by users and users who rated items. As expected from these two plots, most users rate very few items.





We also generated a word cloud of the genres of the movies. A word cloud is a visual representation of text data, typically used to depict particular words where the importance of each word is shown with font size or color. This format is useful for quickly perceiving the most prominent terms to determine its relative prominence. Figure 13 shows some of the most popular genres. As we can see in the word cloud drama and comedy are the most common ones.



Result Analysis

➢ Quantitative Analysis

We first take a look at the comparison of RMSE and MAE errors between a Collaborative Filtering based and Hybrid system. Content-Based Filtering method has only a qualitative property and therefore we'll cover it in the next subsection.

Here we choose top-recommended movies by both systems for 10 users and calculate RMSE errors for each system for comparison. From the RMSE plot for 10 users in Figure 14, we see that the hybrid system has comparatively lower RMSE overall. The average RMSE plot in Figure 15 also shows the superiority of the hybrid system.

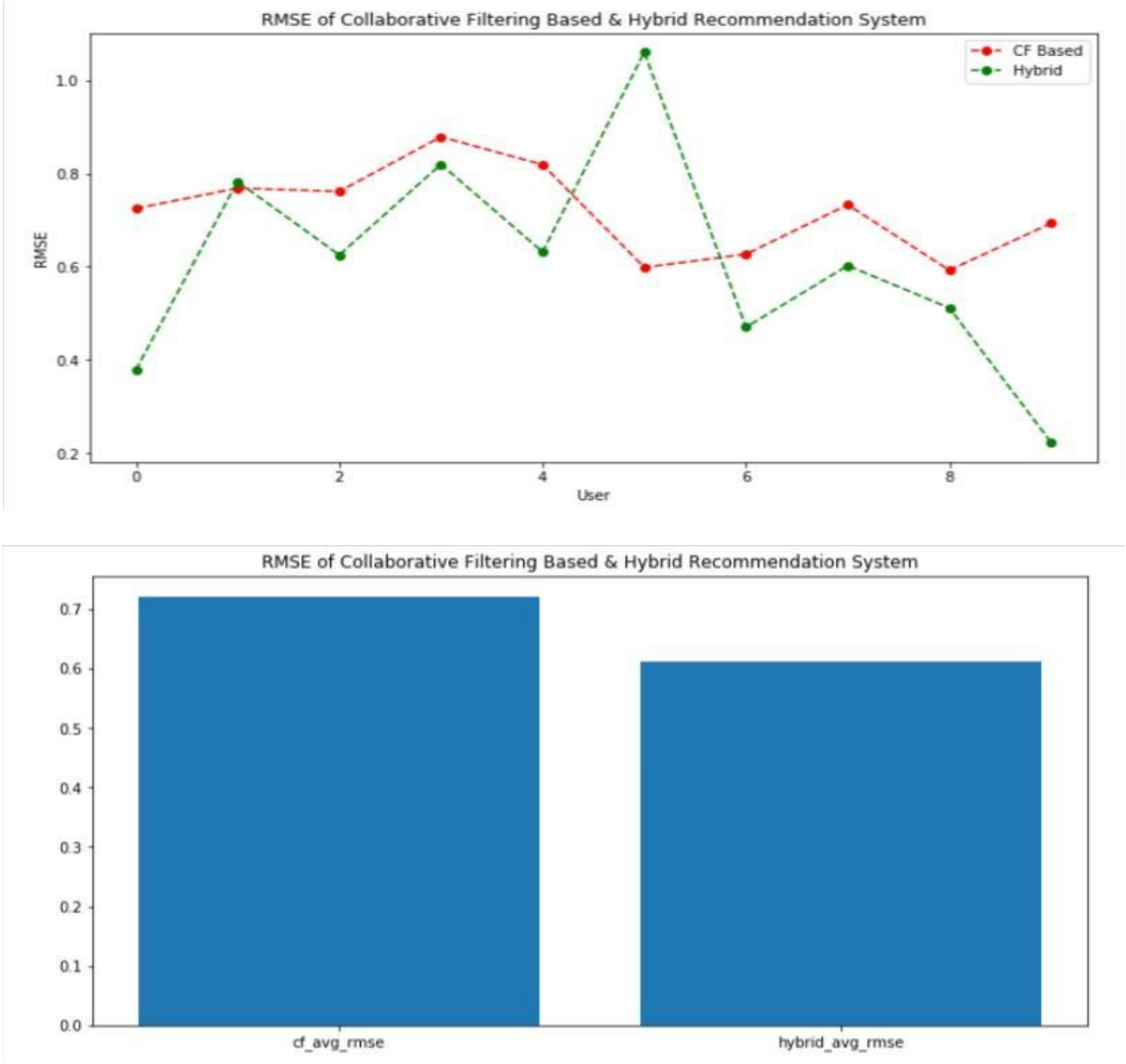


Figure- Average RMSE of Collaborative Filtering Based and Hybrid Recommendation System

We then do the same evaluation for MAE and from Figure 16 and Figure 17 see that the hybrid recommendation system has comparatively lower MAE, i.e., better accuracy.

Figure-MAE of Collaborative Filtering Based and Hybrid Recommendation System

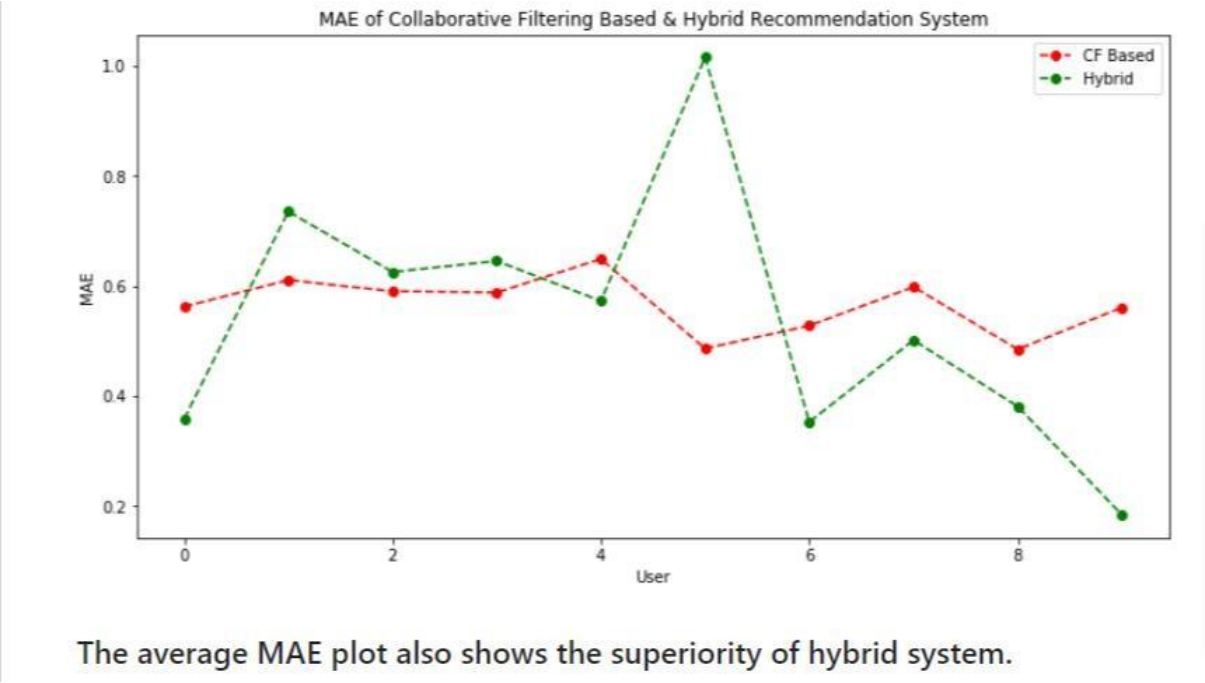
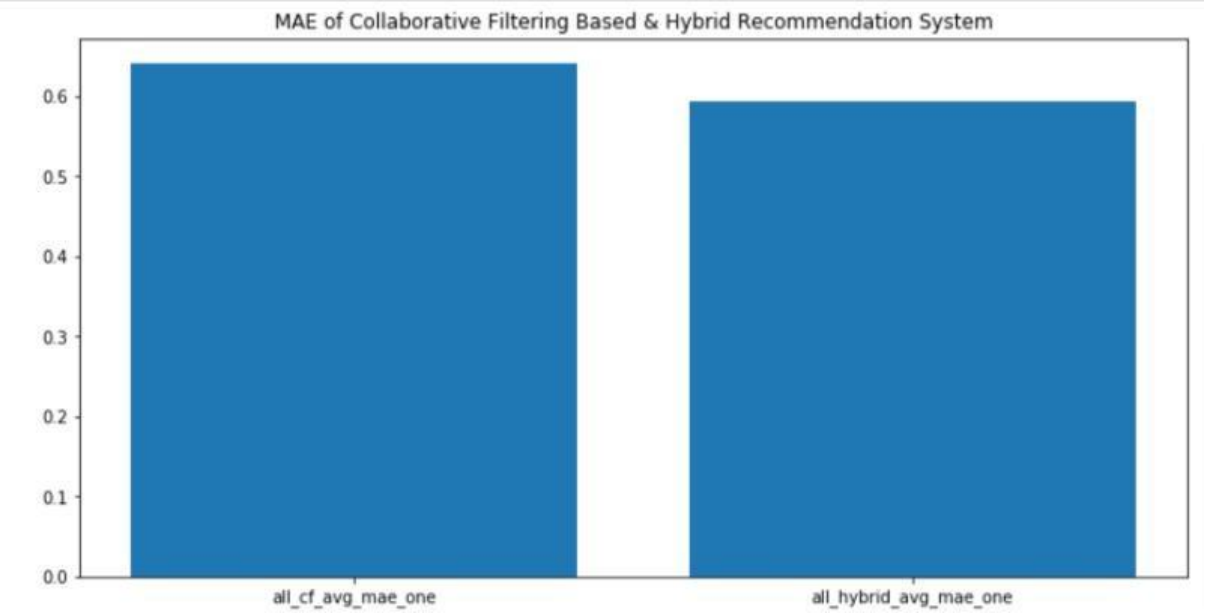


Figure -Average MAE of Collaborative Filtering Based and Hybrid Recommendation System



consider 5 batches of users with each batch containing 5 users for whom we do the same test. We calculated the RMSE of these sets of users and the comparison shows Hybrid system performs comparatively better. The plot in Figure 18 shows the RMSE values. Figure 19 shows the average RMSE of Collaborative Filtering and Hybrid Recommendation System. We did the same for MAE with 5 sets of user groups that is shown in Figure-C. Hybrid system came on top here too. Figure-D shows the average MAE of Collaborative Filtering and Hybrid Recommendation System.

Figure A- RMSE of Collaborative Filtering Based and Hybrid Recommendation System for 5 sets of users

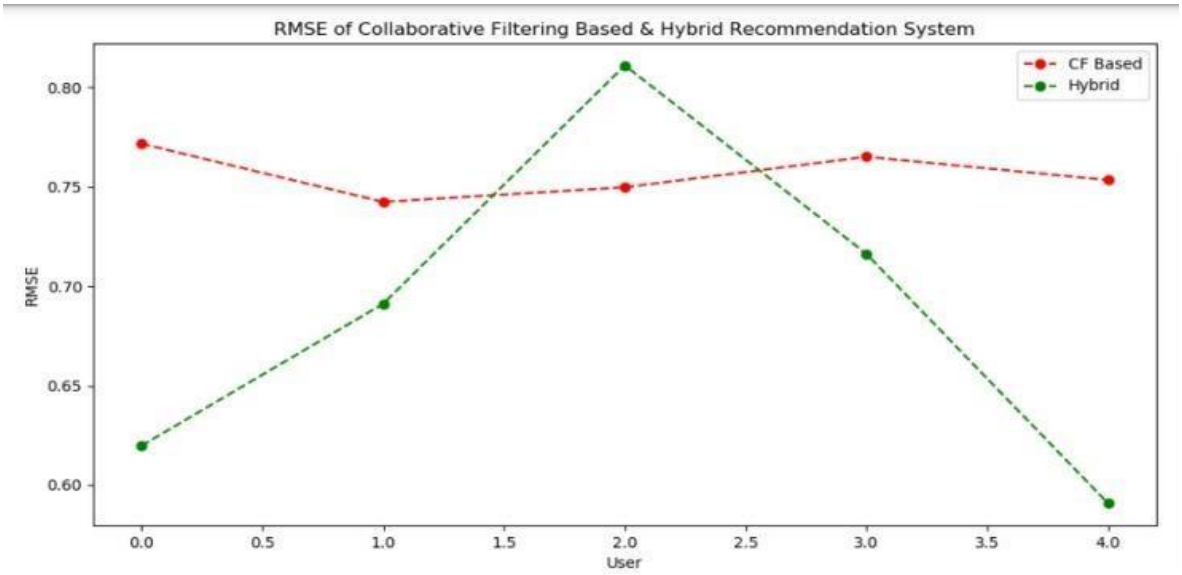
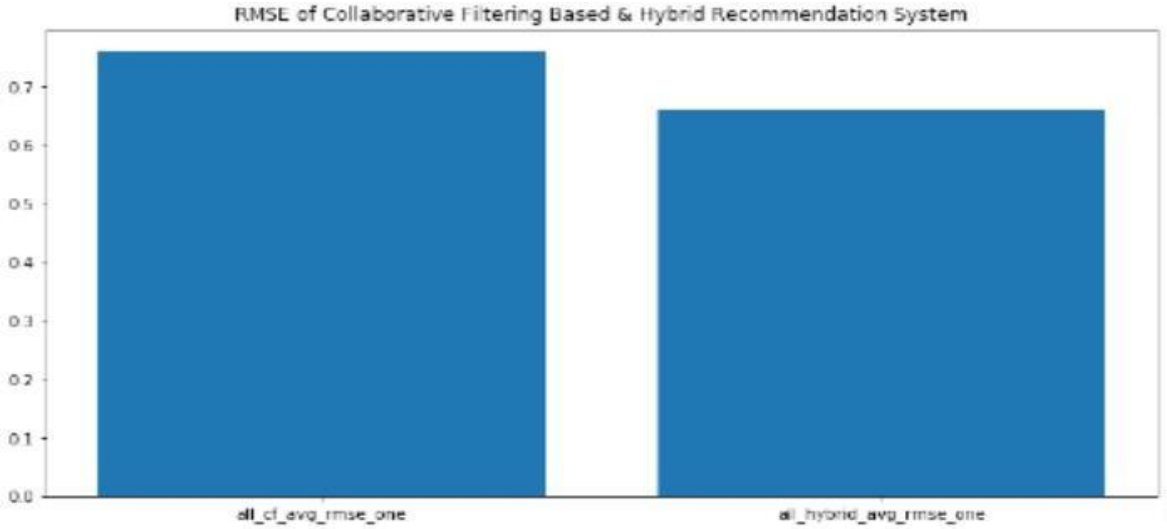


Figure B. Average RMSE of Collaborative Filtering Based and Hybrid Recommendation System for 5 sets of users



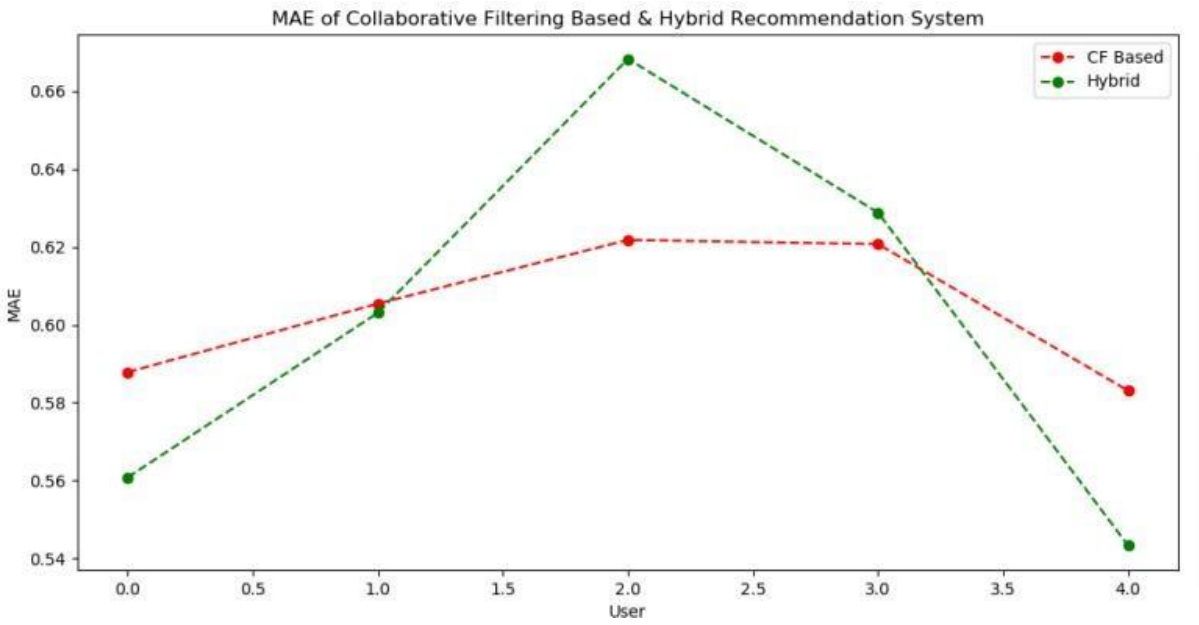


Figure C -MAE of Collaborative Filtering Based and Hybrid Recommendation System for 5 sets of users

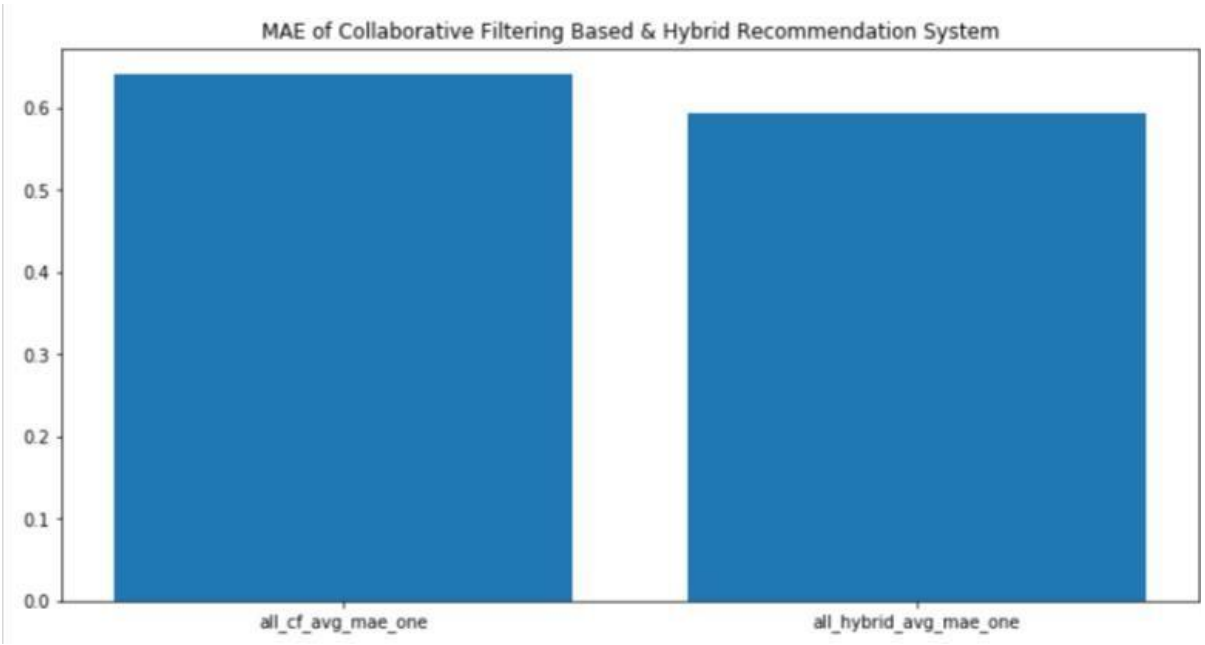


Figure 21. Average MAE of Collaborative Filtering Based and Hybrid Recommendation System for 5 sets of users

➢ Qualitative Analysis

From Table 2 we see that Collaborative Filtering can tell us the movies that a user is likely to rate higher. But it has no way of recommending similar movies to a particular one tailored for the specific user. As we can from the genre's column, the genres are all over the places. Here, we consider User 1 and recommend the top 20 movies he is likely to rate high. Table 2. Top 20 Recommended Movies for a Particular User by Collaborative Filtering Based Recommendation System

movie_id	estimated_rating	title	actual_rating	genres
527	4.995416	[Schindler's List (1993)]	[5]	[Drama|War]
318	4.958150	[Shawshank Redemption, The (1994)]	[]	[Drama]
1172	4.894460	[Cinema Paradiso (1988)]	[]	[Comedy|Drama|Romance]
2905	4.797475	[Sanjuro (1962)]	[]	[Action|Adventure]
1269	4.727696	[Arsenic and Old Lace (1944)]	[]	[Comedy|Mystery|Thriller]
920	4.695179	[Gone with the Wind (1939)]	[]	[Drama|Romance|War]
2019	4.693533	[Seven Samurai (The Magnificent Seven) (Shichi...	[]	[Action|Drama]
904	4.686336	[Rear Window (1954)]	[]	[Mystery|Thriller]
922	4.654332	[Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)]	[]	[Film-Noir]
1234	4.651489	[Sting, The (1973)]	[]	[Comedy|Crime]
1203	4.645870	[12 Angry Men (1957)]	[]	[Drama]
905	4.644219	[It Happened One Night (1934)]	[]	[Comedy]
858	4.639405	[Godfather, The (1972)]	[]	[Action|Crime|Drama]
1197	4.635691	[Princess Bride, The (1987)]	[3]	[Action|Adventure|Comedy|Romance]
1233	4.632157	[Boat, The (Das Boot) (1981)]	[]	[Action|Drama|War]
356	4.629706	[Forrest Gump (1994)]	[]	[Comedy|Romance|War]
1198	4.626178	[Raiders of the Lost Ark (1981)]	[]	[Action|Adventure]
953	4.613348	[It's a Wonderful Life (1946)]	[]	[Drama]
912	4.610552	[Casablanca (1942)]	[]	[Drama|Romance|War]
1242	4.608209	[Glory (1989)]	[]	[Action|Drama|War]

On the other hand, a Content-Based Filtering recommendation system has the option to find us the most similar movies to a given one as seen in Table 3, but it has no intuition into whether a user will like it or not. Here, we consider Movie Name: Toy Story 39 (1995) with Movie ID 1 and recommend the top 20 movies which are similar to the movie, Toy Story.

Table 3. Top 20 Recommended Movies for a Particular Movie by Content-Based Filtering Recommendation System

movie_index	similarity_score	title	movie_id	genres
1050	1.000000	Aladdin and the King of Thieves (1996)	1064	[['Animation', "Children's", 'Comedy']]
2072	1.000000	American Tail, An (1986)	2141	[['Animation', "Children's", 'Comedy']]
2073	1.000000	American Tail: Fievel Goes West, An (1991)	2142	[['Animation', "Children's", 'Comedy']]
2285	1.000000	Rugrats Movie, The (1998)	2354	[['Animation', "Children's", 'Comedy']]
2286	1.000000	Bug's Life, A (1998)	2355	[['Animation', "Children's", 'Comedy']]
3045	1.000000	Toy Story 2 (1999)	3114	[['Animation', "Children's", 'Comedy']]
3542	1.000000	Saludos Amigos (1943)	3611	[['Animation', "Children's", 'Comedy']]
3682	1.000000	Chicken Run (2000)	3751	[['Animation', "Children's", 'Comedy']]
3685	1.000000	Adventures of Rocky and Bullwinkle, The (2000)	3754	[['Animation', "Children's", 'Comedy']]
236	0.869805	Goofy Movie, A (1995)	239	[['Animation', "Children's", 'Comedy', 'Romanc...
12	0.826811	Balto (1995)	13	[['Animation', "Children's"]]
241	0.826811	Gumby: The Movie (1995)	244	[['Animation', "Children's"]]
310	0.826811	Swan Princess, The (1994)	313	[['Animation', "Children's"]]
592	0.826811	Pinocchio (1940)	596	[['Animation', "Children's"]]
612	0.826811	Aristocats, The (1970)	616	[['Animation', "Children's"]]
700	0.826811	Oliver & Company (1988)	709	[['Animation', "Children's"]]
876	0.826811	Land Before Time III: The Time of the Great Gi...	888	[['Animation', "Children's"]]
1010	0.826811	Winnie the Pooh and the Blustery Day (1968)	1023	[['Animation', "Children's"]]
1012	0.826811	Sword in the Stone, The (1963)	1025	[['Animation', "Children's"]]
1020	0.826811	Fox and the Hound, The (1981)	1033	[['Animation', "Children's"]]

A hybrid system gives us the best of both worlds. Table 4 shows that it can recommend similar movies to a particular one that the user is most likely to rate high. Here, we consider User ID 1, Movie Toy Story (1995) with Movie ID 1 and recommend top 20 movies which are similar to Toy Story and the movies which are likely to be rated high by the User 1.

Table 4. Top 20 Recommended Movies for a Particular User and Movie by Hybrid Recommendation System

movie_index	similarity_score	title	estimated_rating	actual_rating
2073	1.000000	American Tail: Fievel Goes West, An (1991)	4.107195	[]
1050	1.000000	Aladdin and the King of Thieves (1996)	4.077366	[]
2285	1.000000	Rugrats Movie, The (1998)	4.061477	[]
3685	1.000000	Adventures of Rocky and Bullwinkle, The (2000)	4.052081	[]
3542	1.000000	Saludos Amigos (1943)	3.732172	[]
3682	1.000000	Chicken Run (2000)	3.595624	[]
3045	1.000000	Toy Story 2 (1999)	3.319540	[]
2072	1.000000	American Tail, An (1986)	3.047335	[]
2286	1.000000	Bug's Life, A (1998)	2.749218	[]
236	0.869805	Goofy Movie, A (1995)	3.779034	[]
1949	0.826811	Bambi (1942)	4.424280	[]
2731	0.826811	Little Nemo: Adventures in Slumberland (1992)	4.384338	[]
2618	0.826811	Tarzan (1999)	4.315832	[]

Therefore, we can conclude that from both qualitative and quantitative perspective, a hybrid recommendation system performs comparatively better than standalone Collaborative Filtering or Content-Based Filtering recommendation system.

17 Final Product Prototypes

- **Feasibility**

This project can be developed and deployed within a few years as SaaS(Software as a Service) for anyone to use.

- **Viability**

Ecommerce websites and Social Media Platforms use personalization as an effective strategy by providing one to one services like product recommendation, information and ratings of the product by satisfying individual users' needs.
Many online stores and E-commerce websites are discovering ways of using Personalized Recommendation system to increase user interaction with the services they provide. There are many e-commerce websites and Online stores in the market, they are not able to reach their product to the customer. They do not know what customer actually wants, what type of products he wanted to buy.
So, it is viable to survive in the long-term future as well but improvements are necessary as new technologies emerge.

- **Monetization**

This service is directly monetizable as it can be directly released as a service on completion which can be used by businesses.

Step1- Business Modeling

The classic marketplace model:

The most popular revenue model for modern marketplaces is to charge a commission from each transaction. When a customer pays a provider, the platform facilitates the payment and charges either a percentage or a flat fee.
One of the ever growing business models that continues to prove highly effective is becoming a marketplace. This means we are simply bringing supply and demand together. AirBNB reigns as one of the top success stories to implement this business model well.Uber has also seen explosive growth using the same mentality to create a marketplace where strangers rent rides from strangers. Providing a service is out, and becoming the marketplace is in the ever growing e-commerce sector.

Why It Works: There are several advantages to using this type of business model. First, one of the greatest benefits is having zero to little overhead, and no inventory. You can get a swanky office space if you want, or you can run the company virtually. When you manufacture a product, you take on a lot more risk and pressure to make sure that inventory is sold. When you are the marketplace, instead of worrying about manufacturing costs, you are simply bringing the sellers to the buyers (and vice versa) and facilitating a transaction, taking a small slice of the pie from each transaction. You give sellers a place to make a profit and reach consumers, while customers are happy to find exactly what they want, usually at a discounted price.

Others Who Have Followed: Amazon is one of the leaders of this business model, creating a marketplace for those who wish to sell items, and those who wish to buy them at a better price. Raise is a a C2C gift card market, that a supply of discounted gift cards from sellers who would rather have the cash to spend as they please. Beast is another example of a marketplace that connects high level consultants for the millennial era with clients looking to outsource unmet needs in their business.

Step2-Financial Modelling:

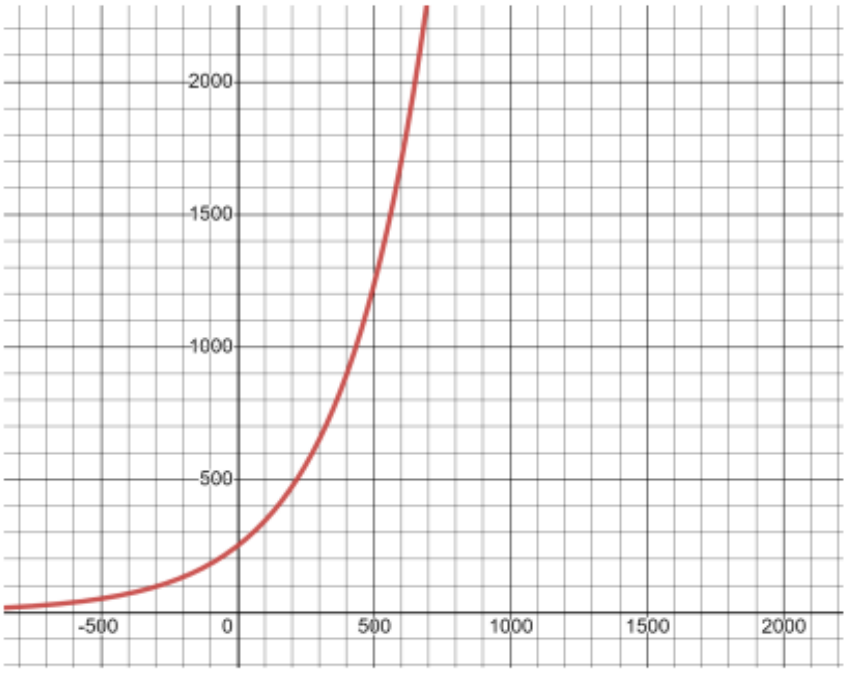
It can be directly launched into the retail market.
Let's consider our price of product = 250 for getting our graph

Financial Equation:

$$\mathbf{Y = X * (1 + r)^t}$$

$$\mathbf{Y = X * (1.035)^t}$$

Y = Profit over time, X = Price of our Product, r = growth rate, t = time interval
1+r = 1 + 3.5% = 1.035



18.Conclusions

Personalized Recommended System in E-commerce, Online Websites and Social platforms help to get customized recommendations for every user by analyzing users data, their purchases, rating and their relationships with other users in more detail.

Many online stores and E-commerce websites are discovering ways of using Personalized Recommendation system to increase user interaction with the services they provide.

There are many e-commerce websites and Online stores in the market, they are not able toreach their product to the customer. They do not know what customer actually wants, whattype of products he wanna buy to give them good profits. So, we will target the customer andshow him some recommended products by using Personalized recommendation system by predicting whether a particular user would prefer an item or not based on the user's profile.Recommender systems are beneficial to both service providers and users. They reducetransaction costs of finding and selecting items in an online shopping environment.

I have hence proposed the application of this technique for E-Commerce and Online stores. This is not a full-fledged plan, but with a considerable amount of work and effort, it seems achievable.