RECOMMENDATION SYSTEM

Submitted by

Harshitha G [RA2011030010020]
Rakshita Raj [RA2011030010030]
Devshree Moghe [RA2011030010049]

Under the Guidance of

Dr. V. M. GAYATHRI

Associate Professor, Department of Networking and Communications

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY COMPUTER SCIENCE AND ENGINEERING

in

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

MAY 2023

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Course Project Report titled **"Recommendation System"** is the bonafide work done by "**Harshitha G. (RA2011030010020), Rakshita Raj (RA2011030010030) and Devshree Moghe(RA2011030010049)**", who carried out under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Faculty In-Charge
Dr. V. M. GAYATHRI
Assistant Professor
Department of Networking & Communications
SRM Institute of Science and Technology

SIGNATURE

HEAD OF THE DEPARTMENT
Dr. K. Annapurani Panaiyappan
Professor and Head,
Department of Networking & Communications
SRM Institute of Science and Technology

ABSTRACT

Recommendation systems are an essential tool for businesses and users to make personalized and relevant decisions. These systems use machine learning and artificial intelligence algorithms to analyze user data and provide recommendations that match their preferences and needs. The success of a recommendation system project depends on various factors, including the accuracy, effectiveness, and coverage of the recommendations, as well as user satisfaction, business impact, scalability, and robustness of the system.

The recommendation system project aims to develop an effective recommendation system that can provide personalized and relevant recommendations to users based on their preferences and behavior. The project uses a collaborative filtering algorithm to analyze user data and provide recommendations for items that the user has not yet interacted with. The system is built using Python and leverages the Pandas and Scikit-learn libraries for data manipulation and machine learning, respectively.

Table of Content

List of Figures

List of Abbreviations

NLP	Natural Language Processing
NLTK	Natural Language Toolkit
AI	Artificial Intelligence

INTRODUCTION:

A recommendation system, also known as a recommender system, is a type of algorithm that makes personalized recommendations to users based on their past behaviors, preferences, and interests. These systems are commonly used in e-commerce, online advertising, streaming services, social networks, and other types of online platforms where users are presented with a large number of options and need help in finding relevant content.

Recommendation systems can use a variety of data sources to make recommendations, including user behavior data, content data, and social data. Collaborative filtering, content-based filtering, and hybrid approaches are common techniques used in recommendation systems.

Collaborative filtering analyzes the behavior of a group of users and makes recommendations based on their collective preferences. For example, if a group of users who have similar tastes have all rated a particular movie highly, the system may recommend that movie to other users with similar preferences.

Content-based filtering, on the other hand, analyzes the characteristics of the content being recommended and suggests similar content to users based on their past interactions with similar content. For example, if a user has shown a preference for action movies, the system may recommend other action movies to them.

Hybrid recommendation systems combine both collaborative filtering and content-based filtering techniques to provide more accurate and diverse recommendations.

Overall, recommendation systems are valuable tools for businesses and users alike. By providing personalized recommendations, they can improve user engagement, increase sales, and enhance the overall user experience.

.

Problem Statement:

Recommendation systems have become increasingly important in recent years as the volume of data available on the internet has grown rapidly. These systems provide personalized recommendations to users based on their past behaviour, preferences, and other relevant data, and can help to improve user engagement, satisfaction, and loyalty.

However, if many users buy the same product but do not leave any feedback or ratings, it becomes difficult for the recommendation system to recommend that particular product. When a new item is introduced on a website that is already using a recommendation system, it will not recommend it until and unless a user buys it. This shelves down the possibility of a good product coming into the limelight.

Every recommendation has it’s pros and cons. But what’s important is that most of its challenges can be tackled by thinking wisely while investing in a recommendation system. We can use aile to avoid such challenges.

OBJECTIVE:

The primary objective of a recommendation system is to provide personalized and relevant recommendations to users based on their preferences, behavior, and other relevant data. However, there are several other objectives that recommendation systems can help to achieve, including:

1. Increased engagement: By providing personalized and relevant recommendations, recommendation systems can help to increase user engagement with a platform or service.

2. Improved user satisfaction: By providing recommendations that meet the needs and preferences of users, recommendation systems can help to improve user satisfaction and loyalty.

3. Increased revenue: Recommendation systems can help to drive revenue by promoting products or services that users are more likely to be interested in. By increasing the relevance of recommendations, recommendation systems can help to increase user engagement, leading to higher conversion rates and increased revenue.

4. Reduced information overload: Recommendation systems can help to reduce information overload by presenting users with a curated selection of items or content that are relevant to their interests and preferences.

5. Improved discovery: Recommendation systems can help users discover new products or services that they may not have otherwise found on their own, leading to increased engagement and revenue.

Scope and Applications:

Scope:

The scope of recommendation systems is very broad, as they can be applied in various industries and settings to provide personalized and relevant recommendations to users. It extend to online retailers who use recommendation systems to suggest products to customers based on their past purchases, browsing history, and other relevant data.

Recommendation systems are used in media and entertainment to suggest movies, TV shows, music, and other content to users based on their past viewing or listening behavior.

They are used in healthcare to suggest treatment options to patients based on their medical history and other relevant data.

Applications:

Recommendation systems are widely used in various industries and settings to provide personalized and relevant recommendations to users. Here are some examples of the applications of recommendation systems:

a) 3. Social media: Social media platforms use recommendation systems to suggest new connections or content to users based on their interests and preferences.

b) 4. Travel: Travel and hospitality companies use recommendation systems to suggest hotels, flights, and other travel-related products or services to users based on their past travel behavior.

c) Education: Recommendation systems can be used in education to suggest courses, learning materials, and other resources to students based on their interests and learning history.

d) Job search: Recommendation systems can be used in job search platforms to suggest jobs to candidates based on their skills, experience, and other relevant factors.

e) Financial services: Recommendation systems can be used in financial services to suggest investment options or financial products to customers based on their investment goals and risk tolerance.

f) Advertising: Recommendation systems can be used to personalize advertising to individual users based on their preferences and behavior.

g) Food and beverage: Restaurants and food delivery services can use recommendation systems to suggest menu items or meals based on customers' dietary preferences and past orders.

General and Unique: How is AI helping?:

Artificial intelligence (AI) is playing a crucial role in improving the performance and accuracy of recommendation systems. In the context of our recommendation system project, AI algorithms to analyze images and videos to understand user preferences and recommend relevant content. Deep learning techniques, such as neural networks, are used in recommendation systems to analyze large amounts of data and learn patterns that can be used to make personalized recommendations. Natural language processing (NLP) is used to understand and analyze text data, such as user reviews, and extract valuable information that can be used to improve recommendations.

AI algorithms can take into account contextual information, such as user location, time of day, and weather, to provide more relevant recommendations.

Reinforcement learning is a technique in which the recommendation system learns from user feedback to improve its recommendations over time.

Overall, AI is helping in recommendation systems by enabling them to process and analyze large amounts of data, learn from user behaviour and feedback, and provide more accurate and relevant recommendations. As AI continues to evolve, it is likely that recommendation systems will become even more personalized and effective in the future.

Software required specifications:

To perform the sentiment analysis project using Python, the following software requirements are needed:

1. Python 3.6 or higher: Python is an interpreted, high-level, general-purpose programming language that is widely used for data analysis and machine learning. It is required to run the Python scripts that will be used in this project.
2. Jupyter Notebook or any Python IDE: Jupyter Notebook is a web-based interactive computing environment that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It is a popular choice for data analysis and machine learning projects. Alternatively, any Python IDE (Integrated Development Environment) such as PyCharm, Spyder, or Visual Studio Code can also be used to write and execute Python scripts.
3. Required libraries: The following Python libraries are required for this project:

- pandas: A library for data manipulation and analysis that provides a dataframe structure for working with tabular data.
- numpy: A library for numerical computing that provides support for large, multidimensional arrays and matrices.
- matplotlib: A library for data visualization that provides a variety of charts and plots.
- seaborn: A library for data visualization that builds on top of matplotlib and provides additional features for creating more complex visualizations. •

4. Data: A dataset of product reviews is required for this project. The dataset should be in a CSV format and should contain various columns such as the product ID, user ID, score, summary, and text of the review.

By installing the required software and libraries and obtaining the necessary data, the recommendation system project can be completed successfully.

LITERATURE

Comparison of proposed system over existing system

The working of an existing recommendation system depends on the specific technique used. However, in general, recommendation systems work by analyzing user behaviour and preferences, as well as item attributes, to make personalized recommendations.

The recommendation system collects data on user behavior and preferences, as well as item attributes, such as genre or price.

The recommendation system uses a recommendation algorithm, such as collaborative filtering or content-based filtering, to analyze the user and item profiles and make personalized recommendations.

The recommendation system is refined and updated based on user feedback and new data to improve its performance.

Our project aims to improve upon the existing recommendation by using Aile,an AIpowered Recommendation Engine for any online business (website and mobile apps).It uses machine learning algorithms to analyze user data and recommend personalized content to user. The results are improved user engagement, higher conversion rates, increased revenue, and loyal customers.

Architecture :

The architecture of a recommendation system typically involves several components that work together to provide personalized recommendations. Here is a general overview of the architecture of a recommendation system:

1. Data storage: The recommendation system requires a database to store user and item data, such as user profiles and item attributes.

2. Data processing: The data processing component is responsible for cleaning, transforming, and preprocessing the data to prepare it for analysis.

3. Recommendation engine: The recommendation engine is the core component of the recommendation system and is responsible for analyzing user and item data to generate personalized recommendations. The recommendation engine can use various techniques such as collaborative filtering, content-based filtering, matrix factorization, and deep learning to generate recommendations.

4. User interface: The user interface component is responsible for presenting the recommendations to the user in a user-friendly and intuitive manner. The user interface can be a web application, mobile app, or other user-facing interface.

5. Feedback mechanism: The feedback mechanism component is responsible for collecting user feedback on the recommendations and using it to improve the recommendations over time.

6. Analytics and monitoring: The analytics and monitoring component is responsible for tracking the performance of the recommendation system and providing insights into user behavior and preferences.

Overall, the architecture of a recommendation system involves several components that work together to provide personalized recommendations. The specific architecture can vary depending on the application and the available data.

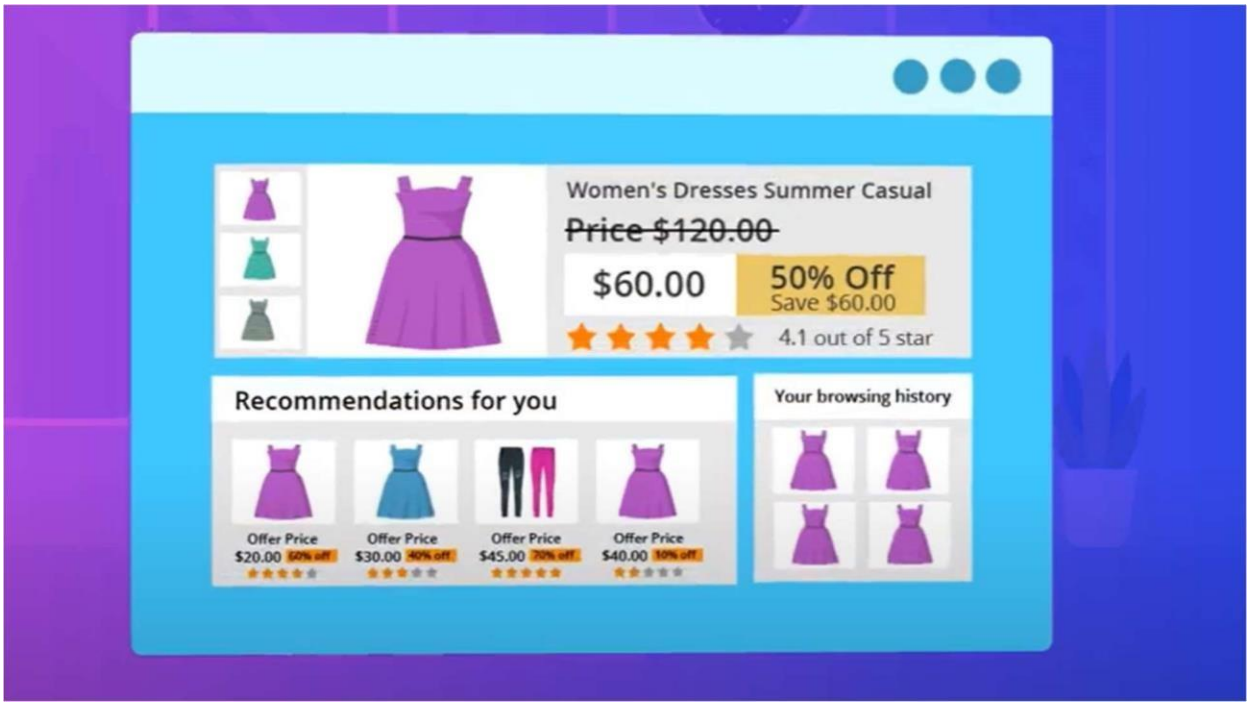
Aile :

With Collaborative filtering, Alie makes recommendations based on your user's behavior. Alie utilizes behavioral patterns such as user preferences, activities, and interactions to find similar or matching users, and proceeds to predict what the user will like based on their similarities with other users. The algorithm can be customized to find similarities based on items, users, or both. With the Nearest Neighborhood Algorithm model, Alie generates a rating system based on the nearest neighbor in your database and recommends the most likely matc

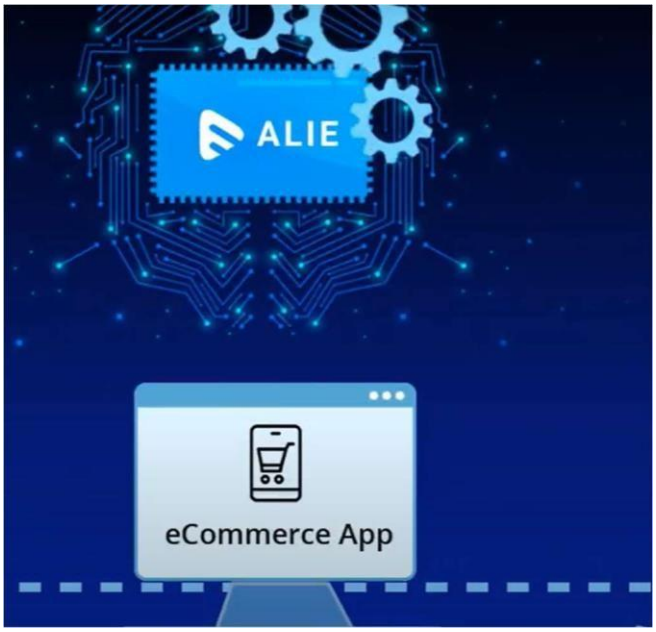
Alie's "Review Recommendations" feature allows you to analyze and review the final output (recommendations) in readable JSON format even before you integrate it on your website or app. It allows you to explore recommendations, see if they are in sync with the input data and algorithm you selected. Alie offers "add-on filters" to remove any unwanted item from final recommendations. For Ex. You can remove recommendations by adding a filter if you do not want to show items that are rated less than 3 out of 5 stars

Reviewing final recommendations is very simple, just log into Alie CMS, create a new project, add data using APIs or JS Plugins, select the algorithm, and click on View Output (Recommendations). Subsequently, you can hook these recommendations on the appropriate pages of your website or app using APIs provided by Alie.

Alie ingests item data, user data & behavioral data from your website or app to provide real-time recommendations. Implementing Alie for your website/app is fairly easy and can be achieved in four simple steps.


Women's Dresses Summer Casual
Price $120.00
$60.00
50% Off
Save $60.00
4.1 out of 5 star
Recommendations for you
Offer Price
$20.00 60% off
Offer Price
$30.00 40% off
Offer Price
$45.00 20% off
Offer Price
$40.00 10% off
Your browsing history


ALIE


ALIE
eCommerce App


ALIE


ALIE
ALIE
Video Streaming Website


ALIE
ALIE
News App
Social Media App

There are three ways with which Alie can collect data. You can use APIs, CSVs or RSS feeds to provide item and user data. You can use Webhooks & JS Plugin or use Cookies for sending user activity or behavioural data.
User and product data can be uploaded with simple CSV files, RSS feeds or APIs. You can also push all meta-information associated with your item & user data. More meta information you provide better it is for system.

Behavioural data is gathered either explicitly - reviews, ratings, likes, etc. or implicitly - gathered from available data streams, such as search history, clicks, order history, and other activities.

Choose an appropriate algorithm from a drop-down menu on Alie CMS. Alie analyzes the user, item, and behavioural data based on a chosen algorithm to generate the recommendations.

APIs return a list of Recommended Product(Item) IDs for a user based on the model that is trained.
You can then use these Product(Item) IDs to hook recommendations on your website or app to create a unique personalized experience. Recommendations can be shown on home pages, product pages or shopping carts, etc.

Code and Output:

AI-PROJECT.py - C:\Users\HP\Desktop\AI-PROJECT.py (3.9.7)

File Edit Format Run Options Window Help

```
#import the reqired libraries
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"
import numpy as np
import pandas as pd
import math
import json
import time
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.model_selection import train_test_split
from sklearn.neighbors import NearestNeighbors
import joblib
import scipy.sparse
from scipy.sparse import csr_matrix
from scipy.sparse.linalg import svds
import warnings; warnings.simplefilter('ignore')
from IPython.display import display
#%matplotlib inline

# Import the dataset and give the column names
columns=['userId', 'productId', 'ratings','timestamp']
electronics_df=pd.read_csv('ratings_Electronics.csv',names=columns)
display(electronics_df.head())
electronics_df.drop('timestamp',axis=1,inplace=True)
electronics_df.info()

rows,columns=electronics_df.shape
print('Number of rows: ',rows)
print('Number of columns: ',columns)

electronics_df1=electronics_df.iloc[:50000,0:]
```

AI-PROJECT.py - C:\Users\HP\Desktop\AI-PROJECT.py (3.9.7)

File Edit Format Run Options Window Help

```
electronics_df1['ratings'].describe().transpose()

print('Minimum rating is: %d' %(electronics_df1.ratings.min()))
print('Maximum rating is: %d' %(electronics_df1.ratings.max()))

most_rated=electronics_df1.groupby('userId').size().sort_values(ascending=False)[:10]

print('Top 10 users based on ratings: \n',most_rated)

counts=electronics_df1.userId.value_counts()
electronics_df1_final=electronics_df1[electronics_df1.userId.isin(counts[counts>=15].index)]
print('Number of users who have rated 25 or more items =', len(electronics_df1_final))
print('Number of unique users in the final data = ', electronics_df1_final['userId'].nunique())
print('Number of unique products in the final data = ', electronics_df1_final['userId'].nunique())

final_ratings_matrix = electronics_df1_final.pivot(index = 'userId', columns ='productId', values = 'ratings').fillna(0)
display(final_ratings_matrix.head())

train_data, test_data = train_test_split(electronics_df1_final, test_size = 0.3, random_state=0)
display(train_data.head())


train_data_grouped = train_data.groupby('productId').agg({'userId': 'count'}).reset_index()
train_data_grouped.rename(columns = {'userId': 'score'},inplace=True)
display(train_data_grouped.head(40))


train_data_sort = train_data_grouped.sort_values(['score', 'productId'], ascending = [0,1])

#Generate a recommendation rank based upon score
train_data_sort['rank'] = train_data_sort['score'].rank(ascending=0, method='first')

#Get the top 5 recommendations
```

```
#Get the top 5 recommendations
popularity_recommendations = train_data_sort.head(5)
display(popularity_recommendations)



def recommend(user_id):
    user_recommendations = popularity_recommendations

    #Add user_id column for which the recommendations are being generated
    user_recommendations['userId'] = user_id

    #Bring user_id column to the front
    cols = user_recommendations.columns.tolist()
    cols = cols[-1:] + cols[:-1]
    user_recommendations = user_recommendations[cols]

    return user_recommendations

find_recom = [10,100,150]   # This list is user choice.
for i in find_recom:
    print("The list of recommendations for the userId: %d\n" %(i))
    print(recommend(i))
    print("\n")
```

The dataset for the recommendation system:

```
           userId   productId  ratings   timestamp
0   AKM1MP6P0OYPR  0132793040      5.0  1365811200
1  A2CX7LUOHB2NDG  0321732944      5.0  1341100800
2  A2NWSAGRHCP8N5  0439886341      1.0  1367193600
3  A2WNBOD3WNDNKT  0439886341      3.0  1374451200
4  A1GI0U4ZRJA8WN  0439886341      1.0  1334707200
```

Maximum and minimum ratings given:

```
memory usage: 1.1+ MB
Minimum rating is: 1
Maximum rating is: 5
```

Getting top 10 users based on ratings:

```
Top 10 users based on ratings:
 userId
A231WM2Z2JL0U3    37
AY8Q1X7G96HV5     31
ALUNVOQRXOZIA     20
A1NVD0TKNS1GT5    19
A243HY69GIAHFI    18
A1RPTVW5VEOSI     17
A1ISUNUWG0K02V    16
A1MJMYLRTZ76ZX    16
A7Y6AVS576M03     15
A3MEIR72XKQY88    15
```

Giving scores to products based users:

	productId	score
0	1400599997	1
1	B00000DM9M	1
2	B00000J061	1
3	B00000J08C	1
4	B00000J1QZ	1
5	B00000J3HB	1
6	B00000J3II	2
7	B00000J3Q7	1
8	B00000J3T1	1
9	B00000J47A	1
10	B00000J4ER	1
11	B00000J4FS	1
12	B00000J4O2	1
13	B00000J570	2
14	B00000J579	1
15	B00000JBIA	1
16	B00000JBJQ	2
17	B00000JBK6	1
18	B00000JBUI	1
19	B00000JDEI	1
20	B00000JFDW	1
21	B00000JFJA	1
22	B00000JFMW	1
23	B00000JHVP	1

Giving ranks to products:

	productId	score	rank
87	B00004RC2D	4	1.0
57	B00002SWHH	3	2.0
99	B00004SC3Y	3	3.0
100	B00004SCKA	3	4.0
6	B00000J3II	2	5.0

Recommending lists to particular users:

```
The list of recommendations for the userId: 10

     userId   productId  score  rank
87       10  B00004RC2D      4   1.0
57       10  B00002SWHH      3   2.0
99       10  B00004SC3Y      3   3.0
100      10  B00004SCKA      3   4.0
6        10  B00000J3II      2   5.0


The list of recommendations for the userId: 100

     userId   productId  score  rank
87      100  B00004RC2D      4   1.0
57      100  B00002SWHH      3   2.0
99      100  B00004SC3Y      3   3.0
100     100  B00004SCKA      3   4.0
6       100  B00000J3II      2   5.0


The list of recommendations for the userId: 150

     userId   productId  score  rank
87      150  B00004RC2D      4   1.0
57      150  B00002SWHH      3   2.0
99      150  B00004SC3Y      3   3.0
```

Result :

The result of the project shows that after applying the AI algorithms, the users were recommended the top products based on the ratings of other users. Here, it shows that the recommendation system used the collaborative filtering algorithm look for the products with the best ratings given by other users and recommend to the new users who have just started speculating the products.

For the new users, the recommendation system can face a challenge for recommending products if they were not given ratings or if there is a new trend in the market, it would also face difficulty in recommending. For this, we can use the Aile recommendation system in which, we will feed the algorithm and dataset that we want and it will work same as any other recommendation system. The advantage of using Aile is that it will help in making the recommendation more flexible, which in turn will allow the us to scale the recommendation system based on the latest trends in the market, to add new filters and parameters etc.

Conclusion and Future enhancements:

In conclusion, recommendation systems have become an essential tool for businesses and users to make personalized and relevant decisions. These systems use machine learning and artificial intelligence algorithms to analyze user data and provide recommendations that match their preferences and needs. Recommendation systems face various challenges, including data sparsity, scalability, and privacy concerns. However, recent advances in machine learning and artificial intelligence have enabled the development of more sophisticated and effective recommendation systems that can handle these challenges.

Future Enhancements:

The future of recommendation systems looks promising, and there are several potential enhancements that can improve the accuracy, efficiency, and personalization of these systems. Some of the future enhancements of recommendation systems include:

a) Context-aware recommendations: Context-aware recommendation systems take into account the user's current context, such as location, time, and social network, to provide more relevant and personalized recommendations.

b) Explainable recommendations: Explainable recommendation systems provide users with explanations for why specific recommendations were made, which can help users understand and trust the system's recommendations.

c) Hybrid recommendation systems: Hybrid recommendation systems combine different recommendation algorithms, such as collaborative filtering and content-based filtering, to provide more accurate and diverse recommendations.

d) Personalized pricing: Personalized pricing uses recommendation systems to determine the optimal prices for products or services based on the user's preferences and behavior.

e) Integration with voice assistants: Integration with voice assistants, such as Amazon Alexa and Google Assistant, can make recommendation systems more accessible and convenient for users.

f) Our project uses Aile to make the recommendation system more flexible and up-to-date with the market trends.

References:

https://www.muvi.com/blogs/challenges-of-using-a-recommendation-system

https://www.linkedin.com/company/alie-ai/about/

https://www.muvi.com/alie/how-itworks