M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur, Tamilnadu.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE

EXPERIENTAL LEARNING: 1
WEB & SOCIAL MEDIA MINING

SUBMITTED BY	HARIPRIYA I	927621BAD013
	JOTHIKA R	927621BAD019
	KEERTHIKA S	927621BAD-24
SUBMITTED TO	Dr. T. SARAVANAN (AP/AI)	
SUBMISSION DATE	MARCH 4 , 2024	

MOVIE RECOMMENDATION DATA MINING

INTRODUCTION:

Movie recommendation systems play a significant role in enhancing user experience by providing personalized suggestions based on their preferences. In this report, we delve into the domain of movie recommendation data mining, using Python and the MovieLens dataset. By employing collaborative filtering techniques and cosine similarity, we aim to generate personalized movie recommendations for a specific user.

ABOUT THE DATASET:

The dataset used in this analysis comprises two main files - 'movies.csv' and 'ratings.csv' - sourced from the MovieLens dataset. The 'movies.csv' file contains information about various movies, while the 'ratings.csv' file provides user ratings for these movies. Merging these datasets allows us to create a comprehensive movie ratings matrix.

UNDERSTANDING THE MOVIE DATA:

The movie recommendation data is structured as a user-item matrix, where users' ratings for different movies are organized. Utilizing collaborative filtering, we calculate the cosine similarity between movies to identify those with similar user ratings. This similarity matrix enables the system to recommend movies based on the preferences of the target user.

SELECTING MINING TECHNIQUE:

The key mining technique employed in this analysis is collaborative filtering, specifically using cosine similarity. This method allows us to identify movies that are closely related based on user ratings. Other potential techniques for movie recommendation systems include content-based filtering, matrix factorization, and hybrid approaches combining multiple methods.

IMPLEMENTATION:

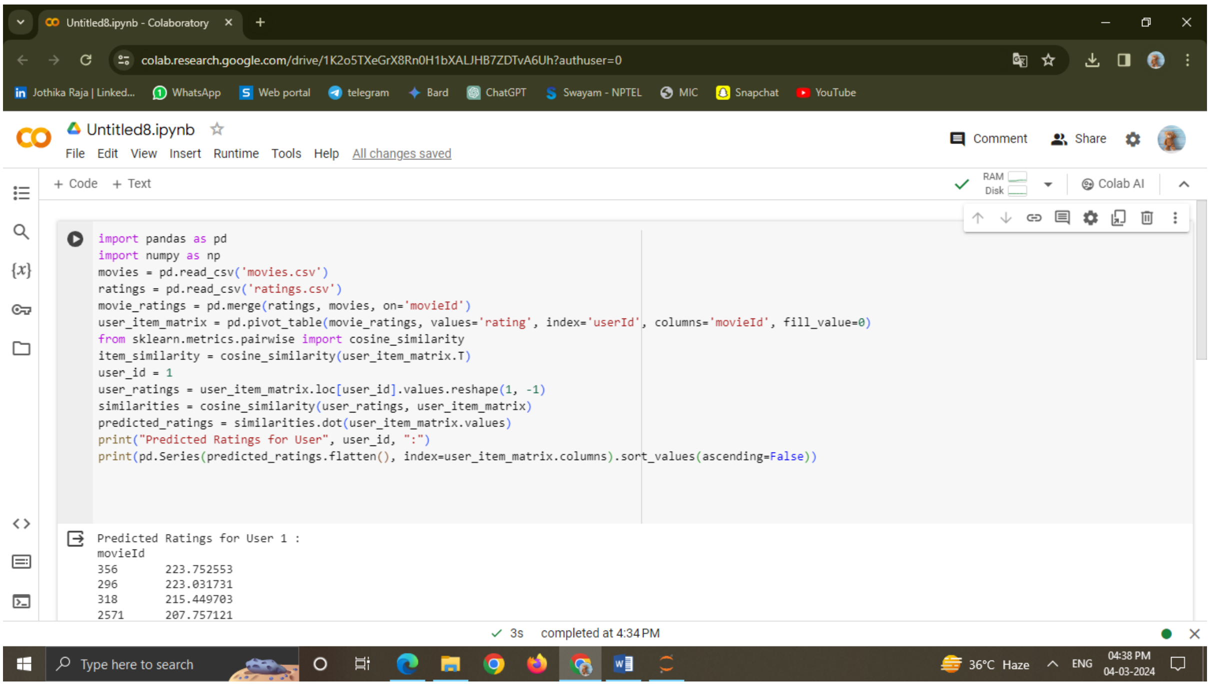
Python serves as the primary programming language for implementing movie recommendation mining techniques. The Pandas library facilitates data loading, preprocessing, and manipulation, while the scikit-learn library provides tools for calculating cosine similarity. The core implementation involves the following steps:

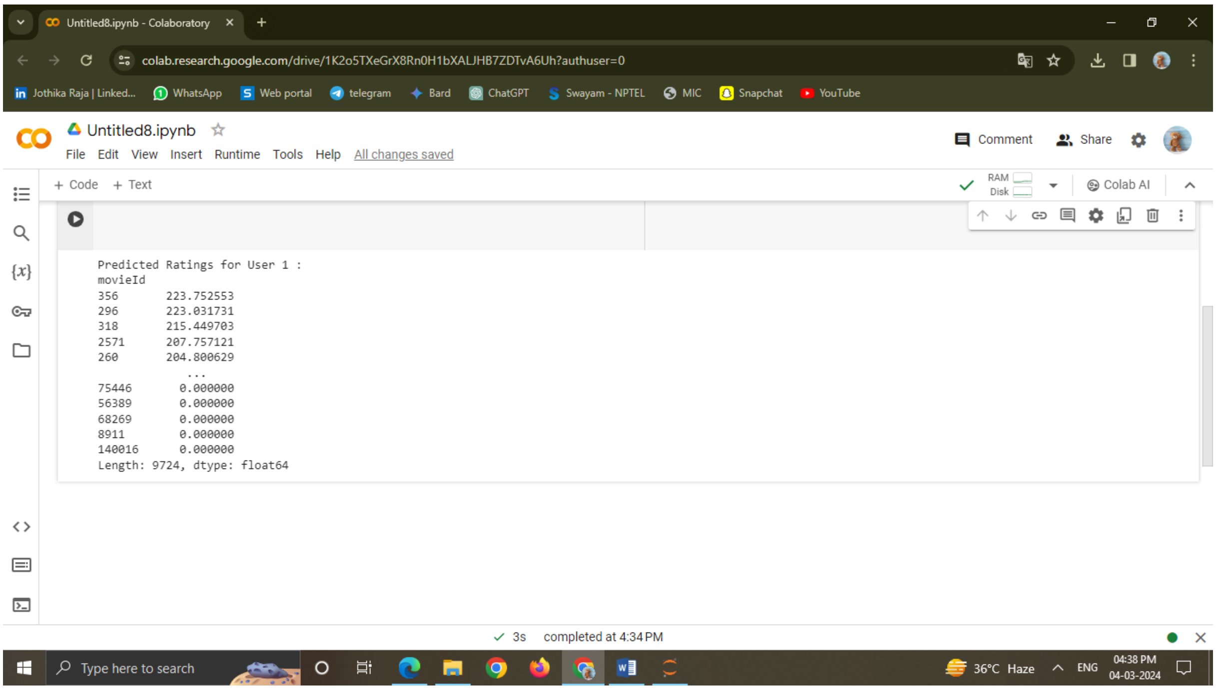
i)Loading and Preprocessing Data: Utilizing Pandas to read 'movies.csv' and 'ratings.csv,' merging the datasets, and creating a user-item matrix.

ii)Calculating Similarity Matrix: Employing cosine similarity to calculate the similarity between movies, forming the basis for recommendations.

iii)Generating Movie Recommendations for a User: Taking a specific user's ratings, calculating the similarity with all movies, and predicting their ratings for unrated movies.

This implementation allows for the generation of personalized movie recommendations based on a user's historical ratings and the similarity between movies in the dataset. By analyzing user-item interactions, the system aims to enhance user satisfaction and engagement with movie content.


Untitled8.ipynb - Colaboratory
colab.research.google.com/drive/1K2o5TXeGrX8Rn0H1bXALJHB7ZDTvA6Uh?authuser=0
Untitled8.ipynb
File Edit View Insert Runtime Tools Help All changes saved
import pandas as pd
import numpy as np
movies = pd.read_csv('movies.csv')
ratings = pd.read_csv('ratings.csv')
movie_ratings = pd.merge(ratings, movies, on='movieId')
user_item_matrix = pd.pivot_table(movie_ratings, values='rating', index='userId', columns='movieId', fill_value=0)
from sklearn.metrics.pairwise import cosine_similarity
item_similarity = cosine_similarity(user_item_matrix.T)
user_id = 1
user_ratings = user_item_matrix.loc[user_id].values.reshape(1, -1)
similarities = cosine_similarity(user_ratings, user_item_matrix)
predicted_ratings = similarities.dot(user_item_matrix.values)
print("Predicted Ratings for User", user_id, ":")
print(pd.Series(predicted_ratings.flatten(), index=user_item_matrix.columns).sort_values(ascending=False))
Predicted Ratings for User 1 :
movieId
356 223.752553
296 223.031731
318 215.449703
2571 207.757121
3s completed at 4:34 PM
04:38 PM 04-03-2024


Untitled8.ipynb - Colaboratory
colab.research.google.com/drive/1K2o5TXeGrX8Rn0H1bXALJHB7ZDTvA6Uh?authuser=0
Untitled8.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Predicted Ratings for User 1 :
movieId
356 223.752553
296 223.031731
318 215.449703
2571 207.757121
260 204.800629
...
75446 0.000000
56389 0.000000
68269 0.000000
8911 0.000000
140016 0.000000
Length: 9724, dtype: float64
3s completed at 4:34 PM
04:38 PM 04-03-2024

CONCLUSION:

In this exploration of movie recommendation data mining using collaborative filtering and cosine similarity, we leveraged the MovieLens dataset to generate personalized movie recommendations for a specific user. The journey involved several key steps, beginning with the loading and preprocessing of movie data through Pandas. The creation of a user-item rating matrix enabled a comprehensive understanding of user interactions with various movies. The core of our approach relied on calculating the similarity matrix using cosine similarity. This matrix allowed us to identify relationships between movies based on user ratings, facilitating the generation of personalized recommendations. The final step involved predicting ratings for the specified user by applying the calculated similarities to the entire movie dataset.