A Project Report on

Movie Recommendation System

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Computer Science

by

Zulfiqar Ali Mirza (18102043)
Jeel Jain(18102041)
Yashi Rathod(18102039)
Vandan Jain (18102031)

Under the Guidance of

Prof.Rahul Ambekar

Department of Computer Science
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2020-2021

Approval Sheet

This Project Report entitled ***Movie Recommendation System"*** Submitted by ***"Zulfiqar Ali Mirza "(18102043), "Jeel Jain"(18102041), "Yashi Rathod"(18102039), "Vanda Jain"(18102031)***is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Computer Science*** from ***University of Mumbai***.

(Prof.Rahul Ambekar)
Guide

Prof. Sachin Malave
Head Department of Computer Science

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***“Movie Recommendation System”*** submitted by ***“Zulfiqar Ali Mirza ” (18102043),“Jeel Jain” (18102041),“Yashi Rathod” (18102039),“Vandan Jain” (18102031)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Science***,to the University of Mumbai,is a bonafide work carried out during academic year 2020-2021.

(Prof.Rahul Ambekar)
Guide

Prof. Sachin Malave
Head Department of Computer Science

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **Movie Recommendation System.** We take this opportunity to express our sincere thanks towards our guide **Guide Prof.Rahul Ambekar** Department of CS, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin Malave** Head of Department,CS, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol Kalugade** BE project co-ordinator, Department of CS, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1:Zulfiqar Ali Mirza
Student ID1:18102043

Student Name2:Jeel Jain
Student ID2:18102041

Student Name3:Yashi Rathod
Student ID3:18102039

Student Name4:Vandan Jain
Student ID4:18102031

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Zulfiqar Ali Mirza 18102043)
(Jeel Jain 18102041)
(Yashi Rathod 18102039)
(Vandan Jain 18102031)

Date:

Abstract

Movie Recommender System recommends movies similar to the movie user likes and analyses the sentiments on the reviews given by the user for that movie. The details of the movies(title, genre, runtime, rating, poster, etc.) are fetched using an API by TMDB,, and using the IMDB id of the movie in the API Recommendation systems collect customer data and auto analyze this data to generate customized recommendations for your customers. These systems rely on both implicit data such as browsing history and purchases and explicit data such as ratings provided by the user.Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. Although, a set of movie recommendation systems have been proposed, most of these either cannot recommend a movie to the existing users efficiently or to a new user by any means.

Contents

Chapter 1

Introduction

Recommender systems (RS) are the software machinery that recommends or suggest appropriate items to users. Three important stages are involved in the tasks of RS, object data collection, similarity decision and prediction computation. Also, Recommendation systems are based on three major approaches. Content-based approaches makes use of the content of the items and even the users. In our methodology, we have used the Genres and Tags. Thus using this method one can find the similarities between the content of one movie and the other which are liked by the users. To predict the likes of a target user, Collaborative Filtering takes into consideration, the neighbors of that target user, and finds the similarity between the neighbors and target user such that the most similar users are selected and their ratings and likes are recommended to the target users [3,5,8]. Thus, user preferences are dependent on the others users present in the active user's neighborhood. Also, the domain dependence nature of CF can make it vulnerable to sparsity and cold start. Further, this class of recommender can be classified into memory based ,model based and the hybrid of the two . Since Collaborative filtering is largely dependent on the ratings of the users, therefore, if the number of users in the domain is low as compared to items then it can lead to cold start Hybrid approaches are the amalgam of content based and collaborative filtering [4,6,7,10,12]. In the later sections, we will discuss that how a hybrid CF algorithm is superior to CF and CB. II

Chapter 2

Project Concept

2.1 Abstract

Movie Recommendation System recommends movies similar to the movie user likes and analyses the sentiments on the reviews given by the user for that movie. The details of the movies(title,genre, runtime, rating, poster, etc.) are fetched using an API by TMDB,, and using the IMDB id of the movie in the API Recommendation systems collect customer data and auto analyze this data to generate customized recommendations for your customers. These systems rely on both implicit data such as browsing history and purchases and explicit data such as ratings provided by the user.Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. Although, a set of movie recommendation systems have been proposed, most of these either cannot recommend a movie to the existingusers efficiently or to a new user by any means.

2.2 Objectives

The main objective of this project is to make a Movie Recommendation System which generally provides movie recommendations on the basis of

1. Content Based Filtering:- Selects items based on the correlation between the content of the items and the user's preferences Compare user profile to content of each item

2. Collaborative Filtering:- chooses items based on the correlation between people with similar preferences. Rate items based on ratings of the users that rated the same items.

Hence, We develop a recommendation system on both CB and CF techniques in python which generally takes inputs from the user and recommends top 10 movies as recommendation to the user or client. And flask helps us to make an interactive user interface to finish our project.

2.3 Literature Review

Paper [1]:-
In this paper, we first introduce CF tasks and their main challenges, such as data sparsity,

scalability, synonymy, gray sheep, shilling attacks, privacy protection, etc., and their possible solutions. We then present three main categories of CF techniques: memory-based, model-based, and hybrid CF algorithms (that combine CF with other recommendation techniques), with examples for representative algorithms of each category, and analysis of their predictive performance and their ability to address the challenges. From basic techniques to the state-of-the-art, we attempt to present a comprehensive survey for CF techniques, which can be served as a roadmap for research and practice in this area.

Paper [2]: -
In this paper we describe our approach to collaborative filtering for generating personalized recommendations for users of Google News. We generate recommendations using three approaches: collaborative filtering using MinHash clustering, Probabilistic Latent Semantic Indexing (PLSI), and covisitation counts. We combine recommendations from different algorithms using a linear model. Our approach is content agnostic and consequently domain independent, making it easily adaptable for other applications and languages with minimal effort. This paper will describe our algorithms and system setup in detail, and report results of running the recommendations engine on Google News

Paper[3]:-
This paper discuss about recommendations of the movies. A movie recommendation is important in our social life due to its strength in providing enhanced entertainment. Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. A recommendation system is used for the purpose of suggesting items to purchase or to see. They direct users towards those items which can meet their needs through cutting down large database of Information. A recommender system, or a recommendation system (sometimes replacing 'system' with a synonym such as platform or engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item[1][2]. They are primarily used in commercial applications. MOVREC also help users to find the movies of their choices based on the movie experience of other users in efficient and effective manner without wasting much time in useless browsing.

Paper[4]:-
In this paper, we have used a hybrid methodology which takes advantage of both Content and Collaborative filtering algorithm into account. The algorithm discussed in this article is different from the previous work in this field as it includes a novel method to find the similar content between two items. The paper incorporates an analysis that justifies this new methodology and how it can provide practical recommendations. The above approach is tested on existing user and objects data and produced improved results when compared with other two favourite methods, Pure Collaborative Filtering, and Singular Value Decomposition.

2.4 Problem Definition

Recommendation systems are utilized in a variety of areas and are most commonly recognized as playlist generators for video and music services like Netflix, YouTube and Spotify, product recommenders for services such as Amazon, or content recommenders for social media platforms such as Facebook and Twitter. These systems can operate using a single input, like music, or multiple inputs within and across platforms like news, books, and search queries. There are also popular recommender systems for specific topics like restaurants and online dating. Recommender systems have also been developed to explore research articles and experts, collaborators, and financial services.

2.5 Scope

Recommendation systems help E-commerce sites to increase their sales. A movie recommendation system named MOVREC, based on collaborative filtering approach makes use of the information provided by users, analyzes them and then recommends the movie that is best suited to the user at that time using k-means algorithm

2.6 Technology Stack

Jupyter Notebook
HTML, CSS
JavaScript
Python
Flask or Django
Pandas
Scikit-learn
NLP (Natural Language Processing)
Tensorflow

2.7 Benefits for Environment And Society

Increased sales/conversion There are very few ways to achieve increased sales without increased marketing effort. Once you setup an automated recommendation system, you get recurring additional sales without any effort.
Increased user satisfaction Shortest path to a sale is great both for you and your customer reducing their effort. Recommendation systems allow you to reduce your customers' path to a sale by recommending them an appropriate option sometimes even before they search for it.
Reduced churn Recommendation system powered emails are one of the best ways to re-engage customers. Discounts or coupons are other effective yet costly ways of re-engaging customers and they can be coupled with recommendations to increase customer's probability of conversion.

Chapter 3

Project Design

3.1 Proposed System

Recommender system is a very hot research topic in recent years. Many researchers raised a lot of different recommendation approaches. The most famous category of these approaches is: • Content-based Recommendation.

- Collaborative-filtering Recommendation.
- Hybrid Recommendation

3.2 Design(Flow of modules)

•Perform Exploratory Data Analysis (EDA) on the data

•Gathering the appropriate data for the model

•Build the Movie Recommender System

•Testing the model on different parameters

•Get recommendations for the movies

•Build a web site for deployment of the model and integrate the same using flask /Django so that it can interact with the model

•Testing and improvement

3.3 Modules

3.3.1 Module 1-

Perform Exploratory Data Analysis (EDA) on the data

•The dataset contains two CSV files, credits, and movies. The credits file contains all the metadata information about the movie and the movie file contains the information like name and id of the movie, budget, languages in the movie that has been released, etc.

3.3.2 Module 2-

Build the Movie Recommender System

•The accuracy of predictions made by the recommendation system can be personalized using the "plot/description" of the movie.

•But the quality of suggestions can be further improved using the metadata of movie. Let's say the query to our movie recommendation engine is "The Dark Knight Rises". Then the predictions should also include movies directed by the director of the film. It should also include movies with the cast of the given query movie.

•For that, we utilize the following features to personalize the recommendation: cast, crew, keywords, genres.

•The movie data is present in the form of lists containing strings, we need to convert the data into a safe and usable structure.

3.3.3 Module 3-

Get recommendations for the movies

•The get recommendations() function takes the title of the movie and the similarity function as input. It follows the below steps to make recommendations.

•Get the index of the movie using the title.

•Get the list of similarity scores of the movies concerning all the movies.

•Enumerate them (create tuples) with the first element being the index and the second element is the cosine similarity score.

•Sort the list of tuples in descending order based on the similarity score.

•Get the list of the indices of the top 10 movies from the above sorted list. Exclude the first element because it is the title itself.

•Map those indices to their respective titles and return the movies list.

•Create a function that takes in the movie title and the cosine similarity score as input and outputs the top 10 movies similar to it.

3.4 References

[1] A Survey of Collaborative Filtering Techniques; Su et al; https://www.hindawi.com/journals/aai/200
[2] Google News Personalization: Scalable Online Collaborative Filtering; Das et al; https://www2007.or
[3] file:///C:/Users/Admin/Downloads/movrec[4] file:///C:/Users/Admin/Downloads/Movierecomm

Chapter 4

Planning for next semester

Plans For Next Semester,

1) Coding for Deployment Phase

a) Web app to deploy the model

b) Back-end program to Integrate Web App (mostly using Flask or Django) and Model. So that, the web app can interact with the model2) Testing and Analysis

2) And, Cloud Computing If Time Permits. That will be :

a) Getting the dataset from the Cloud Storage Blob

b) Using the Machine learning Api for our model

c) Deployment on any of the Cloud Platform (GCP, Azure or AWS)

2) And, Cloud Computing If Time Permits. That will be : a) Getting the dataset from the Cloud Storage Blob

b)Using the Machine learning Api for our model

c) Deployment on any of the Cloud Platform (GCP, Azure or AWS)