A Minor Project Report

on

Recommendation System using Machine Learning Techniques

Submitted to Manipal University Jaipur

towards the partial fulfillment for the award of the degree of

Bachelor of Technology

In Information Technology

By

Aditya Rana

209302339

Karthikey Sharma

209302122

IT 6 B

Under the Guidance of

Mr. Dhananjay Kumar

MANIPAL UNIVERSITY JAIPUR

INSPIRED BY LIFE

Department of Information Technology

School of Information Technology

Manipal University Jaipur 2022-2023

1. Introduction

1.1. Relevance of the project

A recommendation system or recommendation engine is a model used for information filtering where it tries to predict the preferences of a user and provide suggestions based on these preferences. These systems have become increasingly popular nowadays and are widely used today in areas such as movies, music, books, videos, clothing, restaurants, food, places, and other utilities. These systems collect information about a user's preferences and behavior and then use this information to improve their suggestions in the future. Movies are a part of life. Distinct types of movies exist, some for entertainment, education, children, horror, and some for action. There are a variety of movie genres, such as comedy, thriller, animation, action, etc. Another way to distinguish between movies can be by releasing year, language, director, etc. Watching movies online, there are several movies to search for in our most liked movies. Movie Recommendation Systems help us search our preferred movies among these diverse types of movies and reduce the trouble of spending a lot of time searching for our favorable movies. So, it requires that the movie recommendation system should be exceptionally reliable and should provide us with the recommendation of movies that are the same or most match our preferences. The movie Recommendation system is immensely powerful and important. Many companies are using recommendation systems to increase user interaction and enrich a user's shopping experience. Recommendation systems have several benefits, the most important being customer satisfaction and revenue.

Movie recommendation systems use a set of different filtration strategies and algorithms to help users find the most relevant films. The most popular categories of ML algorithms used for movie recommendations include content-based filtering and collaborative filtering systems.

There are majorly three types of recommendation systems:

- Content-based recommendation
- Collaborative filtering-based recommendation
- Hybrid recommendation

Content-based recommendation

A filtration strategy for movie recommendation systems, which uses the data provided about the items (movies). This data plays a crucial role here and is extracted from only one user. An ML algorithm used for this strategy recommends motion pictures that are like the user's preferences in the past. Therefore, the similarity in content-based filtering is generated by the data about past film selections and likes by only one user.

Collaborative-based recommendation

As the name suggests, this filtering strategy is based on the combination of the relevant user's and other users' behaviors. The system compares these behaviors for the most optimal results. It is a collaboration of multiple users' film preferences and behaviors.

Collaborative filtering algorithms are divided into two categories:

User-based collaborative filtering: The idea is to look for similar patterns in movie in preferences in the target user and other users in the database.

Item-based collaborative filtering: The basic concept here is to look for similar items (movies) that target users' rate or interact with.

Hybrid recommendation

Hybrid Recommender System is increasingly popular currently. Combining collaborative filtering and content-based filtering can be more effective in recent research.

2. Motivation

Many times, users face the problem of getting the right content to watch according to their current mood and choice of genre. It takes a lot of time to find the right content which usually leads to irritation and not wanting to watch anymore. The scope of this project is to provide accurate movie recommendations to users. The goal of the project is to improve the accuracy, quality, and scalability of the movie recommendation system compared to pure approaches. This is done using the Hybrid approach by combining content-based filtering and collaborative filtering. To eradicate the overload of data, a recommendation system is used as an information filtering tool in social networking sites. As a result, there is much room to explore this area to improve the scalability, accuracy, and quality of movie recommendation systems. Movie recommendation systems are especially important and powerful. But, due to the problems associated with a pure collaborative approach, movie recommendation systems also suffer from poor recommendation quality and scalability issues.

For building a recommender system from scratch, we face several different problems. Currently, there are a lot of recommender systems based on the content of information, so what should we do if the website does not have enough information about the movie? After that, we will solve the representation of a movie, which is how a system can understand a movie. That is the precondition for comparing similarities between two movies. Movie features such as genre, actor, and director are a way that can categorize movies.

So, we get these questions:

- How to recommend movies when there is no movie information.
- What kind of movie features can be used for the recommender system?
- How to calculate the similarity between two movies.

3. Project Objectives

- Suggest similar movies that have a higher probability of being liked based on the movie selected by the user.
- Providing related content out of the relevant and irrelevant collection of items to users of online service providers.
- Improve the Quality of the movie Recommendation system.

- Improving the Accuracy of the recommendation system.
- Improving Scalability, Enhancing the user experience.

4. Literature Review

The paper titled "Item-Based Collaborative Filtering Recommendation Algorithms" by Sarwar et al. (2001) is a seminal work in the field of recommender systems. In this paper, the authors propose an item-based collaborative filtering algorithm for making personalized recommendations.

The authors begin by discussing the limitations of traditional collaborative filtering algorithms, which rely on user-user similarity to make recommendations. One of the main limitations of user-based collaborative filtering is that it suffers from the sparsity problem, as users typically rate only a small subset of the items in the dataset. This can result in inaccurate and unreliable recommendations.

To address this problem, the authors propose an item-based collaborative filtering algorithm, which relies on item-item similarity to make recommendations. The algorithm works by computing the similarity between items based on the ratings given by users. Items that are rated similarly by users are considered to be similar to each other.

The authors then describe the details of the algorithm, including how to compute item-item similarity, how to make recommendations based on item-item similarity, and how to handle missing ratings. They also discuss the computational complexity of the algorithm and propose several optimization techniques to improve its efficiency.

To evaluate the effectiveness of their algorithm, the authors conduct experiments on two real-world datasets: MovieLens and Jester. They compare their item-based collaborative filtering algorithm with several other recommendation algorithms, including user-based collaborative filtering, content-based filtering, and random recommendation. They find that their item-based algorithm outperforms all other algorithms in terms of recommendation accuracy and coverage.

The authors conclude by discussing the implications of their work for the design of recommender systems. They argue that item-based collaborative filtering is a promising approach to making personalized recommendations, particularly in the presence of sparse data. They also suggest several directions for future research, such as incorporating item features and social network information into the algorithm.

In summary, the paper by Sarwar et al. (2001) is an important contribution to the field of recommender systems. The authors propose an item-based collaborative filtering algorithm that addresses the limitations of traditional collaborative filtering algorithms. They demonstrate the effectiveness of their algorithm through experiments on real-world datasets and suggest several directions for future research.

5. Methodology/ Planning of work:

To achieve the goal of the project, the first process is to do enough background study, so the literature study will be conducted. The whole project is based on a big amount of movie data, so we choose the quantitative research method.
For computing similarity between the different movies in the given dataset efficiently and in the least time and to reduce the computation time of the movie recommender engine we used the cosine similarity measure.

- Data collection: Collecting data from various sources like user ratings, reviews, and purchase history.

- Data cleaning: Cleaning the data by removing irrelevant data and filling missing values.

- Data pre-processing: Pre-processing the data by converting it into a user-item matrix where the rows represent the users, and the columns represent the items. The cells of the matrix represent the user-item interactions.

- Similarity computation: Computing the similarity between the items using measures like cosine similarity or Pearson correlation.

- Nearest neighbours' selection: Selecting the k-nearest neighbours based on the similarity scores.

- Rating prediction: Predicting the ratings of the items that the user has not yet interacted with based on the ratings of the items the user has interacted with and the similarity scores.

- Recommendation generation: Generating the top-n recommendations based on the predicted ratings.

- Evaluation: Evaluating the performance of the recommendation system using metrics like precision, recall, and mean average precision

PHASE 1:

- Define the objectives and requirements for the movie recommendation system.
- Conduct research on existing movie recommendation systems and choose the appropriate machine learning algorithms.

PHASE 2:

- Collect and clean the movie dataset to be used for training the recommendation system.
- Split the dataset into training and testing datasets.

PHASE 3:

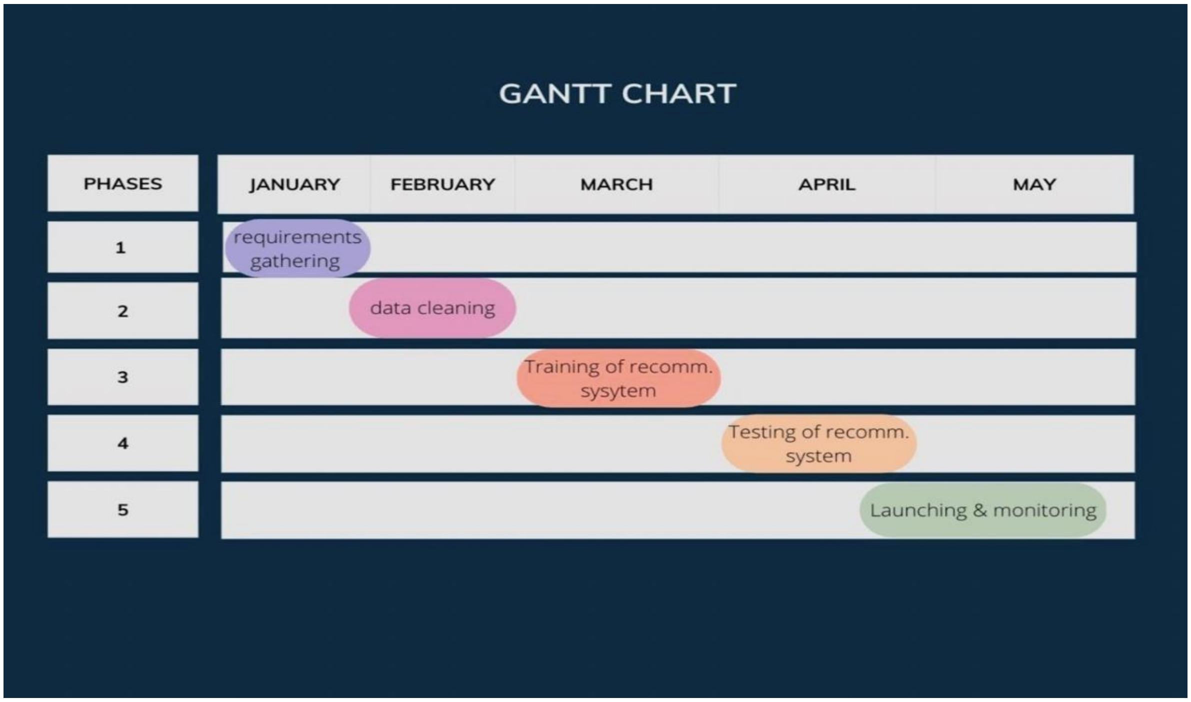
- Train the recommendation system using the training dataset.
- Evaluate the performance of the recommendation system using the testing dataset.

PHASE 4:

- Test the recommendation system with a sample group of users to get their feedback.

PHASE 5:

- Implement the recommendation system in a user-friendly interface.
- Integrate the recommendation system with the relevant movie platforms.
- Launch the movie recommendation system to the public.
- Monitor the performance of the recommendation system and collect user feedback.



5. Facilities required for proposed work:

5.1 Hardware Requirements

- A PC with Windows/Linux OS
- **Processor** with 1.7-2.4gHz speed
- **Minimum** of 4GB RAM
- **2**GB Graphic card

5.2 Software Requirements

- Text Editor (VS-code/WebStorm)
- Jupyter Notebook

The Jupyter Notebook is an open source web application that you can use **to create and share documents that contain live code, equations, visualizations, and text**. Jupyter Notebook is maintained by the people at Project Jupyter.

Python libraries

For the computation and analysis, we need certain python libraries which are used to perform analytics. Packages such as SKlearn, NumPy, pandas, etc. are needed.

SKlearn: It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means, and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

NumPy: NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object and tools for working with these arrays. It is the fundamental package for scientific computing with Python. Pandas: Pandas are one of the most widely used python libraries in data science. It provides high-performance, easy-to-use structures, and data analysis tools. Unlike the NumPy library which provides objects for multi-dimensional arrays, Pandas provides an in-memory 2d table object called a Data frame.

Pandas: Pandas is one of the most widely used python libraries in data science. It provides high-performance, easy-to-use structures, and data analysis tools. Pandas provide an in-memory 2d table object called a Data frame.

Bibliography/References

[1] Sharma, Ritu, Dinesh Gopalani, and Yogesh Meena. "Collaborative filtering-based recommender system: Approaches and research challenges." In *2017 3rd international conference on computational intelligence & communication technology (CICT)*, pp. 1-6. IEEE, 2017

Vedavathi, N., and R. Suhas Bharadwaj. "Deep Flamingo Search and Reinforcement Learning Based Recommendation System for E-Learning Platform using Social Media." *Procedia Computer Science* 215 (2022): 192-201.

Sarwar, Badrul, George Karypis, Joseph Konstan, and John Riedl. "Item-based collaborative filtering recommendation algorithms." In *Proceedings of the 10th international conference on World Wide Web*, pp. 285-295. 2001.