

Movie Recommendation System

Presented by:

102203580- Alok Sinha

102203632- Jyotish Yadav

Faculty Name:

Anjula Mehto

Content

Introduction

Objectives

Methodology

Working Model

Results

Conclusion

References

Introduction

Technical background of project :

A Movie Recommendation System is a type of machine learning application designed to suggest movies to users based on their preferences or similarities between items (movies) or users.

Technical Concepts used :

- Cosine Similarity
- Vectorization of Movie Attributes
- Feature Extraction

Motivation :

With the abundance of movie content available today, users often struggle to decide what to watch. A recommendation system simplifies this process by filtering options based on user preferences.

Introduction

Problem Statement :

To develop an efficient and scalable movie recommendation system that provides users with personalized suggestions based on similarity metrics.

Area of application :

This system can be applied to online streaming platforms, e-commerce websites, and personal movie libraries.

Introduction

Dataset and input format :

index	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	...	runtime	spoken_languages	status
0	0 237000000	Action Adventure Fantasy Science Fiction	http://www.avatarmovie.com/	19995	culture clash future space war space colony so...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	...	162.0	[{"iso_639_1": "en", "name": "English"}, {"iso...	Released
1	1 300000000	Adventure Fantasy Action	http://disney.go.com/disneypictures/pirates/	285	ocean drug abuse exotic island east india trad...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	...	169.0	[{"iso_639_1": "en", "name": "English"}]	Released
2	2 245000000	Action Adventure Crime	http://www.sonypictures.com/movies/spectre/	206647	spy based on novel secret agent sequel mi6	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	...	148.0	[{"iso_639_1": "fr", "name": "Fran\u00e7ais"}, ...]	Released

Objective

Main Objective :

To build a recommendation system that predicts movies a user might like based on cosine similarity of movie feature vectors.

Sub Objective :

- Preprocessing and feature extraction from movie datasets
- Representing movies as vectors based on attributes (e.g., genres, keywords)
- Implementing cosine similarity for recommendations

Methodology

Steps :

1.Data Preprocessing:

Cleaning the dataset, handling missing values, and encoding features (e.g., genres, directors) into vectors.

2.Vectorization:

Represent movies as vectors (e.g., $[1, 0, 1, 0]$).

3.Similarity Calculation:

Use cosine similarity to compute the angle between movie vectors, assigning higher similarity scores for similar movies.

4.Recommendation Engine:

For a given movie or user, sort and recommend the top N most similar movies.

Methodology

Steps :

Input

```
# getting the movie name from the user

movie_name = input(' Enter your favourite movie name : ')

Enter your favourite movie name : fight club
```

Similarity score

```
# getting a list of similar movies

similarity_score = list(enumerate(similarity[index_of_the_movie]))
print(similarity_score)

[(0, 0.0), (1, 0.0), (2, 0.0), (3, 0.03172085537943224), (4, 0.0),

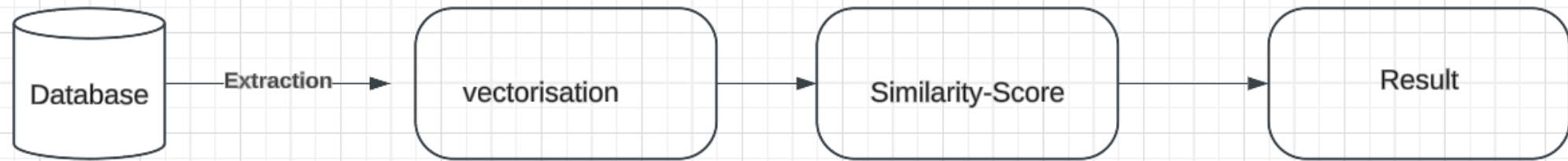
[ ] len(similarity_score)

4803
```

Result

Working Model

Technical Diagram



Flow Diagram

Working Model

Working Module

Steps to Describe:

1. Data Preprocessing:
 - Clean and structure the dataset (e.g., handling missing values).
 - Convert genres, ratings, and other features into vectors.
2. Feature Engineering:
 - Create a vector representation for each movie based on features (e.g., genres, average ratings).
3. Cosine Similarity Computation:
 - Compute cosine similarity between movies to find the closest matches.
4. Recommendation Algorithm:
 - Use similarity scores to recommend movies:

Working Model

Attained Deliverables

1. Preprocessed Dataset:

Cleaned, structured dataset with features such as genres, ratings, and movie descriptions transformed into a usable format for the system.

2. Vectorized Representations:

Numerical vectors for movies and user profiles created using feature extraction and encoding techniques.

3. Similarity Matrix:

A matrix containing pairwise cosine similarity scores between movies, representing how closely related they are.

4. Recommendation System:

A fully functional model that provides personalized movie recommendations based on user preferences.

Results

Tests Cases

1. Search for a movie

```
[49] # getting the movie name from the user

movie_name = input(' Enter your favourite movie name : ')

Enter your favourite movie name : Spider-man
```

2.Sort the movie on basis of their similarity score using cosine similarity

```
[56] # sorting the movies based on their similarity score

sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse = True)
print(sorted_similar_movies)

[(159, 1.0), (5, 0.3188331558421017), (30, 0.31791901982229703), (1559, 0.18281312771525624), (382, 0.16501718739122473), (3575, 0.1616704105513:
```

Results

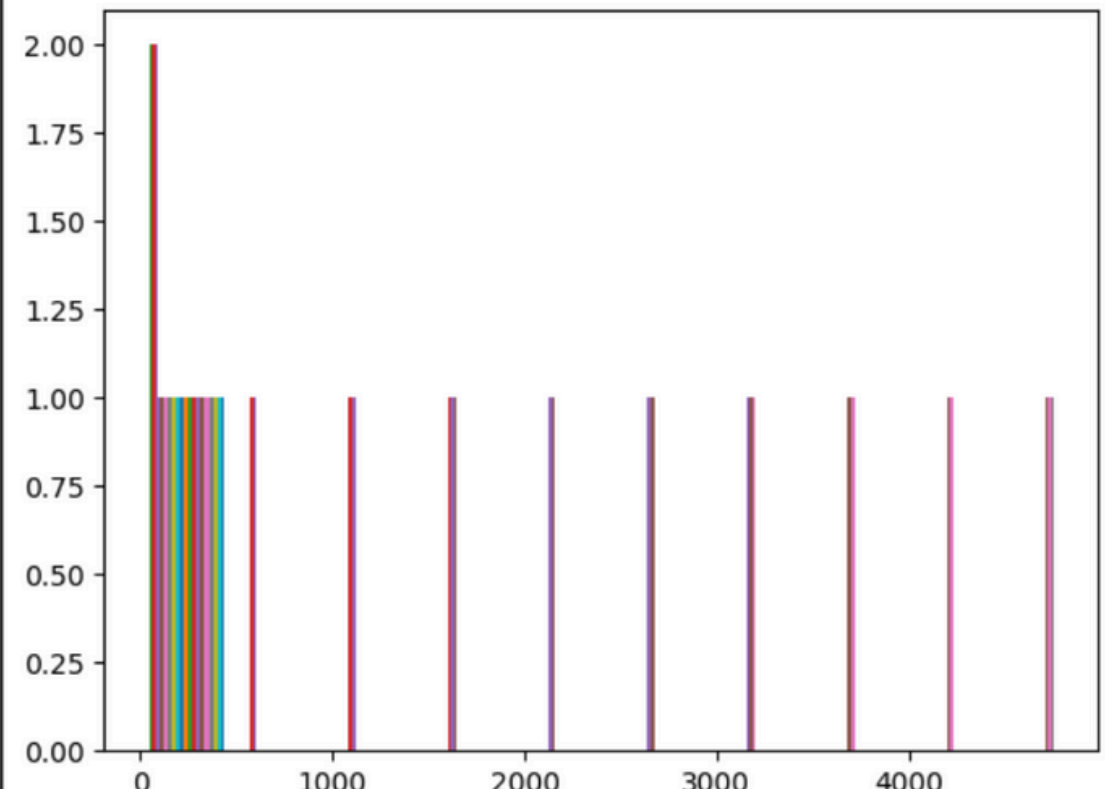
Tests Cases

3. Result

⇒ Movies suggested for you :

- 1 . Spider-Man
- 2 . Spider-Man 3
- 3 . Spider-Man 2
- 4 . The Notebook
- 5 . Seabiscuit
- 6 . Clerks II
- 7 . The Ice Storm
- 8 . Oz: The Great and Powerful
- 9 . Horrible Bosses
- 10 . The Count of Monte Cristo
- 11 . In Good Company
- 12 . Finding Nemo

Output Graph



Conclusion

Justification of Objectives:

The system successfully meets the objectives by efficiently recommending movies based on user preferences and cosine similarity.

Future Scope :

1. **Enhanced Recommendation**

Techniques Hybrid Recommendation

Systems:

Combine collaborative filtering, content-based filtering, and knowledge graphs to improve accuracy and diversity in recommendations.

2. **Multi-Platform**

Deployment Cross-Platform

Integration:

Deploy the system on multiple platforms, including web, mobile apps, and streaming services like Netflix or YouTube.

References

<https://www.kaggle.com/datasets/parasharmanas/movie-recommendation-system>



THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Thank You