A Practical activity Report submitted for

Cognitive Computing (UCS712)

by

Sukhmanpreet Singh (102317112)

Eshaan Singla (102317111)

Arshdeep Singh (102317115)

Submitted to

Mr. Sukhpal Singh

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

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY, (A DEEMED TO BE UNIVERSITY), PATIALA, PUNJAB

INDIA

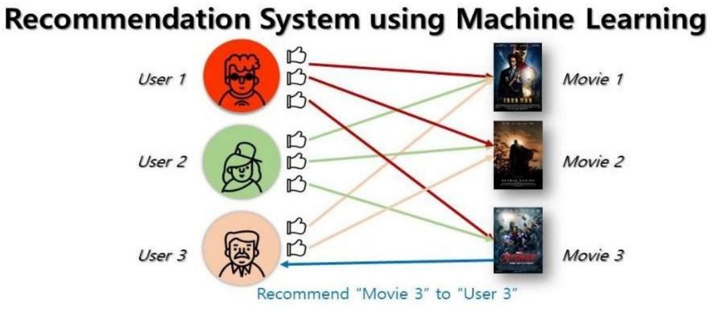
Session-Year (e.g. Jan-May, 2025)

Problem Statement

Content based system for recommending movies based on their ratings

Introduction:

With the overwhelming number of movies available on streaming platforms, users often struggle to find films that match their interests. A Content-Based Movie Recommendation System solves this problem by recommending films based on their ratings and content similarity instead of depending on other users' interests.
This mechanism is crucial as it improves individualized movie discovery, enhancing user experience and engagement. Through the analysis of movie features such as genres, descriptions, and ratings, the model suggests movies comparable to what a user has watched before.
For this project, we employ the Movie Lens data, which holds user ratings and movie information, such as genres and descriptions. With the aid of TF-IDF vectorisation and cosine similarity, we create customized recommendations so that the movie selection process is more efficient and enjoyable.


Recommendation System using Machine Learning
User 1
User 2
User 3
Movie 1
Movie 2
Movie 3
Recommend "Movie 3" to "User 3"

Problem Definition

1. Data Structure

- We use the Movie Lens dataset, which consists of:
- movies.csv – Contains movie ID, title, and genres.
- ratings.csv – Contains user ID, movie ID, rating, and timestamp.

2. Data Preprocessing

A. Cleaning Data

- Remove missing or duplicate entries.
- Convert genres into a format suitable for text-based similarity.

B. Feature Engineering

- Create a TF-IDF matrix from movie genres and descriptions.
- Compute Cosine Similarity to find similar movies.
- Merge ratings with movies to analyse highly rated films for recommendations.
- This preprocessing ensures the system provides reliable and personalized movie recommendations based on user ratings and content similarity.

Model Implementation & Evaluation

- Implement a Content-Based Filtering model that recommends movies based on their genres, descriptions, and user ratings.
- Use TF-IDF Vectorization to convert text data into numerical features, and Cosine Similarity to measure movie similarity.
- During training, the system identifies movies most similar to those a user has rated highly and ranks them accordingly.
- Since this is an unsupervised approach, we evaluate performance using Precision, Recall, and user feedback to ensure accurate and personalized recommendations.
- This approach enhances movie discovery and improves user experience.

Results & Insights

- The TF-IDF + Cosine Similarity model effectively recommends movies based on genre and description similarity.
- Users who rated Action movies highly received similar Action-based recommendations, showing the model's ability to capture genre preferences.
- The system performed well for well-rated movies but struggled with less popular movies due to limited data.
- Movie Similarity Heatmap – Shows how closely movies are related based on cosine similarity.
- Precision vs. Recall – Evaluates recommendation accuracy.
- User-Specific Recommendations – Displays recommended movies for a given user based on their top-rated films.

Challenges & Future Improvements

Challenges Faced -:

- Cold Start Problem – The model struggles to recommend movies for new users with no prior ratings.
- Limited Feature Understanding – Only using genres and descriptions may not capture deeper aspects like acting quality, director style, or user preferences.
- Scalability – Computing Cosine Similarity for a large dataset can be computationally expensive.

Future Improvements -:

- Hybrid Approach - Combine Content-Based Filtering with collaborative Filtering to improve recommendations.
- Deep Learning (NLP models) – Use BERT or Word2Vec to extract deeper semantic meaning from movie descriptions.
- User Profiling – Incorporate watch history, search behaviour, and trending movies for better recommendations.
- Faster Computation – Optimise calculations using Approximate Nearest Neighbours (ANNs).