Movie Recommendation System Using Content-Based Filtering

Submitted in partial fulfillment of the requirements of the degree

B.TECH IN ELECTRONICS AND COMPUTER SCIENCE

By

Shalaka Gangurde **22108B0027**

Supervisor
Prof. Rakshak Sood

Department of Electronics and Computer Science

Vidyalankar Institute of Technology

Vidyalankar Educational Campus,

Wadala(E), Mumbai - 400 037

University of Mumbai

(AY 2025-26)

CERTIFICATE

This is to certify that the Mini Project entitled **"Movie Recommendation System Using Content-Based Filtering"** is a Bonafide work of **Shalaka Gangurde** (**22108B0027**) submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"EXCS".**

Prof. Rakshak Sood

Supervisor

Internal Examiner

Name & Sign

External Examiner

Name & Sign

Table of Contents

Sr. No.	Chapter	Section	Page No.
1	Introduction	1.1 Background 1.2 Problem Statement 1.3 Objectives of the Study 1.4 Scope and Limitations	4
2	Literature Review	2.1 Overview 2.2 Types of Approaches 2.3 Related Work 2.4 Research Gap	5
3	Methodology	3.1 Overview 3.2 Dataset Description 3.3 Data Preprocessing 3.4 Feature Engineering and Transformation • TF-IDF Vectorization • Sigmoid Kernel Similarity 3.5 Implementation	6
4	Results and Discussion	4.1 Weighted Average Rating 4.2 Similarity-Based Recommendation 4.3 Visualization and Analysis	8
5	Comparative Evaluation	—	9
6	Conclusion and Future Scope	6.1 Conclusion 6.2 Future Scope	10
—	References	—	11

Chapter 1: Introduction

1.1 Background

With millions of movies available on platforms like Netflix, Prime Video, and Disney+, users face difficulty selecting movies of interest. Recommender systems solve this challenge by analyzing data and suggesting movies matching user preferences.

There are three main types of recommendation systems:

1. Collaborative Filtering: Based on user behavior and interactions.
2. Content-Based Filtering: Based on item attributes such as genres or overviews.
3. Hybrid Systems: Combine both methods for higher accuracy.
4.

This project focuses on content-based filtering, leveraging TF-IDF and the Sigmoid Kernel for similarity measurement.

1.2 Problem Statement

Traditional popularity-based or collaborative systems often fail in "cold-start" situations, where user data or ratings are unavailable. Hence, this project aims to design a content-based recommendation system that suggests movies using only their content (plot overviews).

1.3 Objectives

- Merge and preprocess TMDB 5000 Movies and Credits datasets.
- Clean and normalize textual data for feature extraction.
- Compute similarity using TF-IDF and Sigmoid Kernel.
- Build an interactive interface using Gradio.
- Visualize ranking metrics like popularity and weighted average.

1.4 Scope and Limitations

The system is restricted to content-based filtering and does not use collaborative or demographic data. However, it can be extended to hybrid systems for improved personalization.

Chapter 2: Literature Review

2.1 Overview

Recommendation systems enhance personalization and retention in digital platforms. Past studies have used similarity-based, collaborative, and hybrid models for effective recommendations.

2.2 Approaches

1. Content-Based Filtering: Uses metadata and textual data for similarity.
2. Collaborative Filtering: Relies on user-item ratings.
3. Hybrid Systems: Integrate both methods for better performance.

2.3 Related Work

Most prior works use cosine similarity over TF-IDF vectors, which captures linear relationships. This project introduces Sigmoid Kernel, a nonlinear similarity function that provides richer semantic associations between movies**.**

2.4 Research Gap

While existing systems rely heavily on user ratings, this approach leverages only textual similarity, making it robust for new or unrated movies.

Chapter 3: Methodology

3.1 Overview

The project follows the stages of:
Data Acquisition → Preprocessing → Feature Engineering → Model Building → Evaluation → Deployment.

3.2 Dataset Description

The TMDB 5000 Movies and Credits datasets contain metadata including: id, title, overview, cast, crew, popularity, vote_count, and vote_average. Both datasets were merged using the common key id (after renaming movie_id).

3.3 Data Preprocessing

- Removed irrelevant columns (homepage, status, etc.)
- Replaced missing overviews with blank strings
- Normalized popularity and rating values
- Applied feature scaling for weighted score computation

3.4 Feature Engineering

$$TF\text{-}IDF(t, d, D) = TF(t, d) \times IDF(t, D)$$

$$TF(t, d) = \frac{\text{Term Count in Document}}{\text{Total Terms in Document}}$$

$$IDF(t, D) = \log \frac{N}{1 + n_t}$$

This highlights rare and meaningful terms for similarity detection.

Sigmoid Kernel Function
A nonlinear kernel that enhances text similarity detection:

$$K(x, y) = \tanh(\alpha x^T y + c)$$

- **(**x, y): TF-IDF vectors
- (\alpha): Scaling factor

- (c): Offset constant

Outputs values between -1 and 1, capturing deeper semantic relationships.

3.5 Implementation Steps

1. Apply TF-IDF on all movie overviews.
2. Compute pairwise Sigmoid Kernel similarity.
3. Create give_recommendations() to return top 10 similar movies.
4. Build a Gradio UI for user interaction.
5. Visualize metrics using Plotly.

Chapter 4: Results and Discussion

4.1 Weighted Average Rating

$$W = \frac{(v \times R) + (m \times C)}{v + m}$$

where

- (v): Votes count
- (R): Average rating
- (m): Minimum votes
- (C): Mean rating

This ensures fair comparison across popular and lesser-known movies.

4.2 Recommendation Results

Example Input: “Iron Man 3”
Output Recommendations:
The Avengers, Thor, Captain America: The Winter Soldier, etc.
→ Movies with similar genres and storylines are correctly retrieved.

4.3 Visualization

Visualizations (via Plotly/Seaborn) illustrate:

- Relationship between popularity and weighted rating.
- Distribution of recommendation scores among top-ranked movies.

Chapter 5: Comparative Evaluation

Approach	Similarity Type	Flexibility	Accuracy	Remark
Popularity-Based	Linear	Low	Moderate	Simple but lacks personalization
TF-IDF + Cosine	Linear	Moderate	Good	Effective but limited to linear similarity
TF-IDF + Sigmoid	Non-Linear	High	Excellent	Captures complex semantic relationships

Sigmoid Kernel demonstrates superior flexibility and accuracy.

Chapter 6: Conclusion and Future Scope

6.1 Conclusion

The project successfully demonstrates a content-based movie recommendation system using TF-IDF and Sigmoid Kernel. It provides efficient and accurate results without relying on user ratings and is suitable for new movies.

6.2 Future Scope

- Integrate collaborative filtering for hybrid systems.
- Use deep learning embeddings (Word2Vec, BERT) for semantic similarity.
- Deploy as a web app with real-time updates.
- Add multilingual support for diverse users.

References

1. TMDB 5000 Movies and Credits Dataset – *Kaggle*.
2. *Scikit-learn* Documentation – TF-IDF Vectorizer & Sigmoid Kernel.
3. Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender Systems Handbook*, Springer.
4. Jurafsky, D., & Martin, J. (2023). *Speech and Language Processing*, Pearson.
5. *Gradio* and *Plotly* Official Documentation.
6. *Netflix Tech Blog* – Recommendation System Insights.
