Content Based Movie Recommendation System Using Feature Extraction

Bhagya A Koushik[1], Dr Ramya R S[2], Deekshitha R S[3], Venugopal K R[4], Deekshitha S[5]
Department of Computer Science and Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India. bhagyakoushik2001@gmail.com , rs.ramya.reddy@gmail.com , deekshithar13@gmail.com , venugopalkr@gmail.com , deekshareddy767@gmail.com

----------ABSTRACT----------

A recommendation system analyses the browsing history and user preferences and provides suggestions through a filtering technique. A movie recommendation system, or a movie recommender system, is a Machine Learning approach to predicting the users' film preferences based on their previous choices and behaviour. Content based Movie Recommendation System using Feature Extraction (CMRSFE) an advanced filtration technique that predicts the possible movie choices based on the user concerns and preferences towards a domain-specific item. In this work, the model mines the movie datasets to extract all the important information, such as, popularity, genres, keywords, overview, cast and crew, necessary for recommendation. The "TMDB-The Movie Database" dataset is made use of, to create the machine learning model for the recommendation system. Finally, the model recommends the top five movies as a recommendation to the user.

Keywords: **Content-based filtering, Count Vectorizer, Cosine similarity, Porter Stemmer.**

I.INTRODUCTION

A movie recommendation system has become an important part in social life due to its contribution to the entertainment industry. Based on the interests of the users, this system suggests a set of movies. Even though, there are many movie recommendation systems available, most of them are not capable of making efficient predictions. The objective of a movie recommendation system basically is to extract the necessary features in order to create a list of items for each user/individual and displays the movies accordingly. There are three types of filtering techniques which can be used i.e Collaborative filtering, Content Based filtering and Hybrid approach- which is a combination of both. Collaborative filtering technique is dependent on user-item matrix which identifies whether the user liked the item or not. It also makes use of ratings as an important criteria in model creation. Content based filtering technique uses the movie information in the dataset and recommends movies based on each and every feature in the feature set.

In this paper, a content-based movie recommendation method is used. The feature sets, such as movie_id, title, genre, overview, cast, director and keywords, that describe a movie are considered for recommending the top five movies.

Motivation:

In the existing schemes [1], the overall performance of the system with respect to recommendations was limited to a single feature 'genre'.

Contributions:

The contributions are summarized as follows:

Model creation using Count Vectorizer with feature sets, movie_id, title, overview, genre, cast, director and keywords.

II. RELATED WORKS:

SRS Reddy et al., [1] proposed Content Based Movie Recommendation System Using Genre Correlation that mines movies based on genre feature and extracts movies based on Euclidean distance in MovieLens dataset. This model does not take into consideration other features apart from genre.

Mahiye Uluyagmur et al., [2] proposed Content Based Movie Recommendation Using Different Feature Sets that convert the feature sets into ratings and recommend movies. Metrics such as Precision, Sensitivity, Specificity, Recall, F-Measure are used to analyse the accuracy of the overall model.

Syed M. Ali et al., [3] proposed Movie Recommendation System Using Genome Tags and Content Based Filtering that uses genome tags coupled with content-based filtering. This uses Principal Component Analysis and Pearson Correlation technique. This model is not very efficient since the technique used assumes a linear relationship between the features.

Souvik Debnath et al., [4] proposed Feature weighting in content based recommendation system using social network analysis that uses hybrid approach. The weight values are estimated from a set of linear regression equations obtained from a social network graph. This model is not efficient since the results of linear regression technique is not so accurate.

N. Pradeep et al., [5] proposed Content Based Movie Recommendation System that recommends movies based on different attributes of a movie. This model does not take into consideration the user's review about a movie.

Márcio Soares et al., [6] Tuning metadata for better movie content-based recommendation systems in which aspects such as the granularity of the descriptions is considered.

Accuracy metrics are used. The perspective analysis and evaluation of lists and diversity of results can be included for better efficiency.

Ramni Harbir Singh et al., [7] proposed Movie Recommendation System using Cosine Similarity and KNN that uses various deep learning approaches to implement Content based recommendation model. The efficiency can be improved by considering more attributes from the feature set.

Ajay Kaushik et al., [8] proposed a Movie Recommendation System using Neural Networks that uses hybrid approach which is a combination of both content based and collaborative techniques along with deep learning methodologies to recommend movies. Metrics are not used to evaluate the overall efficiency.

S. Rajarajeswari et al., [9] proposed Movie Recommendation System that uses a combination of Collaborative and Content based techniques, k-Means, Cuckoo Search, Movie Swarm to recommend a movie. Metrics are not used to evaluate the efficiency of the model.

III. FRAMEWORK: CONTENT BASED MOVIE RECOMMENDATION SYSTEM USING FEATURE EXTRACTION (CMRSFE)

A. Problem Definition

The user enters a movie name present in the dataset as the input. The model extract features such as movie_id, title, genre, keywords, overview, cast, crew from the dataset and provides recommendation of the top five similar movies as the output.

B. CMRSFE Framework

This CMRSFE framework retrieves the top five similar movies based on the multiple features extracted from the dataset. This technique involves the following phases:

(i) Dataset Linking
(ii) Data Pre-Processing
(iii) Vectorization
(iv) Distance Calculation
(v) Main Function
(vi) API Extraction

The modules are explained in following sections: (i) Dataset Linking Phase
In the research paper, dataset called TMDB movie dataset is used. TMDB dataset consists of two files 'tmdb_5000_movies' and 'tmdb_5000_credits'. The 'tmdb_5000_movies' file consists of various attributes such as 'budget', 'genres', 'homepage', 'id', 'keywords', 'original_language', 'original_title', 'overview', etc. The 'tmdb_5000_credits' file consists of attributes such as 'movie_id', 'title', 'cast', 'crew'. These objects are associated with features related to a particular movie. The dataset are in the .csv file format. The necessary attributes are extracted like movie_id, title, genre, keyword, overview, cast and crew.

(ii) Data Pre-Preprocessing Phase

In this phase, the contents in the dataset are pre-processed to remove all the null values and duplicate values. The attributes are converted into list of strings using ast.literal_eval() library for further access. The attributes are pre-processed by user-defined functions to extract only the necessary content. Transformation is applied for the creation of 'tags' column which is a combination of 'genre', 'keywords', 'cast', crew', 'overview' columns using lambda function.

(iii) Vectorization Phase

In this phase, The Bag of Words technique is used for vectorization. Count Vectorizer from sklearn.feature.extraction.text library is used to convert the 'tags' column into vectors. The max_features is set to 5000 and all the stop words are removed from the 'tags' column.

Followed by this, Stemming is done using PorterStemmer from nltk.stem.porter library. A helper function is used to achieve stemming. The changes made are saved back to vectors.

(iv) Distance Calculation Phase

In this phase, Cosine Similarity technique is used as the distance measure to identify the distance between the vectors (angle) to recommend five movies with least distance. cosine_similarity()

$$\cos\theta = \frac{u.v}{\|u\|\|v\|\cos\theta}$$

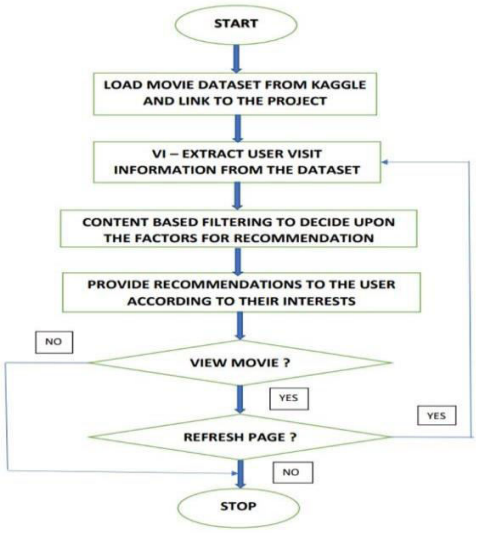
and enumerate() function is used for this purpose.

(v) Main Function Phase
In this phase, a user-defined function called 'Recommend' is used to identify the top five most similar movies from the distances (based on movie index) calculated using cosine_similarity and enumerate function. This recommends the top five movies when called with the movie title.

(vi) API Extraction Phase
In this phase, the posters for each movie is extracted through the API from the api.themoviedb.org site and is linked with the movie index using Streamlit library. When the recommend() function is called the top five

similar movies along with their posters are displayed.

Flowchart:



Algorithm: Content-Based Movie Recommendation System Using Feature Extraction

Input: 'tmdb_5000_movies' and 'tmdb_5000_credits' csv files.

Output: Top five similar movie recommendation.

begin

Step 1: Data Pre-processing to format the attributes of the dataset into an accessible form.

Step 2: Vectorization

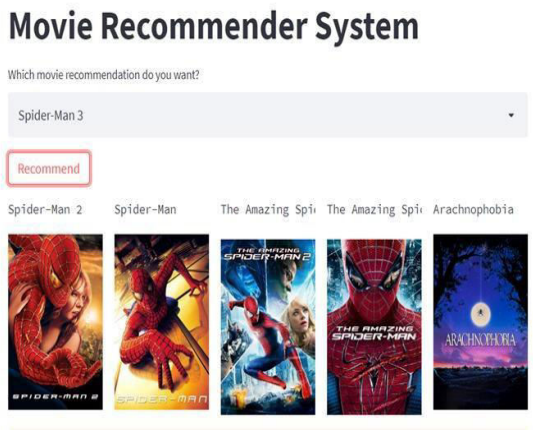
```
from sklearn.feature_extraction.text import
CountVectorizer
vectors=cv.fit_transform(new_df['tags']).toarr
ay()
new_df['tags']=new_df['tags'].apply(stem)
```

Step 3: Distance Calculation

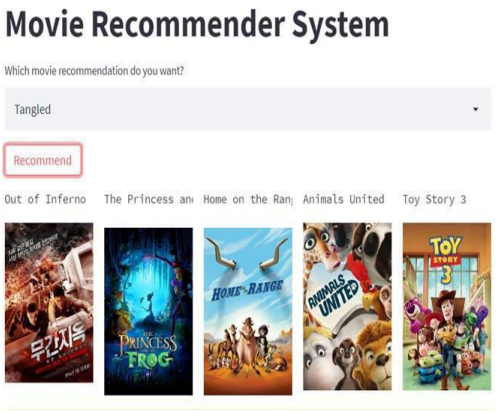
```
from sklearn.metrics.pairwise
import cosine_similarity

similarity=cosine_similarity(vect
ors) Step 4: Main Function
(recommend)
movies_list=sorted(list(enumerate(distances)),
reve rse=True,key=lambda x:x[1])[1:6]
```

end

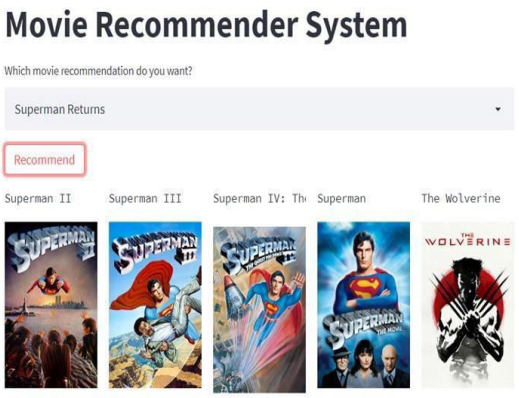
IV. RESULTS



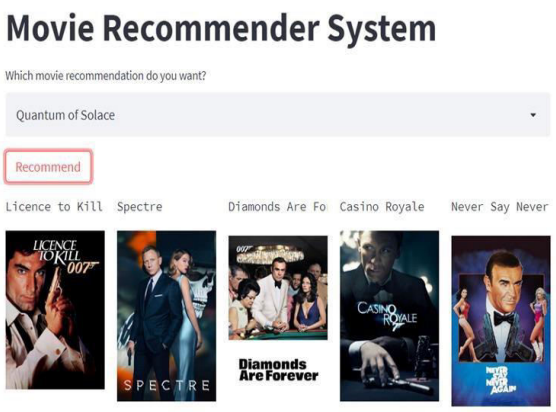
Genre: Sci-Fi – Fantasy



Genre: Animation



Genre: Sci-Fi Animation



Genre: Suspense Thriller

V. CONCLUSIONS AND FUTURE WORK

How to describe a movie, is the most important task because the more accurate a movie is described, the better results recommender system generates. So, from this perspective, an attempt is made for improvement of approach in content-based recommendation System.

Firstly, a content-based recommender algorithm is used, which means there is no cold start problem. Then, the features in the recommender system are listed. Some of them are required for consideration and some are not. Then, the cosine similarity technique is used which is commonly used for calculating the similarities between the different movies.

Based on the results, five different movies are recommended for a specific movie that has been selected.

This model can be improved by introducing metrics to check the efficiency of the recommendations. A Hybrid model can be used which is the combination of collaborative and content-based filtering techniques to improve the efficiency of the overall model.

REFERENCES

[1]https://link.springer.com/chapter/10.1007/978-981-13-1927-3_42

[2]http://www.iaeng.org/publication/WCECS2012/WCECS2012_pp517-521.pdf

[3] https://link.springer.com/chapter/10.1007/978-981-10-8360-0_8

[4]https://dl.acm.org/doi/abs/10.1145/1367497.1367646

[5]http://www.riejournal.com/article_121501_a3717e6cf19a1845e350acb9148751ee.pdf

[6] https://sci-hub.hkvisa.net/10.1007/s11042-014-1950-1

[7]https://www.researchgate.net/publication/344627182_Movie_Recommendation_System_using_Cosine_Similarity_and_KNN?enrichId=rgreq-98ca3089141c53a4899f8f9fcc2be906-XXX&enrichSource=Y292ZXJQYWdlOzM0NDYyNzE4MjtBUzo5NDYxMDQ0OTM0MjQ2NDBAMTYwMjU4MDI5NzA5MA%3D%3D&el=1_x_2&_esc=publicationCoverPdf

[8]https://d1wqtxts1xzle7.cloudfront.net/56235068/V4I2-1373-with-cover-page-v2.pdf?Expires=1657878217&Signature=V95p-TroVBRQOBwR5nZ1njkcDRsJvAmsPsKYeo4H39yLJKelrtxyQdO5qRa5Cs18ikEXFs-as7OuuTnxYitCbK5P4nP-A6lYdd3nmuUybvpdP77OpMZRP~WyRezUvXEQ2DtUchptW~eb8JyZS1mGqyEu9qlfeRgRErJ9m8jR34oWffuQXS2rltvTDt7k4nPp06sV9YImBk4-lwa5AA5E45dTnEbtxwUbVIz17B108AaFY1~wm4Yp7C86lG5avmi4mErXvW~9rrrtwb7ATq8Z3rNuPUqLv0hQT6gHN8154kgVT36aaUTUZOmnaneEHEWVSoTCaB1g2eM-nSEKVPBKlw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

[9] https://sci-hub.hkvisa.net/10.1007/978-981-13-5953-8_28