INFO 7390: ADVANCES IN DATA SCIENCES AND ARCHITECTURE

**FINAL PROJECT**

**TOPIC: MOVIE RECOMMENDER SYSTEM**

**Team Members:**

Abhijit Kunjiraman -001539524

Dipti Dubey -01548499

Merwin Roy -002959569

Raveena Yadav -002106277

# ABSTRACT

This project implements content-based and collaborative filtering recommender systems approach to implement an algorithm that provides relevant movie recommendations to users with the help of Netflix and Movie lens datasets from Kaggle, as well as sentiment analysis on reviews of movies recommended by the system.

The project report provides a summarized description of the project **“Netflix Movie Recommender System”,** models used to implement the recommender system, sentiment analysis and conclusions derived through it using different metrics obtained.

# INTRODUCTION

Recommender systems are information system filtering models that provides relevant data, content, suggestions to users involving a variety of decision-making processes, aiding the users to select a product or a service from overwhelming set of similar looking options. These systems have ubiquitous use case from e-commerce websites recommending products to users for purchase to music applications recommending songs, albums or artists that users might be inclined to add onto their playlist.

In this project, we have focused primarily on Netflix platform which makes great use of recommender systems to suggest movies and shows to users based on variety of factors such as genres, ratings, interests of similar users, viewership patterns etc. Taking Netflix movie dataset into consideration, we have implemented different recommendation engines and sentiment analysis models to suggest similar movies to users along with their reviews.

For the content-based recommendation, the system makes use of cosine similarity metric to measure the similarity between movies from datasets based on different features such as movie genre, actors, and director names.

For collaborative filtering recommender system, the system makes use of Pearson’s correlation to measure similarity between users.

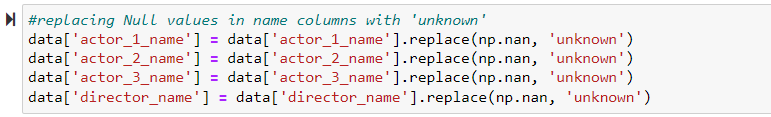
The system also makes use of sentiment analysis, a natural language processing technique (NLP) technique to analyze and identify states and subjective information of reviews associated with movies recommended to users by the recommender system. The sentiment analysis has been implemented using CountVectorizer, CNN (Convolutional Neural Network) Model and LSTM-CNN model which combines the traditional neural networks CNN and LSTM (Long Short-Term Memory) model.

The goal of the project is to study and compare different recommender systems (collaborative and content based) as well as different sentiment analysis models (CountVectorizer, CNN, LSTM-CNN) through metrics obtained and eventually provide near perfect movie recommendation results to users with reviews obtained through the process of sentiment analysis.

# METHODS

## Data Preprocessing & Data Cleaning:

* Null Values replaced with ‘unknown’ for actor and director name features in dataset



* Replacing ‘|’ to highlight genres with spaces instead



* Converting movie titles to lower case

Text

Description automatically generated

A picture containing text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* Dropping duplicate movie titles

Graphical user interface

Description automatically generated

* Dropping null values from dataset

Graphical user interface, text, application, email

Description automatically generated

* Combining actor names, director names and genres into a single column

Graphical user interface, application

Description automatically generated

## 2. Model Evaluation:

Multinomial Naïve Bayes, CNN and CNN-LSTM models have been implemented for the sentiment analysis of movie reviews.

**Multinomial Naïve Bayes model evaluation:**

1. Plotting ROC Curve:

Chart, line chart

Description automatically generated

## Plotting Precision-Recall Curve:

## Table Description automatically generated with medium confidence

## CNN (Convolution Neural Network) model evaluation:

## Plotting Training and Validation Accuracy

## Chart Description automatically generated

## Plotting Training and Validation Loss

## Chart Description automatically generated

## 

## CNN-LSTM (Long Short-Term Memory) model evaluation:

## Plotting Training and Validation Accuracy

Chart, line chart

Description automatically generated

1. Plotting Training and Validation Loss

Chart, line chart

Description automatically generated

## 3. SENTIMENTAL ANALYSIS:

## For sentiment analysis technique, Natural Language toolkit (NLTK) has been implemented to fetch a corpus of stopwords which are the most commonly used words in data which helps in removing unnecessary words that might affect the recommendations provided by the system.

## 3.1 Multinomial Naïve Bayes

## TF-IDF Vectorizer is then implemented to calculate TF-IDF score for each word used in the document or data in the form of a vector which counterbalances the commonly used words in the entire dataset in order to understand the context of review dataset and capture the subjective data accordingly.

## Multinomial Naïve Bayes Classifier is then implemented to classify words and textual data into classes based on their statistical analysis.

## Accuracy of the Sentimental analysis yielded using Multinomial Naïve bayes Classifier

## Graphical user interface, application Description automatically generated

## Plotting the ROC Curve

Text

Description automatically generated

Chart, line chart

Description automatically generated

## Plotting Precision-Recall Curve

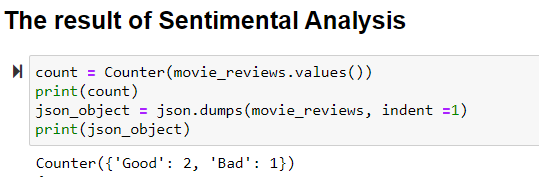


Table

Description automatically generated with medium confidence

From the following plots, we can observe that area covered by the precision-recall curve line of each class is more than 99% which concludes that the model is performing well in predicting classes even in case of imbalanced data.

1. The result of Sentimental Analysis that we obtained from the model



Graphical user interface, text, application, email

Description automatically generated

## 3.2 CNN Model

## Before training the model for CNN, training and testing textual data is cleaned and then tokenized into smaller units such as words or terms. Then the data is padded to process it by the CNN kernels.

## Graphical user interface, text, application Description automatically generated

## Accuracy for CNN Model

## Chart Description automatically generated

## Chart Description automatically generated

## Chart Description automatically generated

## 3.3 LSTM-CNN Model

## CNN-LSTM is the third model architecture implemented for sentiment analysis of movie reviews which integrates  CNN model for the input and an LSTM model to process input time step processed and yielded by the CNN model.

## Accuracy for LSTM-CNN Model

## Text Description automatically generated

## Chart, line chart Description automatically generated

## Chart, line chart Description automatically generated

## 4. Recommender Engines

## 4.1 Content Based Filtering [Using Cosine Similarity Measure]

## CountVectoizer tool has been implemented to obtain a count matrix of words used in different documents.

## In the case of the movie dataset, the system takes into consideration features such as actor and director names and movie genre to describe the frequency of words used in the form of a vector matrix before implementing it into a cosine similarity method to measure similarity between different movies based on the selected features.

## Graphical user interface, text, application, email Description automatically generated

## Graphical user interface, text, application, email Description automatically generated

## 4.2 Collaborative Filtering [Using Pearson’s Correlation Coefficient(PCC) Measure]

## Collaborative Filtering recommender system is implemented in the project using Pearson’s Correlation Coefficient (PCC) measure to evaluate the correlation between different users on features such as movie rating scores given by different users to each movie.

## 

## Text Description automatically generated

## Text Description automatically generated

## RESULTS AND CONCLUSION

|  |  |
| --- | --- |
| **Model** | **Accuracy (%)** |
| Multinomial Naïve Bayes | 97.47 |
| CNN | 99.13 |
| CNN- LSTM | 98.55 |

After a comparative study on the models above, we conclude that the classification model using

the Convolutional Neural Network (CNN) is the best model with an approx. accuracy score of 99.13 %.