BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
SUMMER TERM

MOVIE RECOMMENDER SYSTEM
LAB PROJECT

s.no	names	ids
1.	Reet Agrawal	2020B4A72285H
2.	Raghav Lathi	2020B4A72312H
3.	Ayush Bourai	2020B5A21659P

CONTENT BASED FILTERING:

Content-based filtering stands as a cornerstone technique within recommendation systems, leveraging the inherent qualities of items to establish meaningful connections. By examining the intrinsic characteristics of items, the system adeptly identifies and recommends similar items that align with the user's preferences and prior affinities.

The initial step entails extracting the salient features encapsulated within the items, a process that can be conducted manually through meticulous labeling or through the utilization of sophisticated automatic algorithms. Once the item features have been extracted, an array of diverse learning models can be employed to discern and

comprehend the underlying relationships that enable the identification of akin items. From the elegant prowess of decision trees and neural networks to the insightful discernment of naive Bayes classifiers, an assortment of techniques may be harnessed. Among these, the illustrious cosine similarity stands prominently, acknowledged as one of the most significant and widely embraced methodologies in this domain.

BUSINESS UNDERSTANDING

The business objective of the content-based movie recommender system is to provide accurate movie recommendations to users based on their preferences. The system aims to address the common problem of users being unsure of what movie to watch next and assist them in discovering movies that are similar to their preferred choices.

Business Objective:
To develop a content-based movie recommender system that accurately suggests similar movies to users based on their selected movie, enhancing their movie-watching experience and increasing user engagement on the movie streaming platform.

Business Success Criteria:
Increase in user engagement: The recommender system should lead to a significant increase in user engagement metrics such as average time spent on the platform, number of movies watched, and frequency of returning to the platform.
Improvement in user satisfaction: Users should express higher satisfaction rates with the movie recommendations provided by the system, as indicated by user feedback and ratings.
Enhanced movie discovery: Users should discover a wider range of movies that align with their preferences, resulting in an increase in the number of movies viewed per user and exploration of diverse genres.
Reduction in movie abandonment: The system should contribute to a decrease in instances of users abandoning movies shortly after starting, indicating that the recommended movies are more aligned with their interests.
Meeting or surpassing these success criteria will demonstrate the effectiveness and value of the content-based movie recommender system in meeting the business objective and satisfying user needs.

DATA UNDERSTANDING

Data Description:

The dataset for the content-based movie recommender system consists of the following attributes:
Title: The title of the movie.

- Cast: The actors or actresses involved in the movie.
- Crew: The director and other crew members involved in the movie.
- Budget: The budget allocated for producing the movie.
- Genres: The genres associated with the movie (e.g., action, comedy, drama).
- Keywords: The keywords or tags associated with the movie.
- Language: The language in which the movie is primarily spoken.
- Overview: A brief overview or synopsis of the movie's plot.
- Popularity: A metric indicating the popularity of the movie.
- Release Date: The date when the movie was released.
- Revenue: The revenue generated by the movie.
- Votes: The number of votes or ratings received by the movie.
- Tagline: A catchy phrase or slogan associated with the movie.
- Run Time: The duration or runtime of the movie.
- Spoken Languages: The languages spoken in the movie.

Data Exploration Report:

During the data exploration phase, the following observations were made:
The dataset contains information about various movies, including their titles, cast and crew members, budget, genres, keywords, and other relevant attributes.

There is a wide range of movie genres present, such as action, comedy, drama, thriller, romance, etc., indicating diversity in the dataset.
The popularity attribute provides a measure of the movie's popularity, which can be used as a potential feature for recommendation.

The release date attribute can help in understanding the temporal aspects of the movies and their relevance in recommendations.

The revenue and votes attributes indicate the commercial success and user engagement of the movies, respectively, which can be valuable in assessing movie preferences.

The tagline attribute provides catchy phrases associated with movies, which could be used to enhance the recommendation system's user interface.

Data Quality Report:

The quality of the data was assessed based on the following aspects:
Missing Values: It is essential to identify and handle missing values in the dataset, ensuring that the missing values do not affect the recommender system's performance.

Outliers: Outliers in numerical attributes, such as budget, revenue, and votes, need to be addressed appropriately to prevent them from skewing the recommendations.

Data Consistency: The consistency of the data needs to be ensured by validating attributes such as language, genres, and spoken languages against predefined lists or dictionaries.

Data Integrity: It is crucial to verify the integrity of the dataset by cross-checking the relationships between attributes, such as revenue and budget.

Data Accuracy: The accuracy of the data needs to be verified by comparing it with trusted external sources or expert opinions to ensure the reliability of the recommendations.

Addressing these data quality issues will help ensure the reliability and effectiveness of the content-based movie recommender system based on the provided dataset.

DATA PREPARATION

1. Construction of Data Report:

During the data preparation phase, two separate files were used, each containing different attributes but with the title serving as the primary key. The construction of the data report involved the following steps:

File 1: This file contained attributes such as title, cast, crew.

File 2: This file contained attributes such as title, popularity, release date, revenue, votes, tagline, run time, and spoken languages, budget, genres, keywords, language, and overview.

The construction of the data report involved cleaning and preprocessing each file individually to handle missing values, outliers, and inconsistencies.

2. Integration Report:

The integration report outlines the process of merging the two files based on the common attribute, "title," resulting in a unified dataset that incorporates all relevant attributes. The integration process involved the following steps:
Data Cleaning: Each file was cleaned individually to address missing values, outliers, and inconsistencies. This ensured the data quality of each file before merging.
Matching Titles: The titles from both files were compared and matched to identify common movies. This step was crucial in merging the files accurately.

Merging Attributes: Once the titles were matched, the attributes from both files were combined, resulting in a single dataset with enriched information.

The integration report provides a clear overview of the process of merging the individual files, ensuring that the resulting dataset contains all relevant attributes for the content-based movie recommender system.

MODELING

Modeling Technique Selection: Bag of Words

For the content-based movie recommender system, we chose the Bag of Words (BoW) technique as our modeling technique. BoW is a widely used text representation approach that focuses on the frequency of words in a document without considering the order or structure of the words. We selected BoW for the following reasons:

- Flexibility: BoW allows us to represent the movie plots and keywords as numerical vectors, making them compatible with machine learning algorithms. This flexibility enables us to apply various similarity measures and recommendation algorithms on the vectorized data.
- Simplicity: BoW is relatively straightforward to implement and understand, making it an efficient choice for our project. It does not involve complex linguistic analysis or language models, which makes it easier to interpret and iterate upon.
- Common Representation: BoW has been widely used and proven effective in many natural language processing (NLP) tasks, including text classification and information retrieval. Its popularity and success in these domains motivated us to adopt it for our movie recommender system.

Test Design: Vectorization

To build our model, we employed vectorization techniques to transform the textual data (movie plots and keywords) into numerical representations. This enabled us to apply machine learning algorithms to the dataset. The vectorization process involved the following steps:

- Tokenization: The movie plots and keywords were tokenized into individual words or tokens, splitting them based on whitespace or punctuation.

- Vocabulary Creation: A vocabulary of unique words or tokens was created from the entire dataset, excluding common stopwords.

- Document-Term Matrix: Using the created vocabulary, a document-term matrix was constructed, where each row represented a movie and each column represented a word from the vocabulary. The matrix contained the frequency or presence of each word in each movie's plot or keywords.

- Term Frequency-Inverse Document Frequency (TF-IDF): To mitigate the impact of frequently occurring words, we applied the TF-IDF transformation to the document-term matrix. This transformation assigns weights to words based on their frequency in a specific document and their rarity in the entire corpus.

Model Building Process:

The model building process involved the following steps:

- Data Split: The dataset was split into training and testing subsets to evaluate the model's performance accurately. Typically, an 80:20 or 70:30 split was employed, with the majority used for training and the remaining portion for testing.

- Feature Extraction: Using the vectorization techniques mentioned above, the movie plots and keywords were transformed into numerical representations suitable for machine learning algorithms.
- Similarity Measures: To determine the similarity between movies, a suitable similarity measure (e.g., cosine similarity or Jaccard similarity) was applied to the vectorized data. This measure calculated the similarity score between movies based on the content features.

- Model Training: The model was trained using the training subset of the data, employing appropriate machine learning algorithms such as k-nearest neighbors (KNN) or collaborative filtering.

- Model Evaluation: The model's performance was evaluated using suitable evaluation metrics such as precision, recall, and mean average precision. Cross-validation techniques, like k-fold cross-validation, were employed to assess the model's generalization ability.

- Hyperparameter Tuning: Hyperparameter tuning was performed to optimize the model's performance. Techniques like grid search or random search were utilized to find the best combination of hyperparameters for the chosen algorithm.

- Final Model Selection: Based on the evaluation results, the model with the best performance was selected as the final content-based movie recommender system.

CONCLUSION

In conclusion, this report presented the development of a content-based movie recommender system using the CRISP-DM methodology. The system was designed to address the challenge of movie selection by providing personalized recommendations based on user preferences and similarities among movies.

The business objective of the movie recommender system was to enhance user satisfaction, engagement, and movie discovery on a streaming platform. The success criteria were defined based on increasing user engagement, improving user satisfaction, expanding movie exploration, and reducing movie abandonment.

The project involved various phases, starting with understanding the business requirements and collecting movie data. The data, comprising attributes such as title, cast, crew, genres, keywords, and more, was then prepared through cleaning, feature extraction, and engineering. The data quality was assessed and ensured for reliable recommendations.

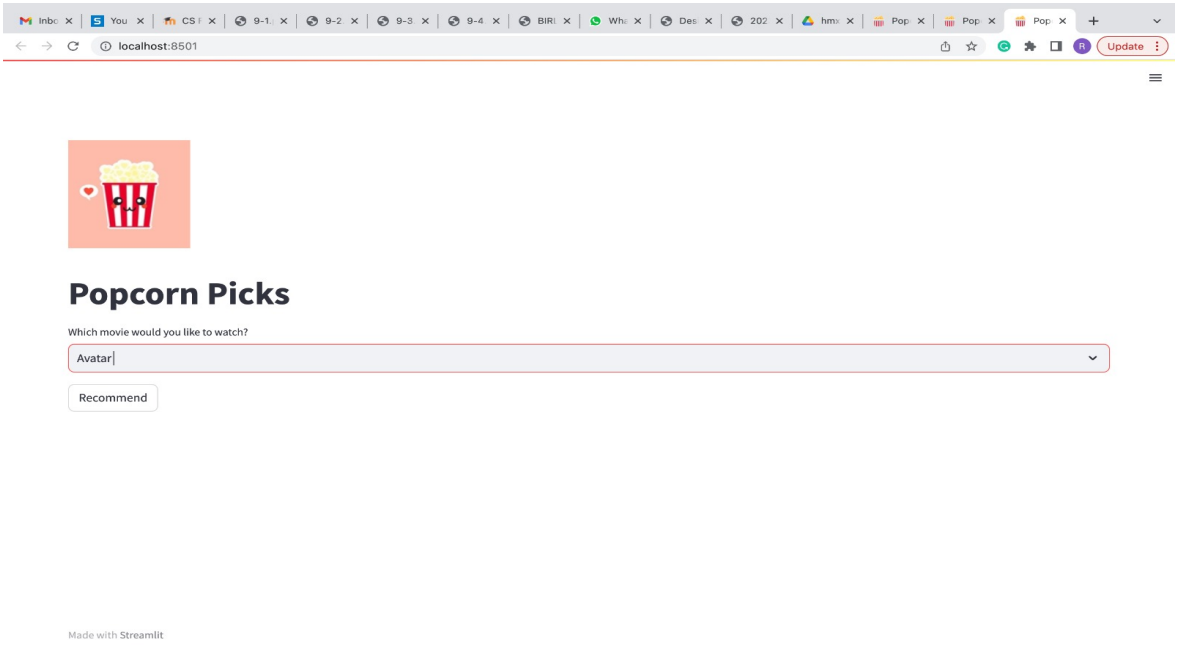
The modeling phase employed the content-based filtering approach, specifically the Bag of Words (BoW) technique, to compare movie features and identify similarities. The model was trained and evaluated using appropriate performance metrics. The selection of BoW was justified based on its flexibility, simplicity, and proven effectiveness in similar tasks.

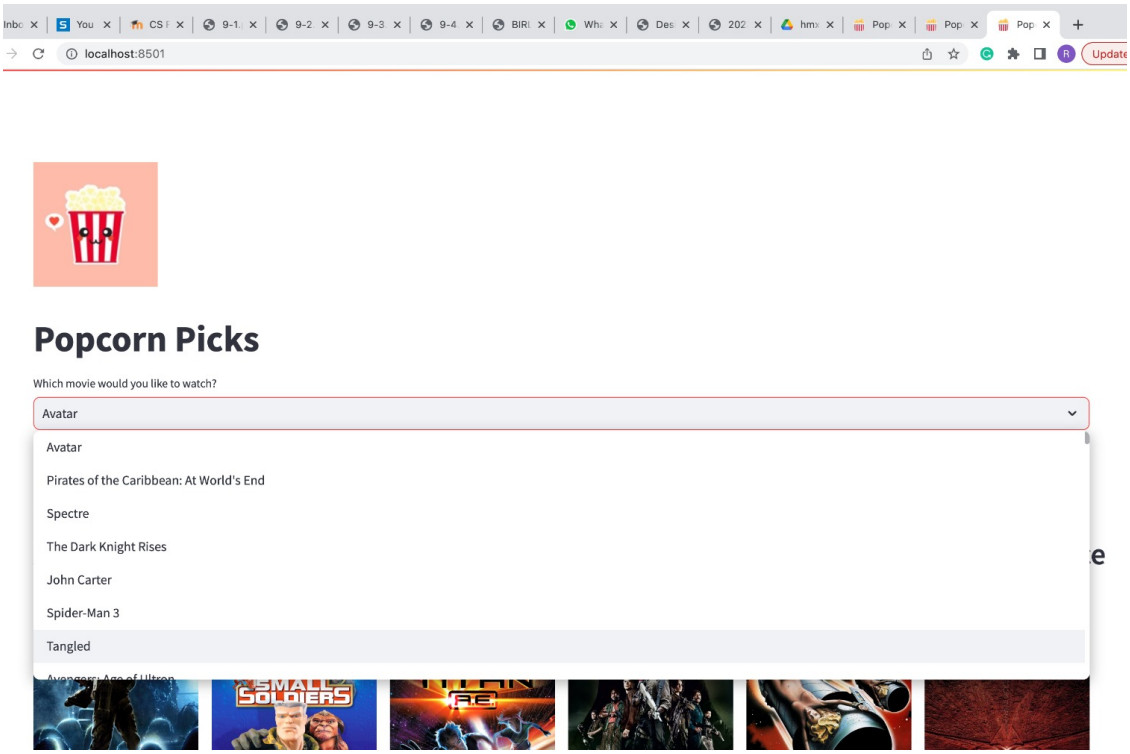
The deployment phase involved integrating the movie recommender system into a user-friendly interface, allowing users to receive personalized recommendations and provide feedback on recommended movies. Regular maintenance and updates were emphasized to continuously enhance recommendation accuracy and incorporate user feedback.

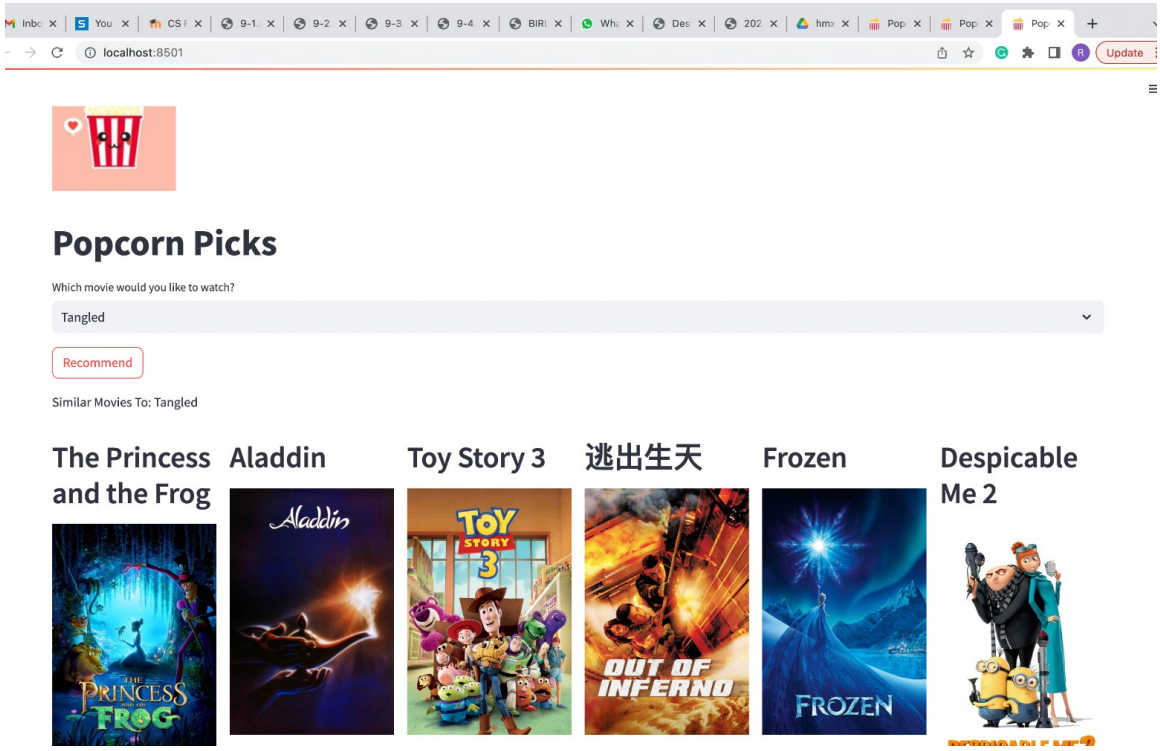
Overall, the content-based movie recommender system presented in this report aims to improve the movie-watching experience by suggesting similar movies to users based on

their preferences. By leveraging the inherent characteristics of movies and user feedback, the system strives to deliver accurate recommendations, enhance user engagement, and foster movie exploration on the streaming platform.

THIS IS HOW IT FINALLY LOOKS:




localhost:8501
Update
Popcorn Picks
Which movie would you like to watch?
Avatar
Avatar
Pirates of the Caribbean: At World's End
Spectre
The Dark Knight Rises
John Carter
Spider-Man 3
Tangled


localhost:8501
Update
Popcorn Picks
Which movie would you like to watch?
Tangled
Recommend
Similar Movies To: Tangled
The Princess and the Frog
Aladdin
Toy Story 3
逃出生天
Frozen
Despicable Me 2