

To build a Content based Movie Recommendation system based on Movie tags

NAGACHAITANYA S

Final Thesis Report  
MS IN DATA SCIENCE  
Liverpool John Moores University

DECEMBER 2023

## **Acknowledgement**

First and foremost, I extend my deepest gratitude to my family, with a special mention to my mother and sister. Their unwavering belief in me and constant encouragement have been the pillars that propelled me forward in my academic journey. Their support has been instrumental in my pursuit of continued studies.

I am profoundly thankful to my wife, whose unwavering support has been a constant throughout my academic journey. I am particularly grateful for her teachings on various hacks and tips for effective document formatting.

A heartfelt thanks goes to my dedicated supervisor, Mr. Tarun Duggal. His consistent support and guidance have played a pivotal role in the success of my Master's studies and related research. His mentorship has been invaluable, providing me with the necessary direction and insights.

I would also like to express my sincere appreciation to Dr. Rupal for her informative recorded videos, which have been an invaluable resource in my academic endeavours. Additionally, my gratitude extends to Dr. Manoj for his live sessions, where he generously shared his knowledge and expertise, addressing all doubts related to thesis writing. Their contributions have significantly enriched my learning experience.

In conclusion, I am deeply thankful to all those who have played a part in my academic journey, contributing to my growth and success. Your support and guidance have been indispensable, and for that, I am truly grateful.

## **Abstract**

In the rapidly evolving landscape of the movie industry, new organizations face challenges in providing personalized movie recommendations to users due to limited initial user profiles and item ratings. This research presents a comprehensive approach to address this challenge by developing an optimized content-based movie recommendation system. By leveraging insights from EDA, in structuring home page of a website and implementing a search bar for users to input movie names, the project aims to tackle the 'cold start' problem

In this paper we have analysed Feature selection techniques Chi-square on categorical columns and Pearson correlation test on Numerical columns to determine the relation between the features and user ratings. NLTK methods are employed for data pre-processing and TF-IDF - vectorizer for converting textual data to numerical data. Tags created with different sets of features are evaluated using cosine similarity. Later Weights added to different features are evaluated. This thesis also discussed how to leverage insights from EDA to be incorporated in structuring a website Homepage.

## 1. Contents

Acknowledgement .....	2
Abstract .....	3
1 CHAPTER 1: INTRODUCTION .....	10
<b>1.1 Background of the Study: .....</b>	<b>10</b>
<b>1.2 Problem Statement.....</b>	<b>11</b>
<b>1.3 Aim and Objective .....</b>	<b>12</b>
<b>1.4 Research Questions .....</b>	<b>13</b>
<b>1.5 Scope of the Study .....</b>	<b>13</b>
1.5.1 Cold Start Problem: .....	13
1.5.2 Feature Combinations:.....	13
1.5.3 Model Development and Optimization: .....	13
1.5.4 Website Integration: .....	14
1.5.5 Deployment: .....	14
1.5.6 Limitations:.....	14
1.5.7 Dataset: .....	14
1.5.8 Evaluation Metrics:.....	14
1.5.9 Statistical Significance: .....	14
<b>1.6 Significance of the Study.....</b>	<b>14</b>
<b>1.7 Structure of the Study.....</b>	<b>15</b>
CHAPTER2: Literature Review .....	16
<b>2.1 Introduction .....</b>	<b>16</b>
<b>2.2 Recommendation Systems in the Digital Age .....</b>	<b>16</b>
2.2.1 Early Recommendation Systems .....	17
2.2.2 The Rise of Collaborative Filtering .....	17
2.2.3 Content-Based Recommendation Systems .....	18
2.2.4 Hybrid Filtering: .....	23
<b>2.3 The Cold Start Problem in Movie Recommendation: .....</b>	<b>28</b>

<b>2.4 Leveraging Social media data for Movie recommendations:</b>	<b>30</b>
<b>2.5 Feature engineering and Selection in content based recommendation:</b>	<b>31</b>
2.5.1 Feature Engineering:	31
2.5.2 Feature Selection	32
<b>2.6 Evaluation Metrics for Recommendation Systems:</b>	<b>33</b>
<b>2.7 Comparison of Techniques:</b>	<b>34</b>
<b>2.8 Discussion:</b>	<b>38</b>
<b>2.9 Summary:</b>	<b>39</b>
<b>CHAPTER 3 RESEARCH METHODOLOGY</b>	<b>40</b>
<b>3.1 Introduction:</b>	<b>40</b>
<b>3.2 Methodology</b>	<b>40</b>
3.2.1 Data Collection:	41
3.2.2 Data Pre-processing	41
3.2.3 Feature Selection:	42
3.2.4 Feature Engineering:	43
3.2.5 Model Development	44
3.2.6 Model Evaluation	45
3.2.7 Website Development:	45
<b>3.3 Content-based filtering</b>	<b>47</b>
<b>3.4 Tools:</b>	<b>48</b>
3.4.1 Python (version 3.9.7)	48
3.4.2 Pandas (version 1.4.3)	48
3.4.3 NLTK (version 3.8.1)	49
3.4.4 Django (version 4.2.7)	49
3.4.5 Jupyter Notebook/Google colab	49
3.4.6 Hardware Requirement	50
<b>3.5 Summary</b>	<b>50</b>

CHAPTER 4: ANALYSIS & IMPLEMENTATION.....	51
<b>4.1 Introduction:.....</b>	<b>51</b>
<b>4.2 Dataset:.....</b>	<b>51</b>
<b>4.3 Data Preparation .....</b>	<b>51</b>
4.3.1 Handling Categorical Features in JSON Format .....	52
4.3.2 Identifying and Eliminating Missing Values:.....	53
4.3.3 Feature Engineering: Enhancing Textual Insights .....	53
<b>4.4 Exploratory Data Analysis .....</b>	<b>54</b>
4.4.1 Genres & Keywords .....	54
4.4.2 Popularity & Ratings .....	57
4.4.3 Weighted Rating .....	58
4.4.4 Languages .....	59
4.4.5 Top Grossed movies .....	60
4.4.6 Top Rated Movies .....	60
<b>4.5 Exploratory Data Analysis (Bivariate analysis) .....</b>	<b>61</b>
4.5.1 Director Vs Budgets& Revenue .....	61
4.5.2 Director vs Ratings .....	62
4.5.3 Actor vs Budget & Revenue .....	63
4.5.4 Actor vs Ratings: .....	64
4.5.5 Genres vs Popularity.....	65
<b>4.6 Chi-Square Test on Categorical columns .....</b>	<b>66</b>
4.6.1 Hypothesis Testing .....	66
4.6.2 Data Preparation .....	67
<b>4.7 Pearson Correlation Coefficient .....</b>	<b>68</b>
4.7.1 Hypothesis Testing .....	68
<b>4.8 Data Pre-processing:.....</b>	<b>68</b>
<b>4.9 Experiment 3: Feature Combinations .....</b>	<b>69</b>

<b>4.10 Fine tuning Movie Recommendations by assigning weights to columns.....</b>	<b>71</b>
<b>4.11 Implementation of TF-IDF Vectorization and Cosine Similarity.....</b>	<b>71</b>
<b>4.12 Deployment of Django-Based OTT Application on AWS .....</b>	<b>72</b>
4.12.1 Django Application: .....	72
4.12.2 Integration of Content-Based Recommendation System: .....	73
4.12.3 Homepage and Overview page with Recommendations:.....	74
4.12.4 Deploying OTT Application in AWS.....	74
<b>4.13 Summary.....</b>	<b>75</b>
<b>CHAPTER 5: RESULTS AND DISCUSSIONS .....</b>	<b>76</b>
<b>5.1 Introduction .....</b>	<b>76</b>
<b>5.2 Insights from Chi-square test.....</b>	<b>76</b>
<b>5.3 Insights from Pearson Correlation test .....</b>	<b>77</b>
<b>5.4 Evaluating Recommendations.....</b>	<b>77</b>
<b>5.5 Evaluating tags with Added weights.....</b>	<b>82</b>
<b>5.6 Interpretations from EDA .....</b>	<b>85</b>
<b>5.8 Summary.....</b>	<b>89</b>
<b>Chapter 6: Conclusions and Recommendations .....</b>	<b>90</b>
<b>6.1 Introduction .....</b>	<b>90</b>
<b>6.2 Discussion and Conclusion .....</b>	<b>90</b>
6.2.1 Feature Selection Insights:.....	90
6.2.2 Comparative Analysis of Feature Combinations:.....	90
6.2.3 Weighted Tags .....	90
6.2.4 Role of EDA .....	90
6.2.5 Real-world Applicability and Validation: .....	91
<b>6.3 Contribution to knowledge.....</b>	<b>91</b>
<b>6.4 Future Recommendations.....</b>	<b>91</b>
<b>REFERENCES .....</b>	<b>92</b>

APPENDIX A: RESEARCH PROPOSAL .....	95
-------------------------------------	----

## List of Figures

<b>Figure 2.1: Content Based Recommendation system.....</b>	<b>19</b>
<b>Figure 3.1 : Research Methodology .....</b>	<b>41</b>
Figure 4.1 Screenshot of Jason format of a feature column keywords .....	52
Figure 4.2 Transformed Columns of the "Keywords" Feature .....	53
Figure 4.3 Top Genres in Movies .....	54
Figure 4.4 Number of genres per Movie.....	55
Figure 4.5 Top Keywords Most number of movies have .....	56
Figure 4.6 Boxplot of 'Number of Keywords' .....	56
Figure 4.7 Boxplot of Popularity .....	57
Figure 4.8 Distribution of Ratings .....	58
Figure 4.9 Boxplot of Weighted ratings .....	59
Figure 4.10 Pie - Chart of Languages .....	60
Figure 4.11 Top 10 Most popular Movies .....	60
Figure 4.12 Movies with Most average rating of their Movies .....	61
Figure 4.13 Director with Higher Budgets and Revenues .....	62
Figure 4.14 Directors with Most average ratings for their movies .....	63
Figure 4.15 Actors with higher budgets and revenues.....	64
Figure 4.16 Actors with most average rating for their movies .....	65
Figure 4.17 Visualization representing genre vs popularity .....	66
Figure 4.18 contingency table created for genre column.....	67
Figure 4.19 Data Pre-Processing Steps .....	69
Figure 4.20 Tag created by concatenating Categorical and Numerical columns .....	69
Figure 4.21 Steps to generate Recommendations .....	70
Figure 4.22 Steps to Generate recommendations using Weighted tags.....	71
Figure 4.23 Steps to Integrate of Content-Based Recommendation System .....	73
Figure 5.1 Search Bar with Dropdown suggestions.....	86
Figure 5.2 Options Displaying Top Genres in the Homepage.....	86
Figure 5.3 Display of top 5 Popular, Top grossing and highly rated movies .....	87
Figure 5.4 Figure showing overview of Avatar movie .....	87
Figure 5.5 Recommendation for the selection of a movie Avatar .....	88
Figure 5.6 Recommendations of a selection of Movie Spider-Man 3 .....	88

## List of Tables



Table 5.1 Results from Chi-Square test .....	76
Table 5.2 Results from Pearson Correlation Test .....	77
Table 5.3 Recommending Movies for avatar.....	78
Table 5.4 Recommended Movies & Similarity score for the Movie Spider-Man 3.....	79
Table 5.5 Recommended Movies for the Movie Pirates of the Caribbean: At World's End...	80
Table 5.6 Recommended Movies for the Movie Tangled .....	81
Table 5.7 Tag1 with added weights for the movie Avatar.....	82
Table 5.8 Tag1 with added weights for the movie Spider-Man 3 .....	83

## **CHAPTER 1: INTRODUCTION**

### **1.1 Background of the Study:**

In the digital age, recommendation systems have become integral to our online experiences, guiding users in their interactions with a vast array of digital content, products, and services. These systems are particularly prominent in the domain of digital video libraries and Over-The-Top (OTT) platforms, such as Netflix, Amazon Prime Video, and Hulu. These platforms have mastered the art of delivering well-tailored recommendations to their audiences, drawing from the extensive data they accumulate on user behaviours and preferences (Kumaar et al., 2022).

Traditional Movie recommendation systems have heavily relied on rich user profile data and item ratings to fine-tune their suggestions. By analysing user interactions and preferences, they provide users with personalized content recommendations that match their tastes (Kannikaklang et al., 2022). However, a significant challenge arises when new organizations venture into the highly competitive movie industry. These newcomers often find themselves without the wealth of historical user interactions and data that established companies possess.

This predicament highlights the cold start problem, a significant hurdle for organizations with limited user profiles and item ratings (Pujahari and Sisodia, 2022). The cold start problem presents a formidable challenge for delivering accurate and personalized recommendations. In the absence of substantial user interaction data, the conventional collaborative filtering and user-item interaction models used by established platforms may prove ineffective. This issue underscores the need for innovative solutions that can cater to organizations with limited initial user engagement data.

To overcome this challenge, we will harness insights obtained from Exploratory Data Analysis to inform the design of our homepage. Additionally, we will implement the True Bayesian Estimate (TBE) to refine movie ratings. This approach, popularly employed by platforms like IMDB, is instrumental in rectifying ratings for movies with fewer votes that might be incorrectly perceived as highly rated. TBE serves as a crucial strategy for new entrants in the movie industry, enabling them to provide accurate suggestions to users right from the inception of their engagement with the platform.

However, an effective recommendation system should not be limited to addressing the cold start problem alone. It should also explore the dynamics of feature combinations and their influence on recommendation accuracy (Singla et al., 2020). Feature combinations encompass a wide range of attributes related to movies, including genres, directors, actors, keywords, and plot overviews. These attributes are the building blocks of content-based recommendation systems, and their judicious combination can significantly impact the quality of recommendations. For feature selection we have decided to use Feature selection techniques like chi-square or Pearson correlation test

The utilization of advanced feature selection techniques in our research introduces a distinct element of novelty. Rather than relying on conventional methods, we embrace sophisticated approaches to enhance the efficacy of our content-based recommendation system. By incorporating cutting-edge feature selection techniques, we aim to elevate the accuracy and relevance of our recommendations, setting our research apart and contributing a novel dimension to the field. This innovative aspect reinforces our commitment to providing state-of-the-art solutions for organizations entering the movie industry, addressing not only the cold start problem but also emphasizing the importance of optimizing feature combinations for superior recommendation outcomes

## **1.2 Problem Statement**

The objective of this research is to design and implement a robust content-based movie recommendation system tailored for newly established organizations. The primary challenge to be addressed is the initial "cold start" problem, where the system lacks sufficient user data for personalized recommendations. To mitigate this, the system will initially provide recommendations based on the insights derived from EDA.

This study aims to investigate the impact of various feature combinations on recommendation similarity within the content-based recommendation approach. By exploring different feature combinations and evaluating their influence on similarity, the research seeks to optimize the recommendation system. Specifically, feature selection techniques like Chisquare and Pearson coefficient will be explored to enhance the precision of the recommendation algorithm.

Furthermore, the optimized movie recommendation system will be seamlessly integrated into a website. This integration will facilitate an enhanced movie-watching experience for users by presenting popular movie recommendations at the outset. As users interact with the platform

and provide preferences, the system will progressively evolve to deliver personalized movie recommendations, thereby improving user satisfaction and engagement on the website.

### **1.3 Aim and Objective**

The primary aim of this research is to develop a dynamic movie recommendation website that enhances user engagement by initially presenting popular movie recommendations and subsequently tailoring recommendations based on users' preferences. The system will utilize content-based filtering to achieve this, enhancing the overall movie-watching experience for user

The research objectives have been developed in alignment with the aim of this study and include the following:

- Conduct exploratory data analysis (EDA) to gain insights into the structure, patterns, and potential issues within the movie dataset.
- Identify and handle missing values in the dataset. Options include removing rows with missing data, filling missing values with averages or medians, or using more advanced imputation techniques.
- Check for and remove duplicate entries in the dataset.
- Ensure consistency in data formats. For example, standardize date formats, convert categorical variables to a consistent format, and address any inconsistencies in numerical representations.
- Create new features that might enhance the model's predictive power. For a movie dataset, this could involve extracting features like release year from the date, or creating binary flags for specific genres.
- Develop a recommendation strategy to provide users with popular movie recommendations using Cosine Similarity, informed by EDA insights.
- Explore feature engineering techniques, guided by EDA findings, to optimize the utilization of dataset features for improved recommendation accuracy.
- Evaluate and compare the accuracy of various recommendation models by utilizing combinations of different tags within the pre-processed and explored dataset.
- Validate the models using Cosine similarity or other appropriate techniques for feature selection based on the Similarity scores.
- Optimize the selected recommendation model with weighted tags which also serves as a fine tuned model.
- Deploy the optimized recommendation model on the AWS platform, utilizing either the Django or Flask framework for seamless integration into the website.
- Implement a user-friendly interface on the website that enables users to easily interact with the recommendation system using the insights from EDA.

## **1.4 Research Questions**

1. How does the selection and combination of different features from the dataset influence the accuracy of a content-based movie recommendation system?
2. How can the optimized system be effectively deployed on a website?
3. To what extent do features selected from Chi-square and Pearson correlation tests contribute to accurate movie recommendations when tags are created using those features?
4. How effectively do insights gained from Exploratory Data Analysis (EDA) aid in structuring and designing the homepage of an Over-The-Top (OTT) website?

## **1.5 Scope of the Study**

This Research focuses on the development, evaluation and deployment of an optimized content based movie recommendation system for new organizations entering the movie industry. Scope includes below aspects:

### **1.5.1 Cold Start Problem:**

The study addresses the initial cold start problem faced by new organizations with limited user profiles and item ratings (Deldjoo et al., 2019). It concentrates on Popular movie recommendations from dataset using True Bayesian Estimate which is popularly being used in websites like IMDB for the users who have not yet interacted extensively with the system yet. Study aimed to leverage insights from EDA to overcome cold start problem.

### **1.5.2 Feature Combinations:**

The research emphasizes various feature combinations, including movie attributes such as genres, directors, actors, keywords, plot overview etc. The scope includes investigating how different combinations influence the quality and accuracy of recommendations. This study also explores usage of Various Feature selection techniques that are commonly being used on text datasets for CBRS.

### **1.5.3 Model Development and Optimization:**

The study involves designing and implementing multiple content-based recommendation models, each utilizing a distinct feature combination. The scope

extends to optimizing these models based on evaluation metrics to achieve higher recommendation accuracy.

#### **1.5.4 Website Integration:**

The research includes the development of a website using Django/Flask interface that enables users to input their preferences and receive personalized movie recommendations. The study does not emphasize a robust website as this is just a prototype.

#### **1.5.5 Deployment:**

The study culminates in the deployment of the optimized content-based recommendation system on a production environment accessible to users. The scope includes ensuring system functionality and performance on the website.

#### **1.5.6 Limitations:**

The scope acknowledges that while content-based filtering is effective, hybrid methods that combine content-based and collaborative filtering are not within the primary focus of this study. Additionally, while feature combinations are explored, the study does not delve into complex deep learning architectures.

#### **1.5.7 Dataset:**

The study uses a tmdb dataset from kaggle containing movie attributes such as genres, directors, actors, and textual descriptions. The scope does not involve data collection or creation of new datasets.

#### **1.5.8 Evaluation Metrics:**

The study employs standard evaluation metrics such as precision, recall measure recommendation system accuracy. It does not introduce new evaluation metrics.

#### **1.5.9 Statistical Significance:**

While the research aims to provide insights and practical solutions, it may not comprehensively cover statistical significance analysis due to time constraints.

### **1.6 Significance of the Study**

- Our Study Addresses the "cold start" problem, which is a significant hurdle for new organizations lacking extensive user profiles and item ratings using insights from EDA.
- Research provides a roadmap for new organizations to establish a strong foothold in the competitive movie industry. By leveraging content-based recommendation strategies and analysing feature combinations, these organizations can offer relevant and personalized content to users from the outset.
- Personalized recommendations enhance the user experience by helping users discover content aligned with their preferences. This study contributes to creating a more engaging and satisfying movie-watching experience for users, thereby increasing customer retention and loyalty.
- While collaborative filtering and hybrid methods are common, focus on content-based filtering and the examination of different feature combinations adds novelty to the approach. This can lead to innovative techniques that resonate particularly well with new organizations' resource constraints.
- Guiding new organizations in selecting meaningful features from their dataset to enhance recommendation accuracy. This is particularly valuable in domains with limited initial data, as it streamlines the decision-making process.
- The deployment of the optimized recommendation system on a website demonstrates the practical implementation of this research. It serves as a real-world application that showcases the feasibility of this approach and its potential impact on user engagement.
- Exploration of different feature combinations, model development, and optimization contributes to the academic understanding of content-based recommendation systems. Findings from this study can enrich the body of knowledge on recommendation strategies tailored to specific scenarios.
- The methodology can be adapted and extended to other domains beyond movies. Organizations facing similar challenges in various industries can draw insights from this study to develop personalized recommendation systems.

## **1.7 Structure of the Study**

Chapter 1 provides the Back ground of the study and the Aim of Objectives of this research work. This chapter also discussed the Research questions that needs to be answered along on the journey and significance of this study.

Chapter 2 mentions the Related studies related to recommendation systems and their approach to solve a problem. This chapter also discusses the gaps in other studies related to content based recommendation systems. Chapter 3 gives a detailed walkthrough of the methodology followed during the experimentation stage.

Chapter 4 discusses the Experiments performed for generating content based Recommendations. This chapter also provides insights from EDA and also different feature selection techniques being employed for selecting features that help creating tags. Chapter 5 provides the results yielded from the experiments from chapter 4 and Finally Chapter 5 concludes the work on the thesis and discussed future contribution

## **CHAPTER2: Literature Review**

### **2.1 Introduction**

The movie is one of the integral components of our everyday entertainment. The worldwide movie industry is one of the most growing and significant industries and seizing the attention of people of all ages. In the world of OTTs, importance of recommending movies has rapidly increased. This Section introduces different Recommendation systems and previous works done in the fields of various recommendation systems

### **2.2 Recommendation Systems in the Digital Age**

The digital age has witnessed an unprecedented proliferation of data and content across various online platforms. As the volume of available data continues to grow, the need for effective information filtering and personalized recommendation systems becomes paramount. Recommendation systems have emerged as essential tools to assist users in navigating this vast sea of information and aiding in decision-making processes.

In this digital era, recommendation systems have evolved dramatically in response to the changing landscape of user behaviours, preferences, and technological advancements. These systems primarily aim to predict user preferences and recommend items that align with those preferences, ultimately enhancing user satisfaction and engagement. Understanding the trajectory of recommendation systems involves tracing their evolution through distinct phases:



### **2.2.1 Early Recommendation Systems**

The early recommendation systems were rudimentary and often rule-based. They relied on simplistic algorithms, basic matching techniques, and demographic information to provide recommendations. A fundamental approach during this phase was collaborative filtering based on user-item interactions. The initial phase of recommender systems, spanning the majority of the 1990s, was dedicated to the development of methodologies aimed at tackling the issue of information overload. Their primary objective was to deliver specific services to users, striving to alleviate the overwhelming abundance of information (Konstan and Terveen, 2021). The surge of digital information in the 1990s brought about a dire need for efficient information management. The pioneering recommender systems of this era emerged as a response to this challenge. Their fundamental goal was to assist users in navigating the vast sea of data by offering personalized recommendations. By focusing on techniques like collaborative filtering and content-based filtering, these systems aimed to alleviate information overload by presenting users with suggestions tailored to their preferences and behaviours.

### **2.2.2 The Rise of Collaborative Filtering**

Collaborative recommendation systems are a foundational paradigm in the realm of recommendation technology. They are designed to assist users in discovering new items or content based on the preferences and behaviours of a community of users. The core idea is to identify similarities among users or items, recommending items to a user based on the preferences of other users with similar tastes. Collaborative filtering (CF) methods produce recommendations based on usage patterns without the need of exogenous information about items or users (Koren et al., 2021).

#### **2.2.2.1 Key Components of Collaborative Recommendation Systems:**

##### **User-User Collaborative Filtering:**

- In this approach, recommendations are made to a user based on the preferences and behaviours of other users who are similar to them.
- Algorithms analyse user-item interaction data to determine these similarities, often employing techniques such as cosine similarity or Pearson correlation.
- Recommendations are generated by identifying items liked or used by similar users that the current user has not interacted with.

##### **Item-Item Collaborative Filtering:**

- This approach focuses on recommending items that are similar to the ones a user has shown interest in.
- Algorithms determine item similarities based on user interactions, suggesting items that are comparable to those the user has previously engaged with (Xue et al., 2019).
- Item-item collaborative filtering often requires pre-computed item similarities, commonly using methods like cosine similarity or Pearson correlation (Xin et al., 2019).

#### **2.2.2.2 Advantages of Collaborative Recommendation Systems:**

- Collaborative filtering prioritizes users' preferences and behaviours, delivering recommendations that align with individual tastes
- By leveraging the collective wisdom of a user community, collaborative systems introduce users to items they might not have discovered otherwise.
- Collaborative systems can adapt to evolving user preferences, ensuring recommendations remain relevant over time.

#### **2.2.2.3 Challenges and Considerations (Mohamed et al., 2019):**

**Cold Start Problem:** Collaborative filtering struggles when a new user or item enters the system, as there is insufficient interaction data to make accurate recommendations.

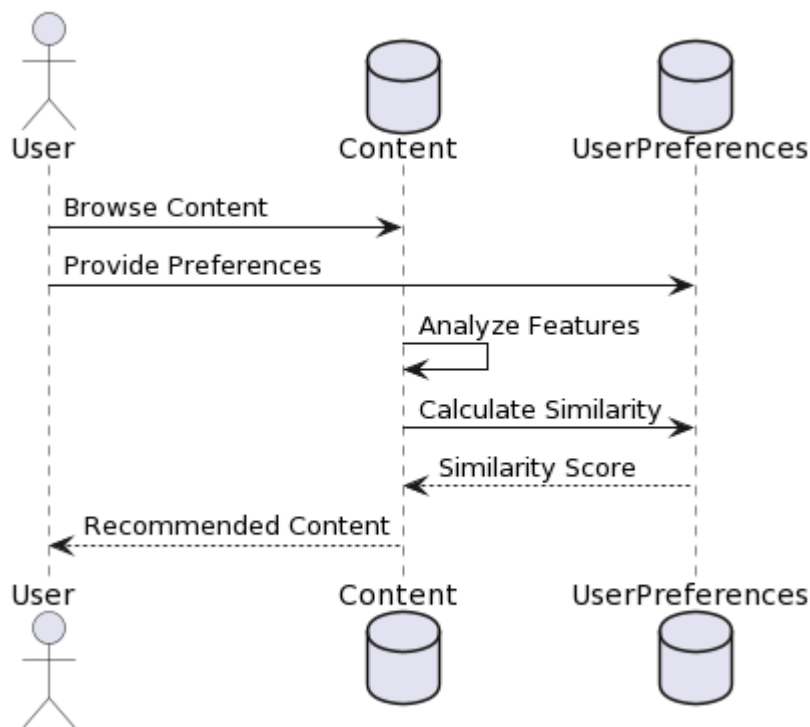
**Sparsity:** Sparse data, where users have interacted with only a small fraction of the available items, can hinder the accuracy and effectiveness of collaborative recommendations.

**Privacy and Trust:** Addressing privacy concerns and establishing user trust is critical when utilizing user data to create recommendations in collaborative systems.

Collaborative recommendation systems, despite their challenges, continue to be a fundamental tool in various domains, driving personalized user experiences and enhancing engagement with diverse content. Advances in machine learning and data analysis have led to more sophisticated collaborative filtering techniques, further improving the quality and relevance of recommendations.

### **2.2.3 Content-Based Recommendation Systems**

Content-based recommendation systems form a fundamental approach in the field of recommender systems. These systems focus on understanding and analyzing the intrinsic characteristics of items and users to provide tailored recommendations. By employing various machine learning and natural language processing techniques, content-based recommendation systems strive to suggest items that align with a user's preferences and past interactions (Lops et al., 2019). Figure 2.1 explains how movies are recommended using CBRS.



**Figure 2.1: Content Based Recommendation system**

### **2.2.3.1 Core Elements of Content-Based Recommendation Systems:**

Content-based recommendation systems are designed to deliver personalized recommendations by analysing item characteristics and user profiles. These systems rely on a

set of core elements that drive their functionality. Here are the essential components of content-based recommendation systems:

### **Feature Extraction:**

Content-based systems start by extracting features from the items to be recommended. These features can include textual attributes (e.g., keywords, descriptions), categorical data (e.g., genre, director, actors), and numerical attributes (e.g., ratings, release year).

### **User Profiling:**

A user profile is constructed based on the user's historical interactions and preferences. This profile is created using the features extracted from items the user has liked, rated, or interacted with.

### **Item Profiling:**

Each item in the system is characterized by a set of features. These features represent the inherent attributes of the item, such as its genre, plot keywords, or cast. Item profiling involves associating relevant features with each item in the catalogue. We will refer the feature created by concatenating these features as tag

### **Content-User Matching:**

The recommendation process hinges on the matching of user profiles with item profiles. Content-based systems assess the similarity between the features in a user's profile and those in the item profiles. The higher the similarity, the more likely the item is to be recommended to the user.

### **Recommendation Generation:**

Recommendations are generated by identifying items with features closely resembling the user's profile. These items are suggested as they are expected to align with the user's preferences. Common similarity metrics include cosine similarity, Jaccard similarity, or Euclidean distance.

### **Content-Based Filtering:**

Content-based filtering is the core recommendation technique used in these systems. It involves filtering items based on their features, aligning them with the user's preferences.

### **Profile Adaptation:**

Over time, user profiles are adapted to reflect changing preferences and behaviors. Content-based systems employ machine learning to continuously update and fine-tune user profiles.

### **Feature Engineering:**

Careful selection and engineering of item features play a vital role in the quality of recommendations. Expertly designed features are essential to ensure the system captures relevant item characteristics.

### **Text Analysis and Natural Language Processing (NLP):**

When dealing with textual attributes, such as movie descriptions or user reviews, content-based systems often employ NLP techniques to extract meaningful information and sentiment analysis for user profiles.

### **Scalability and Efficiency:**

Content-based systems must be designed to handle large datasets efficiently. Techniques like dimensionality reduction (e.g., using techniques like TF-IDF or word embeddings) (Dessi et al., 2021) may be employed to improve scalability.

These core elements work together to enable content-based recommendation systems to provide personalized and relevant suggestions to users. By considering both the features of items and the preferences of users, these systems enhance the user experience and promote engagement with the recommended content.

#### **2.2.3.2 Advantages of Content-Based Recommendation Systems:**

Content-based recommendation systems offer several advantages that make them valuable in various applications. These advantages include:

**Reduced Dependency on User Data:** Content-based systems are less reliant on extensive user interaction data. They can provide recommendations to new users with limited historical data, effectively mitigating the cold start problem.

**Personalization:** These systems offer highly personalized recommendations by considering the specific preferences and behaviours of individual users. The recommendations are tailored to align closely with each user's tastes.

**Transparency:** Content-based recommendations are often more interpretable and transparent to users. Recommendations are based on the features and characteristics of items, making it easier for users to understand why a particular item is suggested.

**Diverse Recommendations:** Content-based systems can suggest a wide array of items based on the varied features considered. This diversity can enhance user satisfaction and engagement by introducing them to a broader range of content.

**No Privacy Concerns:** Content-based systems do not typically require personal user information or data from other users. As a result, privacy concerns associated with collaborative filtering methods are alleviated.

**Scalability:** Content-based systems can be more scalable, especially when dealing with a large number of items, as the primary focus is on item features rather than user interactions.

**Handling Niche Interests:** Content-based systems are effective at recommending niche or specialized items that may have limited user interactions but match the specific features that a user prefers.

**Reduced Sparsity Impact:** These systems can function well even when user-item interaction data is sparse. The primary reliance on item features can help overcome data sparsity issues.

**Continuous Learning:** Content-based systems can adapt to changes in user preferences over time. As users interact with new items, their profiles can be updated to reflect evolving interests.

**Item Explanation:** Content-based recommendations often come with clear explanations based on item features. Users can understand why a particular recommendation is made, leading to increased trust and satisfaction.

These advantages make content-based recommendation systems a valuable tool in various domains, including e-commerce, content streaming, and news article recommendations. While they have some limitations, such as potential over-specialization and the need for effective feature engineering, content-based systems remain an essential part of the recommendation

landscape, often used in combination with collaborative and hybrid approaches for a more comprehensive recommendation strategy.

### **2.2.3.3 Challenges and Considerations:**

**Over-Specialization:** Content-based systems can sometimes recommend items too similar to the user's previous choices, limiting exposure to diverse content.

**Feature Engineering Complexity:** Designing and selecting relevant features for items can be a complex task, impacting the quality of recommendations.

**New and Trending Items:** Content-based systems might struggle to recommend newly released or trending items, especially when user interactions are limited.

Content-based recommendation systems, despite their challenges, play a pivotal role in personalized recommendations across multiple domains. Advances in natural language processing and machine learning continue to enhance these systems, enabling more precise and effective content-based recommendations. Integrating content-based approaches with collaborative methods often leads to hybrid recommendation systems, leveraging the strengths of both paradigms to provide robust and accurate suggestions to users.

### **2.2.4 Hybrid Filtering:**

Hybrid recommendation systems are a sophisticated class of recommender systems that merge multiple recommendation techniques to improve the quality and accuracy of suggestions (Walek and Fojtik, 2020). By combining the strengths of different recommendation paradigms, such as collaborative filtering, content-based filtering, and more, hybrid systems aim to deliver more precise and personalized recommendations, mitigating the limitations of individual approaches (Bahl et al., 2020).

#### **2.2.4.1 Core Principles of Hybrid Recommendation Systems:**

Hybrid recommendation systems are built on a set of fundamental principles that guide their design and operation. These principles form the foundation for combining multiple recommendation techniques, ensuring that the resulting system provides accurate, personalized, and high-quality suggestions. Here are the core principles of hybrid recommendation systems:

### **Diverse Recommendation Techniques:**

Hybrid systems integrate various recommendation techniques, such as collaborative filtering, content-based filtering, knowledge-based filtering, and more. These techniques offer different ways to analyse user behaviour and item characteristics.

### **Customizable Weighting:**

Hybrid systems assign weights to each recommendation technique, allowing for customization. These weights reflect the influence or effectiveness of each technique in generating recommendations. Weights can be adjusted to adapt to specific user profiles or contexts.

### **Seamless Integration:**

Hybrid systems seamlessly integrate different recommendation methods, ensuring that they work cohesively to produce a unified set of recommendations. The integration process aims to minimize data fragmentation and inconsistencies.

### **Context Awareness:**

Hybrid systems are designed to be context-aware. They consider user contexts, such as the user's history, preferences, and interactions, as well as item characteristics, including genre, popularity, and attributes, to tailor recommendations to the specific situation.

### **Enhanced Recommendation Quality:**

The primary goal of hybrid systems is to improve recommendation quality. By combining multiple techniques, they aim to provide more accurate and relevant suggestions, enhancing the user experience.

### **Addressing Limitations:**

Hybrid systems are capable of mitigating the limitations of individual recommendation methods. For example, they can overcome the cold start problem, sparsity, and limited item coverage, which often challenge single-method recommendation systems.

### **User-Centric Approach:**

Hybrid systems prioritize the user's preferences, aiming to deliver recommendations that align closely with individual tastes and behaviours. The user is at the center of the recommendation process.



### **Continuous Learning:**

Hybrid systems often employ machine learning and data analysis to continuously learn from user interactions and adapt to changing preferences. They evolve over time to offer up-to-date recommendations.

These core principles guide the development and operation of hybrid recommendation systems, enabling them to provide users with personalized, context-aware, and high-quality recommendations. The combination of various techniques and the adaptability of these systems contribute to their effectiveness in enhancing user engagement and satisfaction.

### **2.2.4.2 Types of Hybrid Recommendation Systems:**

#### **Weighted Hybrid Systems:**

Weighted hybrid recommendation (Walek and Fojtik, 2020) systems are a sophisticated subset of hybrid recommendation systems. These systems expertly combine multiple recommendation techniques by assigning different weights to each method, resulting in tailored, high-quality recommendations. By intelligently determining the influence of each technique on the final recommendation, weighted hybrid systems offer an adaptable and accurate recommendation solution.

Weighted hybrid recommendation systems provide a powerful means of fine-tuning recommendation strategies. By effectively managing the influence of each technique, these systems strike a balance between personalization and system efficiency. Continuous monitoring and refinement of the assigned weights are essential to maintaining recommendation quality and user satisfaction.

#### **Switching Hybrid Systems:**

Switching hybrid recommendation systems represent a versatile approach within the domain of hybrid recommender systems. These systems are designed to adapt to different user scenarios by selecting the most appropriate recommendation method based on user behavior, item characteristics, or other relevant factors. By intelligently switching between recommendation techniques, switching hybrid systems aim to optimize the quality and personalization of recommendations.

Switching hybrid recommendation systems offer an adaptive and context-aware approach to recommendations. By dynamically selecting the most appropriate recommendation method, they strive to deliver highly personalized and effective suggestions. The ongoing evolution of recommendation algorithms and data processing techniques continues to enhance the effectiveness and efficiency of switching hybrid systems, making them a valuable tool for businesses looking to provide personalized and contextually relevant user experiences.

### **Cascade Hybrid Systems:**

Cascade hybrid recommendation systems represent an advanced approach within the realm of hybrid recommendation techniques. These systems take a sequential approach to generating recommendations, where one recommendation method is applied to pre-filter the item set, and another method is subsequently used to refine the recommendations further. By orchestrating multiple techniques in a step-by-step manner, cascade hybrid systems aim to provide high-quality, context-aware recommendations.

Cascade hybrid recommendation systems offer a nuanced approach to recommendation generation. By considering user and item contexts at multiple stages and progressively refining recommendations, these systems aim to provide users with highly relevant, context-aware, and engaging suggestions. Ongoing research and development in the field of recommendation technology continue to refine the effectiveness and efficiency of cascade hybrid systems, making them a valuable tool for businesses seeking to deliver personalized and high-quality user experiences.

#### **2.2.4.3 Advantages of Hybrid Recommendation Systems:**

Hybrid recommendation systems, which combine multiple recommendation techniques, offer several advantages over single-method recommendation systems. These advantages include:

**Improved Recommendation Quality:** By leveraging the strengths of different recommendation methods, hybrid systems often provide more accurate and relevant suggestions. They reduce the impact of individual method limitations, leading to higher recommendation quality.

**Personalization:** Hybrid systems can deliver highly personalized recommendations. They consider both user behaviour and item characteristics, resulting in suggestions that closely align with individual preferences.

**Enhanced Coverage:** Hybrid systems can address the cold start problem, which affects new users and items with limited interaction data. They offer recommendations for both new and existing users and items, ensuring broader coverage.

**Robustness to Data Sparsity:** These systems can handle sparse data effectively. By combining methods, they reduce the impact of data sparsity, ensuring that recommendations remain accurate and relevant even when interaction data is limited.

**Adaptability:** Hybrid systems can adapt to changes in user behavior and preferences over time. They continuously learn from user interactions and evolve their recommendation strategies to provide up-to-date suggestions.

**Context Awareness:** Hybrid systems often consider contextual factors, such as time, location, or user activity, when generating recommendations. This context-awareness leads to more relevant and timely suggestions.

**Diversity in Recommendations:** The combination of different methods can introduce diversity into recommendations. Users are exposed to a broader range of content, preventing recommendation fatigue and encouraging exploration.

**Flexibility:** Hybrid systems are flexible and can be customized to specific business objectives and user needs. Weight adjustments, method selection criteria, and other parameters can be fine-tuned to achieve desired outcomes.

**Enhanced User Experience:** By offering personalized and high-quality recommendations, hybrid systems can improve the overall user experience. Users are more likely to engage with the system and find content that matches their interests.

**Recommendation for Various Use Cases:** Hybrid systems are versatile and can be applied to various domains, including e-commerce, content streaming, news recommendations, and more. They are suitable for a wide range of recommendation scenarios.

These advantages make hybrid recommendation systems a valuable choice for businesses and platforms seeking to provide top-quality, personalized user experiences. By integrating

different recommendation techniques, they offer a comprehensive and adaptable approach to recommendation technology.

#### **2.2.4.4 Challenges and Considerations:**

##### **Complexity:**

Implementing and maintaining hybrid systems can be complex, as it involves integrating and optimizing multiple recommendation methods.

##### **Scalability:**

Ensuring that the hybrid system remains efficient and scalable as the amount of data and users grows can be a challenge.

##### **Data Integration:**

Integrating and harmonizing data from various sources and techniques is vital but can be challenging.

Hybrid recommendation systems represent an advanced and versatile approach to recommendation technology. By unifying various recommendation paradigms, these systems can provide users with recommendations that are not only more accurate but also adaptable to changing preferences and evolving content. Advances in machine learning and data analysis continue to drive the development and enhancement of hybrid recommendation systems, making them a vital tool for businesses seeking to deliver top-quality personalized user experiences.

### **2.3 The Cold Start Problem in Movie Recommendation:**

The "Cold Start Problem" in movie recommendation is a significant challenge that recommendation systems face when dealing with new or less-established movies, as well as new users. This problem occurs because traditional collaborative and content-based recommendation methods rely on historical user-item interactions and data to make personalized suggestions (Deldjoo et al., 2019). When data is limited or non-existent, as is often the case with new items or users, these systems struggle to generate accurate and relevant

recommendations. Here are the key aspects of the Cold Start Problem in movie recommendation:

### **New Movies with Limited Data:**

For recently released or less-established movies, there is often a scarcity of user interactions, such as ratings, reviews, and views. Without this historical data, it becomes challenging to understand the preferences and characteristics of these movies.

### **New Users with Sparse Profiles:**

When a new user joins a movie recommendation platform, they may not have provided any or very little information about their movie preferences. The system lacks insight into their tastes and interests.

### **Inadequate User-Item Overlap:**

Traditional collaborative filtering methods rely on finding users with similar tastes and recommending items that these similar users have enjoyed. In the absence of user-item overlap, it is difficult to identify such similar users or items.

### **Reduced Recommendation Accuracy:**

The Cold Start Problem can result in inaccurate recommendations for both new users and new movies. Users may receive irrelevant or unrelated movie suggestions, leading to a subpar user experience.

### **Missed Discovery Opportunities:**

New users may miss out on discovering movies they would enjoy, while new movies struggle to gain visibility and traction among potential viewers.

To address the Cold Start Problem in movie recommendation, several strategies can be employed:

**Popularity-Based Recommendations:** For new users, the system can initially provide recommendations based on movie popularity or trends. This approach aims to engage users and gather initial interaction data.

**Content-Based Recommendations:** Content-based recommendation methods can be effective for new movies. These methods analyse the attributes and features of movies (e.g.,

genre, director, actors) to suggest films that are similar in content to those the user has expressed interest in.

**Hybrid Systems:** Combining collaborative and content-based techniques in a hybrid system can mitigate the Cold Start Problem. These systems can provide recommendations based on movie content and gradually introduce collaborative recommendations as more user data is collected.

**Incentivize User Feedback:** Encouraging new users to provide explicit feedback through ratings or reviews can help the system understand their preferences more quickly.

**Data Augmentation:** The use of external data sources, such as social media or reviews from other platforms, can supplement the limited data for new movies.

Addressing the Cold Start Problem is crucial for providing a seamless and effective movie recommendation experience, particularly for users exploring new content and for movies seeking wider audiences. Strategies that adapt to the unique challenges of new items and users are essential in building robust recommendation systems in the dynamic world of movie entertainment.

## **2.4 Leveraging Social media data for Movie recommendations:**

Leveraging social media data for recommendations is an innovative approach that can enhance the quality and relevance of suggestions across various domains, including movies, products, news, and more. Social media platforms provide a wealth of user-generated content and interactions, making them a valuable source of information for recommendation systems. There are also some studies which suggests to handle cold start problem using Social media data (Herce-Zelaya et al., 2020). Here are the key considerations when incorporating social media data into recommendation systems:

### **Data Collection:**

To harness social media data for movie recommendations, data needs to be collected from platforms like Twitter, Facebook, Instagram, and others. This data can include user comments, reviews, likes, shares, and posts related to movies.

### **Sentiment Analysis:**

Apply sentiment analysis techniques to social media data to determine user sentiments and opinions about movies (Rahman and Hossen, 2019) . This analysis can help identify whether users have a positive, negative, or neutral sentiment toward a film.

### **Trend Analysis:**

Monitor social media trends and discussions related to movies. This can include identifying trending topics, hashtags, and keywords associated with popular or upcoming films.

## **2.5 Feature engineering and Selection in content based recommendation:**

The cornerstone of content-based recommendation systems lies in the astute engineering of features and the judicious selection of these attributes. This section delves into the intricate processes of feature engineering and selection, which are integral to the effectiveness of content-based recommendation systems.

### **2.5.1 Feature Engineering:**

In content-based recommendation systems, the art of feature engineering involves the creation of meaningful attributes that vividly represent the items under consideration. These attributes encapsulate essential information pertaining to the items, ensuring that the recommendation process is grounded in comprehensible features. Some notable aspects of feature engineering include:

**Item Attributes:** The foundation of content-based recommendation systems rests on the identification of item attributes. These attributes, which encompass elements such as genres, directors, actors, and plot keywords, serve as the bedrock of the system's understanding of items.

**Textual Data Processing:** In instances where textual data accompanies items, the art of natural language processing (NLP) (Singh and Singh, 2019) comes into play. Techniques like tokenization, sentiment analysis, keyword extraction, and topic modelling are adeptly applied to extract important features from text.

**Encoding Categorical Data:** Transforming categorical attributes into numerical formats is a critical task. Approaches such as one-hot encoding, label encoding, or embedding layers in deep learning models are applied to represent these attributes numerically.

**Feature Scaling:** To ensure compatibility across features, numerical attributes are often normalized or scaled. Techniques such as z-score normalization or min-max scaling are commonplace in this context.

**Image and Multimedia Data:** For items comprising multimedia content, convolutional neural networks (CNNs) and other deep learning models are harnessed to extract feature representations from images and multimedia elements.

**Content Metadata:** The inclusion of additional metadata, such as user-generated tags and descriptions, further enriches the feature set, enhancing the system's ability to characterize items.

**User Interaction Data:** User interactions, encompassing actions such as item views, likes, shares, and ratings, are ingeniously translated into features that capture user preferences and engagement.

### 2.5.2 Feature Selection

The process of feature selection is equally pivotal in refining content-based recommendation systems. Feature selection is the art of identifying and retaining the most influential attributes, thereby enhancing the precision and relevance of the recommendations. Some noteworthy facets of feature selection are as follows:

**Relevance Assessment:** Each feature is meticulously evaluated for its relevance to the recommendation task. Features that exhibit a substantial influence on the quality of recommendations are preserved.

**Correlation Analysis:** The interplay between features is comprehensively examined, with redundant or highly correlated attributes flagged for potential elimination, effectively mitigating concerns related to dimensionality.

**Feature Importance:** Insight into feature importance is gained through machine learning models, particularly decision trees, which guide the selection of features with the most substantial impact.

**Dimensionality Reduction:** Principal component analysis (PCA) and linear discriminant analysis (LDA) are considered for reducing dimensionality while preserving the critical information (Ayesha et al., 2020).



**Sequential Feature Selection:** Iterative strategies like forward selection, backward elimination, and recursive feature elimination are deployed to discern the optimal feature subset

**User Feedback:** User feedback is a crucial guide in the feature selection process, ensuring the retention of attributes deemed valuable by users.

**Adaptive Feature Selection:** Adaptability is introduced through methods that dynamically update feature sets in response to evolving user preferences.

The process of feature engineering and selection is fundamental in elevating the performance of content-based recommendation systems. These processes empower the system to retain informative features while minimizing noise and mitigating dimensionality concerns, thereby enhancing the user experience.

## 2.6 Evaluation Metrics for Recommendation Systems:

Evaluation metrics play a pivotal role in assessing the performance and effectiveness of recommendation systems. In this section, we delve into the diverse range of evaluation metrics used to gauge the quality of recommendations. These metrics provide a quantitative means to measure how well a recommendation system meets its objectives and how it aligns with user preferences. The choice of evaluation metrics is a critical decision in the design and assessment of recommendation systems. Some of the most commonly used evaluation metrics include:

**Precision:** Precision measures the ratio of relevant items (true positives) in the recommendations to the total number of items recommended. It quantifies how accurately the system selects items that users actually find relevant. High precision indicates that the system rarely suggests irrelevant items .

**Recall:** Recall calculates the ratio of relevant items retrieved in the recommendations to the total number of relevant items in the dataset. It measures the system's ability to find all the relevant items. A high recall means the system is effective in identifying most of the relevant items.

**F1 Score:** The F1 score combines precision and recall into a single metric, providing a balanced measure of recommendation quality. It's particularly useful when precision and recall need to be balanced, as it considers both false positives and false negatives.

**Mean Average Precision (MAP):** MAP considers the average precision at each relevant item's rank and provides a comprehensive measure of recommendation quality. It rewards systems that present relevant items early in the list.

**Root Mean Square Error (RMSE):** RMSE is commonly used for rating prediction tasks. It quantifies the difference between predicted and actual ratings. Lower RMSE values indicate better prediction accuracy.

**Mean Absolute Error (MAE):** Similar to RMSE, MAE measures prediction accuracy by assessing the absolute differences between predicted and actual ratings. It is less sensitive to extreme errors.

### **User Satisfaction:**

Collect feedback from users through surveys, ratings, or other means to understand their satisfaction with the recommendations. User feedback is valuable for capturing subjective preferences.

## **2.7 Comparison of Techniques:**

This project seeks to develop a content-based movie recommendation system tailored for new organizations, addressing the initial cold start problem using Popular movie recommendation from the dataset and investigating the influence of varying feature combinations on recommendation accuracy. The system will then be optimized based on these findings and seamlessly integrated into a website, allowing users to receive popular recommendations initially and later personalized movie recommendations based on his preferences on the website and enhancing their movie-watching experience.

Content based recommendation systems are one of the popular recommendations systems in the field of study. A study (Havolli et al., 2022) used Netflix dataset and suggests Adamic-Adar measures are effective for recommending new items. TF-IDF and Word2Vec models are employed to build a content-based recommendation system.

(Pujahari and Sisodia, 2022) Examines the concept of enhancing item features through matrix factorization. Additionally, it emphasizes that the scarcity and disparities in the datasets pose difficulties in collecting an ample amount of item feature data. It underscores the significance of the system adapting to correct inaccurate recommendations

(Kannikaklang et al., 2022) conducted research on the Movielens database to develop a hybrid recommendation system by integrating collaborative and content-based filtering. Their study incorporated models such as User K-NN, Item K-NN, Matrix Factorization, and Biased Matrix Factorization, comparing their performance using metrics like RMSE and MAE. Their findings indicated that Matrix Factorization, Biased Matrix Factorization, and Factor-wise Matrix Factorization are well-suited for collaborative filtering, while suggesting that Content-based filtering may not be optimal for large datasets.

(Darban and Valipour, 2022) Study leverages graph-based features and autoencoders and proposed a recommendation system. The primary objective of their study was to address the cold start problem by establishing relationships between users based on their similarities, represented as nodes in a similarity graph.

(Kumaar et al., 2022) utilized the IMDbPY Python library as their dataset and employed TF-IDF models with cosine similarity, Jaccard recommender models, and word-count cosine similarity models on the 'Keywords' and 'Plot Overview' features separately. Their research concluded that 'Keywords' were the most effective feature, with the Count Vectorizer model providing the best recommendations.

(Pradeep et al., 2020) attempted to merge certain features and employed cosine similarity to offer movie recommendations. They emphasized that their system does not consider other user profiles during the recommendation process.

(Javed et al., 2021) introduced a context-aware recommender system that filters items based on users' interests, in combination with a context-based recommender system for item recommendations. Their study concluded that ontology-based recommendation systems, when combined with other techniques, are widely employed for context-aware resource recommendations

(Albayati and Ortakci, 2022) suggested the application of content-based filtering for marketing purposes on specific social media platforms. Their study employed TF-IDF and cosine similarity for recommendations and found that the system successfully predicted target users for selling and providing services, achieving an accuracy rate of 86.2 based on users' tweets.

(Sahu et al., 2022) utilized IMDb and TMDB data to implement a multiclass classification model with an accuracy rate of 96.8%. Their research employed content-based

recommendations to create a new dataset with predicted ratings and voting information. They presented a multiclass model using deep learning with CNN architecture.

(Rendle et al., 2020) This study revisits the comparison between Neural Collaborative Filtering (NCF) and traditional Matrix Factorization. They analyse the advantages and disadvantages of these two recommendation approaches. This paper provides a critical reevaluation of NCF and Matrix Factorization, offering insights into which method may be more suitable for specific recommendation tasks.

A survey on graph based neural network (Wu et al., 2022) in RS analysed the challenges of applying GNN on different types of data and tried to state new perceptive pertain to the development of the field. This study proposed classification scheme for existing works on GNNs.

In the research conducted by (Ferrari Dacrema et al., 2019) a comprehensive examination of recent neural algorithms for top-n recommendation was carried out. The findings of their analysis unveiled several noteworthy observations. Notably, they reported that reproducing published research in this domain remains a challenging endeavour, signalling a call for more systematic and reproducible research practices. Additionally, their study shed light on the surprising discovery that simpler algorithms, both in terms of conceptual and computational complexity, often outperformed their more intricate counterparts on certain datasets. These results underline the necessity for a more rigorous approach to algorithmic evaluation and a plea for enhanced research practices in the field. Furthermore, their analysis was primarily confined to papers published in specific conference series, emphasizing the need for a broader exploration across diverse publication outlets and recommendation problem types. This study also outlined the intention to incorporate traditional algorithms, particularly those based on matrix factorization, in future research to offer a more comprehensive evaluation of recommendation techniques.

In (Benkessirat et al., 2021) study, a game theory-based recommendation system was presented, introducing a unique approach that integrates cooperative game theory and solution concepts to preselect similar user communities while considering intrinsic user relationships. To ascertain the effectiveness of their algorithm, they conducted a series of well-designed experiments, evaluating their CF-GT algorithm using precision, recall, and MAE metrics. Additionally, they conducted a comparative study to measure the performance of their proposed algorithm against traditional collaborative filtering and k-means-based collaborative

filtering methods. This Study demonstrated a remarkable boost in predictive accuracy with their innovative algorithm, underlining its potential to enhance recommendations. Their unique approach includes precise neighbourhood preselection, differentiating it from conventional methods.

This study (Batmaz et al., 2019) offers a thorough review of deep learning-based recommender systems, with a focus on addressing challenges, exploring applications across recommendation domains, and understanding the driving factors behind recommendations. It also provides a quantitative assessment of relevant publications and suggests future research directions in this evolving field. The Study concludes that deep learning has become a unifying approach across various computer science subfields, fostering collaboration and providing effective solutions to complex problems. The study categorizes the literature into four critical aspects: deep learning models, addressing recommender system challenges, application domains, and the relationship between deep learning techniques and recommendation properties. While deep learning holds promise for recommender systems, challenges like accuracy and scalability remain, leaving room for future research and improvement.

(Pintas et al., 2021) focuses on Feature Selection (FS) methods in text classification, analysing 1376 papers from 2013 to 2020. The study presents a comprehensive categorization schema for these methods and highlights the key aspects of experiments, including datasets, languages, machine learning algorithms, and validation methods. It offers valuable insights for researchers aiming to position their work within the existing literature and ensures a standardized, unbiased classification of included studies. This study emphasizes the complexity and significance of tailored Feature Selection (FS) methods in text classification. The proposed categorization scheme serves as a valuable tool for analysing existing research and guiding new studies. Additionally, the mapping of experiment settings aims to assist in the design of future experiments. One notable future endeavour is the creation of an open platform for testing FS methods specific to text classification.

(Albayati and Ortakci, 2022) focuses on the classification of Arabic documents into predefined categories, a challenging task due to the vast volume of Arabic content on the internet. The research introduces the improved CHI (ImpCHI) Square method and evaluates its impact on six well-known classifiers, including Random Forest, Decision Tree, Naïve Bayes, and more. The dataset comprises 9055 Arabic documents divided into twelve categories, with performance assessed using F-measure, recall, precision, and Time build model. The findings

indicate that ImpCHI square significantly enhances classification results compared to the normal CHI square method across all performance criteria for all classifiers

(Silveira et al., 2019) evaluates the recommender systems, an essential aspect in enhancing user satisfaction. It examines both traditional and contemporary evaluation metrics, including accuracy, Root Mean Square Error, P@N for top-k recommendations, and newer concepts like novelty, diversity, and serendipity. The paper surveys and organizes various research works that define these concepts and offer metrics for recommendation evaluation. It categorizes these metrics based on their objectives and sets the stage for potential future research topics focused on improving user satisfaction.

(Fan et al., 2019) explores the application of Graph Neural Networks (GNNs) to social RS, highlighting their potential to improve user recommendations by effectively integrating user-user social graphs and user-item graphs. The paper introduces a novel GNN framework, GraphRec, designed to address key challenges in social recommendations, such as encoding both user interactions and opinions, handling heterogeneous social relation strengths, and dealing with users participating in multiple graphs. Experimental results on real-world datasets validate the effectiveness of the GraphRec framework in enhancing social recommendations.

## **2.8 Discussion:**

The reviewed studies consistently emphasize the importance of content-based recommendation systems, particularly in domains like movie recommendation. These systems leverage movie features such as genre, cast, director, and plot keywords to generate personalized suggestions for users. Many studies have proposed a Hybrid Recommendation system (Gunawardana and Meek, 2009) as an ideal option to address the cold start problem. They also emphasize the use of evaluation metrics, including accuracy, Root Mean Square Error, and newer concepts like novelty, diversity, and serendipity.

Several studies discussed the challenges in content-based recommendation, especially feature engineering. The nuanced relationships between movie features and user preferences pose significant challenges. Therefore, the accurate representation of movie content is essential for recommendation system performance. Some studies proposed advanced techniques such as deep learning, including convolutional neural networks (Christakou et al., 2007) (CNNs),

GNNs, to handle the complexity of feature combinations. Such approaches demonstrate the potential for substantial improvements in recommendation accuracy.

The nuanced relationships between movie features are less explored in various studies, Furthermore, none of them have explored feature selection techniques such as Chi-Square or Improved Chi-Square. Some studies have rightly incorporated all recommendations and provided a website with all the recommendations.

## **2.9 Summary:**

This chapter discusses the application of various recommendation systems in different industries, with a particular focus on movie recommendations. It provides a general idea of the concept of recommendation systems and their rapid development, especially within the movie industry. The chapter delves into various techniques that have been reviewed, exploring their impact on the user experience and their ability to recommend movies based on individual tastes. It's clear that these recommendation systems have transcended their initial roles and have become integral components of modern online services. Their innate ability to enhance user engagement and satisfaction has not only impacted the movie industry but has also fundamentally reshaped the dynamics of customer interactions. We have also reviewed the use of evaluation metrics for recommendations, also various similarities to generate top N recommendations.

## **CHAPTER 3 RESEARCH METHODOLOGY**

### **3.1 Introduction:**

The primary goal of this research is to conceptualize, design, develop, and evaluate a specialized Content-Based Recommender System (CBRS) tailored for Over-The-Top (OTT) platforms. Achieving this objective necessitates a structured and systematic methodology that encompasses six key stages. The first stage involves the identification and collection of reliable and relevant data sources. In the second stage, data pre-processing is conducted to ensure the data is cleaned and formatted appropriately for analysis.

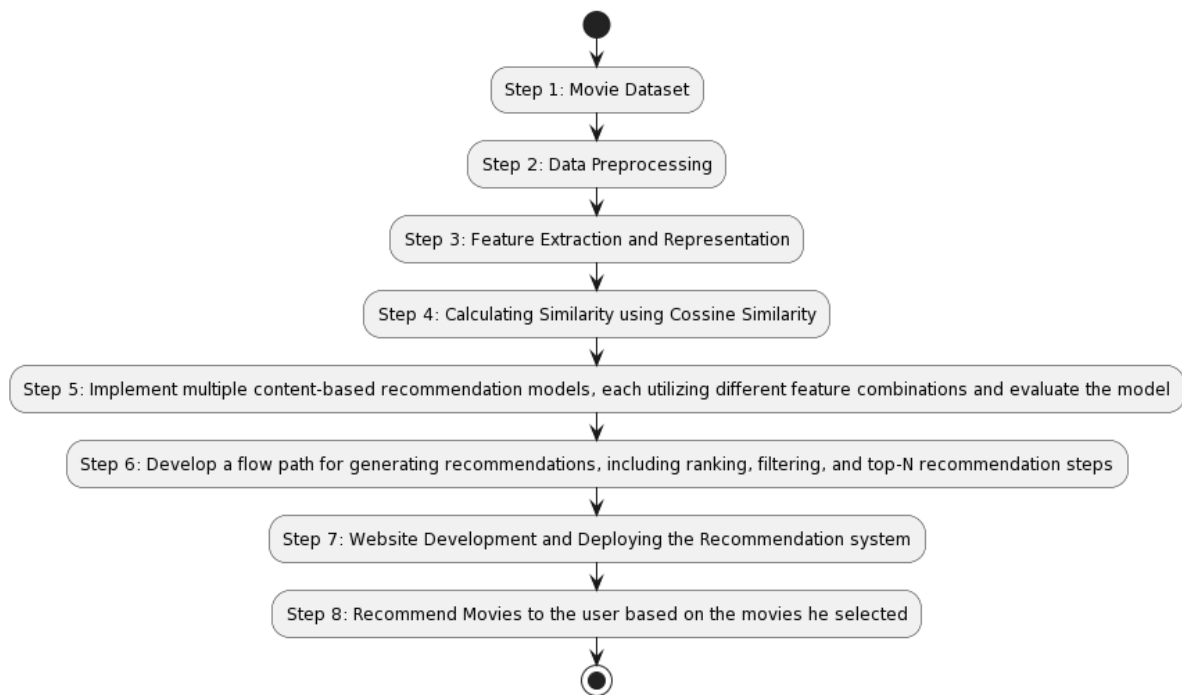
The third stage focuses on feature engineering and employs Natural Language Toolkit (NLTK) techniques to extract meaningful and valuable features from the dataset. In the fourth stage, feature selection techniques such as Chi-square or Imp Chi-Square are applied to refine the dataset and identify the most significant features. The fifth stage revolves around the development of the recommendation model, utilizing techniques like cosine similarity to generate Top-N recommendations. This step plays a pivotal role in ensuring the accuracy and effectiveness of the recommendations provided by the CBRS.

Finally, in the last stage, a user-friendly website will be created to incorporate and deploy the CBRS model. This end-to-end project ensures that users can seamlessly access personalized movie suggestions in alignment with their preferences. The comprehensive approach outlined in this methodology guarantees the robustness and practicality of the CBRS for OTTs. It is the methods and techniques used in this chapter that serve as the building blocks for the CBRS and contribute to the empirical understanding of its performance.

### **3.2 Methodology**

The objective of this CBRS is to design a movie recommendation system for Over-The-Top (OTT) platforms. The scope encompasses a global audience interested in diverse genres and languages. Let us see understand all the stages our methodology detail.





**Figure 3.1 : Research Methodology**

### 3.2.1 Data Collection:

The selected dataset is sourced from Kaggle, specifically the TMDB dataset. This dataset includes essential columns, such as cast, crew, keywords, and plot overviews, necessary for implementing our study. It aims to cover a diverse range of genres, languages, and release years, incorporating both popular and niche movies to ensure broad appeal. To mitigate potential bias, special attention is given to underrepresented genres. The Content-Based Recommendation System (CBRS) is designed to promote serendipity by recommending movies outside users' typical preferences. We utilize the entire TMDB dataset without sampling to ensure a comprehensive analysis. The dataset undergoes a validation process for accuracy and consistency, addressing missing or inconsistent data points to maintain the quality of recommendations. By selecting the TMDB dataset from Kaggle, we ensure a rich source of information to build a CBRS that caters to diverse user preferences.

### 3.2.2 Data Pre-processing

To enhance the dataset for the development of the CBRS tailored for OTT movie recommendations, a meticulous data pre-processing strategy is executed. The process commences with addressing missing data, rectifying inconsistencies, and standardizing formats to ensure data integrity. For text data, a comprehensive approach involves tokenization, removal of stop words, and lemmatization, optimizing it for subsequent feature extraction (Yogish et al., 2019).

Feature engineering is integral, encompassing the creation of binary indicators for popular genres, extraction of features like release years, and encoding categorical variables for numerical compatibility. The dataset undergoes scaling and normalization to establish consistent scales for numerical features, while noise removal, handling duplicates, and imbalanced data treatment contribute to refining data quality. Merging datasets, a crucial step, consolidates information from multiple sources, enriching the dataset.

Integration with collaborative filtering is facilitated through pre-processing steps, and user preferences are mapped to numerical scales. Furthermore, the dataset undergoes splitting for training and testing sets, addressing long-tail items, and implementing privacy measures to ensure ethical handling of user data. Continuous validation and meticulous documentation practices guarantee the dataset's readiness for constructing a robust CBRS, providing accurate and meaningful OTT movie recommendations.

### **3.2.3 Feature Selection:**

The initial step involves employing techniques like Chi-Square or Information Gain to select the most relevant features from the dataset. This ensures that the system focuses on the attributes that significantly contribute to content preferences. Features like genre, cast, director, and keywords may be prioritized based on their impact on user preferences. Main aim of this section is to select relevant features that encapsulate the key aspects that influence a user's content preferences.

#### **3.2.3.1 Chi-Square:**

Chi-square ( $X^2$ ) can be utilized for feature selection, particularly when dealing with categorical variables. The chi-square test of independence is a statistical test that assesses if there is a significant association between two categorical variables (Sayassatov and Cho, 2020). When applied to feature selection, the chi-square test helps identify the features that are most relevant to the target variable.

Here's a general approach to using chi-square for feature selection

**Formulate the Hypotheses:** Null Hypothesis ( $H_0$ ): There is no significant association between the feature and the target variable.

Alternative Hypothesis ( $H_1$ ): There is a significant association between the feature and the target variable.

**Compute the Chi-square Statistic:** For each feature, create a contingency table that shows the distribution of the feature and the target variable.

Calculate the expected frequencies under the assumption that the variables are independent.

Use these values to compute the chi-square statistic.

**Determine Significance:** Compare the computed chi-square statistic to a critical value from the chi-square distribution with the appropriate degrees of freedom.

Alternatively, use a p-value to determine if the association is statistically significant.

**Select Features:** Features with a significant association (rejecting the null hypothesis) are considered important for prediction and can be selected.

### 3.2.3.2 Pearson Correlation Test:

The Pearson correlation coefficient measures the strength and direction of a linear relationship between two continuous variables (Chen et al., 2021). It quantifies how well a straight line can describe the relationship between the variables.

#### Hypothesis Testing:

- Null Hypothesis ( $H_0$ ): There is no significant correlation between the variables.
- Alternative Hypothesis ( $H_1$ ): There is a significant correlation between the variables.

### 3.2.4 Feature Engineering:

Natural Language Processing (NLP) techniques, such as those provided by the Natural Language Toolkit (NLTK), can be applied for feature engineering. This involves extracting meaningful information from textual data, such as movie descriptions. By transforming

unstructured text into structured features, the system gains a deeper understanding of the content.

Furthermore, the integration of user-item interaction data, such as user ratings and viewing history, into the feature set enhances the model's ability to capture personalized preferences. Techniques like cosine similarity may be employed to measure the similarity between user profiles and content features.

Through meticulous feature selection and engineering, the CBRS not only optimizes its ability to understand content but also tailors recommendations to individual user tastes. This ensures a more accurate and personalized content recommendation process.

### 3.2.5 Model Development

The culmination of the preparatory stages leads us to the pivotal phase of model development, where the architectural framework for the Content-Based Recommender System (CBRS) takes shape. Leveraging the processed features and the carefully engineered representations, the system adopts a content-based approach for personalized recommendations on Over-The-Top (OTT) platforms.

The central focus revolves around employing Natural Language Processing (NLP) techniques for feature extraction from textual data. Textual Feature Representation, a critical element, involves transforming the processed text into numerical representations. Two prominent methods utilized are Term Frequency-Inverse Document Frequency (TF-IDF) vectors and Word Embeddings (e.g., Word2Vec, GloVe).

**TF-IDF (Term Frequency-Inverse Document Frequency):** TF-IDF is a well-established technique that captures the importance of a word within a document relative to its frequency across the entire corpus. It assigns weights to words based on their significance, enabling the system to understand the contextual relevance of terms within the dataset. This method proves valuable in capturing the essence of the textual data and enhancing the discriminative power of the features (Singh and Shashi, 2019).

**Word Embeddings (Word2Vec, GloVe):** Word embeddings involve representing words in a continuous vector space, capturing semantic relationships between words. Models like Word2Vec and GloVe are adept at learning distributed representations of words, providing a

nuanced understanding of the contextual semantics. These embeddings allow the system to discern subtle semantic nuances and improve the system's ability to generate meaningful recommendations.

As the textual features are effectively harnessed, the subsequent step involves merging these features with other relevant ones. This fusion of features ensures a holistic understanding of the content, contributing to a comprehensive user-item similarity model. The model, primarily grounded on cosine similarity, facilitates the generation of Top-N recommendations, aligning the system with the user's preferences.

### **3.2.6 Model Evaluation**

The robustness and effectiveness of the developed Content-Based Recommender System (CBRS) are critically assessed through a comprehensive model evaluation process. This phase is indispensable in gauging the system's performance, ensuring that it aligns with the overarching goal of delivering accurate and meaningful recommendations on Over-The-Top (OTT) platforms.

**Top-N Recommendations:** The success of the CBRS hinges on its capacity to generate relevant and engaging recommendations for users. The Top-N recommendation evaluation assesses the system's ability to suggest content that aligns with users' tastes, thereby enhancing user satisfaction and engagement.

**Comparative Analysis:** The CBRS is benchmarked against existing recommendation models to ascertain its comparative advantages. Comparative analysis involves evaluating the proposed model against established benchmarks, ensuring it outperforms or, at the very least, matches the performance of state-of-the-art recommendation systems.

. The model evaluation phase is iterative, involving fine-tuning and optimization based on the feedback and insights gleaned from the evaluation metrics. This iterative approach ensures that the CBRS continually evolves, adapting to dynamic user preferences and content landscapes within OTT platforms.

### **3.2.7 Website Development:**

The final stages of the Content-Based Recommender System (CBRS) implementation involve developing a user-friendly website and seamlessly deploying recommendations into the

platform. These steps ensure that the model's valuable insights are accessible to end-users in a practical and interactive manner.

### 3.2.7.1 Front-End Development

The user interface (UI) is a critical component of the CBRs, shaping the overall user experience. Front-end development focuses on creating an intuitive and visually appealing website that facilitates user interactions. Elements such as an aesthetically pleasing design, easy navigation, and responsive features contribute to an engaging user interface.

To address the cold start problem, we will strategically feature popular movies from the dataset on the home page. This selection will be based on the True Bayesian estimate weighted rating, a metric widely employed in platforms like IMDb

$$\text{Weighted Rating (WV)} = (\text{votes} \div (\text{votes} + \text{min votes})) \times R + (\text{min votes} \div (\text{votes} + \text{min votes})) \times C$$

$R$  = Mean of Movie ratings

Votes = Total no. votes for the movie

min votes = The minimum number of votes needed to secure a place in the Top 250 is presently set at 25,000.

$C$  = The average (mean) vote across the entire report

This approach ensures that the home page showcases movies with a balance of popularity and quality, as reflected by user ratings.

Additionally, we will integrate a search option at the top of the home page, empowering users to either type the movie name or select a popular movie from a dropdown menu. This feature serves a dual purpose: allowing users to explore movies of interest and providing the system with initial user preferences to generate recommendations based on the previously developed model.

### 3.2.7.2 Back-End Integration

The back-end development involves integrating the CBRs with the website, enabling seamless communication between the user interface and the recommendation engine. This integration

ensures that user interactions trigger the recommendation algorithm, resulting in real-time and personalized content suggestions.

#### **3.2.7.3 Recommendation Display:**

The website prominently features the recommendations generated by the CBRS. Recommendations may be displayed on the homepage, in specific content categories, or through personalized user dashboards. The goal is to make the recommended content easily accessible and visible to users, encouraging exploration and engagement.

#### **3.2.7.4 Deployment Strategies:**

The deployment of the CBRS into the production environment involves selecting suitable strategies to ensure optimal performance. Cloud-based deployment, containerization (e.g., Docker), and micro services architecture are common approaches that enhance scalability, reliability, and maintainability.

### **3.3 Content-based filtering**

Content-based filtering is a recommendation system technique that tailors suggestions to users based on the inherent characteristics of items and the user's historical preferences. In the specific context of a movie recommendation system, this approach revolves around leveraging the distinctive features of movies and the user's past interactions to generate personalized recommendations.

The first step in content-based filtering involves feature extraction. This encompasses mining information from various sources such as movie descriptions, keywords, genres, cast, and director. Additionally, numerical features like release year, duration, and average ratings are considered. This amalgamation of textual and numerical features provides a comprehensive representation of each movie.

To process the textual data effectively, text pre-processing techniques are employed. This involves tokenization and cleaning to ensure that the extracted information is meaningful. The next stage involves calculating TF-IDF scores—a mechanism that captures the significance of terms in the textual data. Concurrently, numerical features are normalized to ensure equitable weight in the recommendation process.

The user profile is a crucial component, reflecting the user's historical preferences based on movies they have rated highly. The cosine similarity between the user profile and the features of each movie is then computed. This quantifies the likeness between the user's preferences and the inherent characteristics of the movies.

Recommendations are generated by ranking movies based on their cosine similarity scores. The top-N movies are selected as personalized recommendations for the user. To tackle the cold start problem, where there might be insufficient user data, popular choices are incorporated and weighted by a True Bayesian estimate.

The integration with the front-end is a pivotal aspect, where a user-friendly interface is implemented to display personalized movie recommendations. Interactive features are included, allowing users to explore recommendations and provide feedback. This two-way interaction enhances the user experience and refines the recommendation process.

A continuous learning mechanism is embedded in the system. Feedback from user interactions is used to update the user profile, ensuring that the recommendations evolve with changing preferences. Periodic retraining of the model is carried out to stay attuned to dynamic user behaviour and industry trends.

In essence, content-based filtering offers a nuanced and personalized recommendation experience, considering both textual content and numerical features. The orchestrated use of TF-IDF, cosine similarity, and continuous learning mechanisms ensures a dynamic and user-centric recommendation system in the realm of movie suggestions.

### **3.4 Tools**

In the pursuit of developing a robust content-based recommendation system for movies, a carefully selected set of tools has been chosen to empower various stages of the research process. Each tool serves a specific purpose, contributing to the efficiency, accuracy, and user-friendliness of the system.

#### **3.4.1 Python (version 3.9.7)**

At the core of the toolkit is Python, a versatile and widely adopted programming language. Python's extensive ecosystem of libraries, coupled with its readability and ease of use, makes it an ideal choice for tasks ranging from data pre-processing to model development.

#### **3.4.2 Pandas (version 1.4.3)**



Pandas, a robust data manipulation library, plays a crucial role in handling datasets effectively. Its capabilities in cleaning, organizing, and merging datasets simplify complex data-related tasks, ensuring that the recommendation system operates with well-prepared and structured data.

### **3.4.3 NLTK (version 3.8.1)**

NLTK takes centre stage in the pre-processing of textual data. With a suite of libraries for natural language processing, NLTK aids in tasks such as tokenization and cleaning of textual information, allowing the system to effectively analyse movie descriptions and keywords.

### **3.4.4 Django (version 4.2.7)**

Django is a high-level web framework for Python, employing the Model-View-Controller (MVC) architectural pattern. Known for its simplicity and rapid development capabilities, Django includes features like an Object-Relational Mapping system, an automatic admin interface, URL routing, and a secure template engine. Its modular structure allows developers to organize projects into reusable applications. Django is widely used for web development, content management systems, and e-commerce platforms. The community provides extensive documentation, and its versatility is demonstrated by the framework's use in well-known websites such as Instagram and Pinterest. Overall, Django offers a robust and scalable solution for building web applications. This ensemble of tools, carefully chosen and integrated, forms the technological backbone of the research endeavour. Together, they empower the creation of an innovative content-based recommendation system, promising a refined user experience and enhanced decision-making in the realm of movie suggestions (Idris et al., 2020).

### **3.4.5 Jupyter Notebook/Google colab**

Jupyter Notebook and Google Colab are powerful tools widely used in data science, machine learning, and various scientific computing applications.

**Jupyter Notebook:** The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience. It is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It supports multiple programming languages, including Python, R, and Julia.

**Google Colab:** Google Colab, short for Colaboratory, is a free, cloud-based version of Jupyter Notebooks provided by Google. It allows users to write and execute Python code in a collaborative environment, eliminating the need for local installations. Colab offers free access to GPU resources, which is particularly advantageous for machine learning tasks that require substantial computing power. It integrates seamlessly with Google Drive, facilitating easy sharing and collaborative work on notebooks. Colab has become a popular choice among data scientists and machine learning practitioners for its convenience and resource accessibility.

### **3.4.6 Hardware Requirement**

Operating system: Windows 11.

Memory (RAM): 8GB

## **3.5 Summary**

This chapter delivers a thorough insight into the methodology employed for the development of a Movie Content-Based Recommendation System (CBRS). The methodology unfolds in essential steps, commencing with a detailed analysis of the dataset and pre-processing measures to guarantee data quality. Notably, the chapter extensively explores the application of Natural Language Toolkit (NLTK) techniques, shedding light on their significance in the process. Furthermore, it delves into the importance and application of TF-IDF and Word2Vec, providing a comprehensive understanding of their roles in the recommendation system.

A notable feature of this section is the in-depth discussion on various feature selection techniques, with Chi-Square being a novel inclusion in this study. The chapter elucidates the significance and application of Chi-Square, underscoring its uniqueness as a feature selection method within the context of the study. Overall, the methodology chapter serves as a comprehensive guide, incorporating innovative techniques and highlighting the distinctive features that contribute to the development of an effective Movie Content-Based Recommendation System.

A notable aspect is the creation of an end-to-end website, allowing users to interact seamlessly with the recommendation system. The model, fine-tuned for accuracy, is deployed on the website's backend, providing real-time recommendations based on user input.

## **CHAPTER 4: ANALYSIS & IMPLEMENTATION**

### **4.1 Introduction:**

This chapter introduces the practical implementation of content based recommendation. The primary objectives of this phase are analysing user preferences and movie attributes, uncovering the patterns that defines cinematic tastes and implementation of the CBRS translating these insights into a personalized cinematic experience.

The initial phase encompasses data collection, loading, and cleaning to render it suitable for analysis and recommendation construction. Subsequently, the second phase involves a thorough examination of user behaviours, viewing patterns, and movie features. This analysis lays the foundation for our content-based movie recommendation system.

Moving forward, the Experimentation phase will delve into the data, seeking to comprehend which features contribute to the high voting averages of movies. The aim is to determine whether the selected features collectively enhance the quality of recommendations. Finally, the deployment phase involves the implementation of an end-to-end OTT application, incorporating a Django App hosted on an AWS EC2 instance as the model application.

These phases collectively illustrate the sequential progression from data preparation to system deployment, each playing a crucial role in the development and enhancement of our content-based movie recommendation system.

### **4.2 Dataset:**

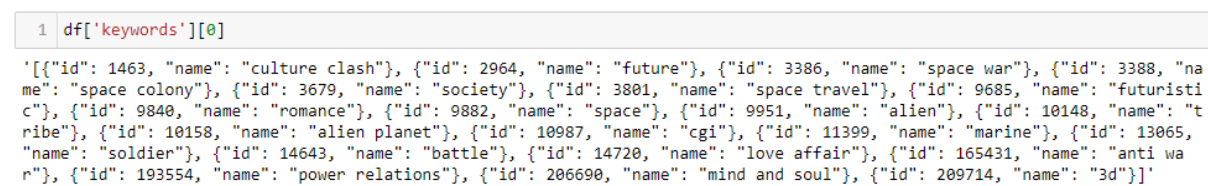
For this study, a comprehensive dataset was procured from the Kaggle platform, with a specific focus on the TMDb (The Movie Database) data. Dataset has two crucial csv files and both have different columns which are essential for this study. Each file has distinct information about movies provided. File1 is enriched with essential data related to movies. It includes details such as Genres, Keywords, Movie Overview, Runtime, Release Date, and other pertinent information crucial for understanding the characteristics of each movie. and The second CSV file focuses on the credits of movies, providing insights into the cast and crew involved in each production.

### **4.3 Data Preparation**

The collected dataset from Kaggle's TMDb, consisting of two CSV files, “tmdb\_5000\_movies.csv” and “tmdb\_5000\_credits.csv”, was efficiently loaded into a Jupyter notebook for analysis using Pandas data frames. File1, with 20 columns, encompasses comprehensive movie details, including Genres, Keywords, Overview, Runtime, Release Date, among others. File2, with 4 columns, captures credits information such as cast and crew. Both files share 4803 rows. After merging on the common feature 'Title of the movie,' the combined data frame yields 4809 rows and 23 columns, providing a consolidated and enriched dataset for our content-based movie recommendation system.

### 4.3.1 Handling Categorical Features in JSON Format

A noteworthy characteristic of our dataset is the prevalence of categorical features stored in JSON format, encapsulating vital information within dictionaries. Figure 4.3.1 represents the structuring of the 'keyword' feature, illustrating the organization of information in the form of dictionaries.



```
1 df['keywords'][0]
```

```
'[{"id": 1463, "name": "culture clash"}, {"id": 2964, "name": "future"}, {"id": 3386, "name": "space war"}, {"id": 3388, "name": "space colony"}, {"id": 3679, "name": "society"}, {"id": 3801, "name": "space travel"}, {"id": 9685, "name": "futuristic"}, {"id": 9840, "name": "romance"}, {"id": 9882, "name": "space"}, {"id": 9951, "name": "alien"}, {"id": 10148, "name": "tribe"}, {"id": 10158, "name": "alien planet"}, {"id": 10987, "name": "cgi"}, {"id": 11399, "name": "marine"}, {"id": 13065, "name": "soldier"}, {"id": 14643, "name": "battle"}, {"id": 14720, "name": "love affair"}, {"id": 165431, "name": "anti war"}, {"id": 193554, "name": "power relations"}, {"id": 206690, "name": "mind and soul"}, {"id": 209714, "name": "3d"}]'
```

**Figure 4.1 Screenshot of Jason format of a feature column keywords**

In addition to 'keyword,' there are five additional columns exhibiting a similar JSON format but with distinct dictionary structures. To streamline the extraction of information, bespoke functions were developed. These functions successfully parsed and transformed the varied dictionary shapes into coherent lists as shown in Figure 4.2. The subsequent figure encapsulates the transformed columns of the "Keywords" feature, presenting a consolidated view of the enriched data structure derived from these categorical columns.

```

1 df['keywords'].head()
0    [culture clash, future, space war, space colon...
1    [ocean, drug abuse, exotic island, east india ...
2    [spy, based on novel, secret agent, sequel, mi...
3    [dc comics, crime fighter, terrorist, secret i...
4    [based on novel, mars, medallion, space travel...
Name: keywords, dtype: object

```

**Figure 4.2 Transformed Columns of the "Keywords" Feature**

### 4.3.2 Identifying and Eliminating Missing Values:

To fortify the integrity and reliability of our analyses, a systematic approach was taken to address missing values in the dataset. Initially, the decision was made to drop the first two features, 'Homepage' and 'Tagline,' each containing more than 800 missing values. Recognizing their limited contribution and to prevent potential distortion in subsequent analyses.

Furthermore, additional scrutiny revealed a handful of columns with one or two missing values, notably within the 'Overview' feature—a pivotal component in Content-Based Recommendation Systems (CBRS). Given the significance of this feature, rows containing missing values in the 'Overview' were selectively deleted.

### 4.3.3 Feature Engineering: Enhancing Textual Insights

As part of the data pre-processing phase, we engaged in feature engineering to extract additional numerical insights from existing text columns. Four new numerical columns were created to enrich our dataset.

- 'No of Keywords': This column quantifies the number of keywords associated with each movie, providing a numeric representation of the richness of descriptive terms.
- 'No of Genres for a Movie': Offering a numerical count, this column captures the diversity of genres associated with each movie, shedding light on the multidimensional nature of genre categorization
- 'Year of Release': Extracted from the release date information, this column condenses the temporal aspect, providing a numerical representation of the movie's release year for chronological analyse

- 'Number of Words in Overview Columns': By quantifying the length of movie overviews in terms of words, this column facilitates a nuanced exploration of the textual content, offering insights into the depth and complexity of movie description

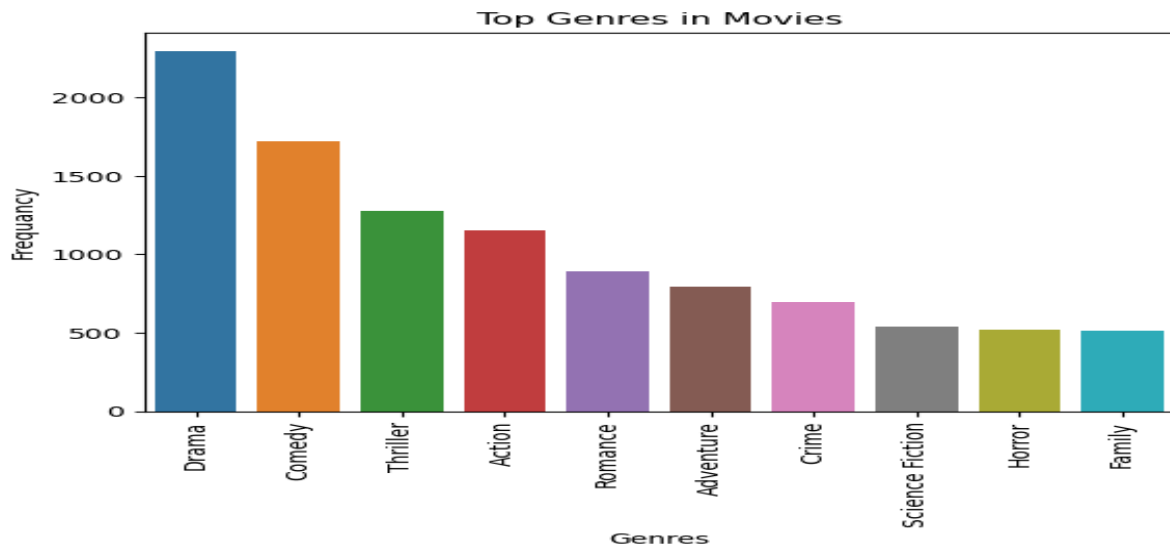
These newly created numerical columns are strategically designed to enhance our ability to analyse textual data more effectively and will serve as valuable assets during the Exploratory Data Analysis (EDA) phase, contributing to a more comprehensive understanding of our dataset.

## 4.4 Exploratory Data Analysis

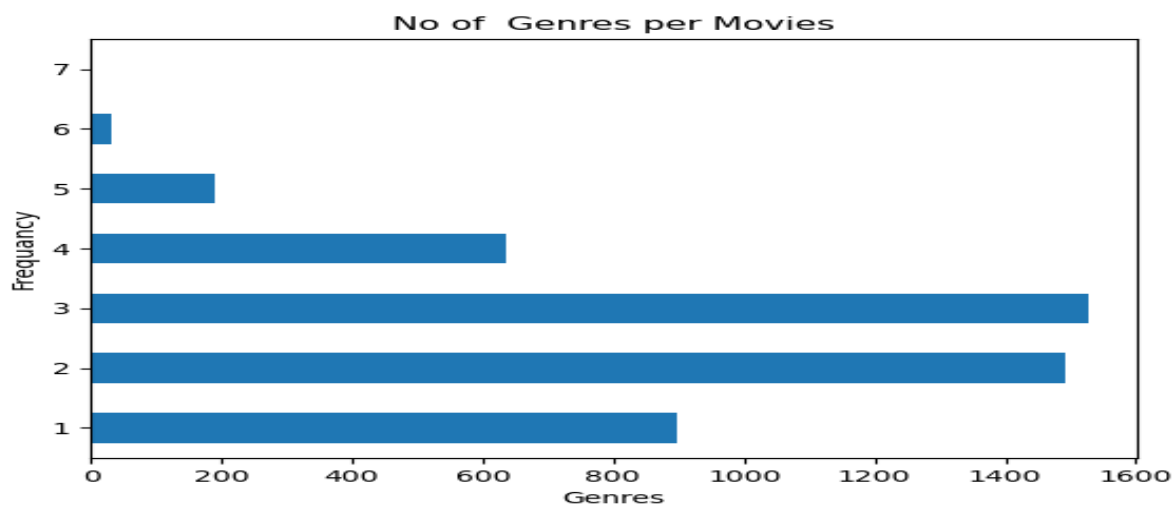
### 4.4.1 Genres & Keywords

The predominant genre in the dataset is 'Drama,' with a substantial majority of movies falling into this category. Following closely are Comedy, Thriller, and Action genres.

A notable observation is that a significant portion of movies in the dataset falls under the multi-genre category. Additionally, there are instances where the Genre column is missing for some movies.



**Figure 4.3 Top Genres in Movies**

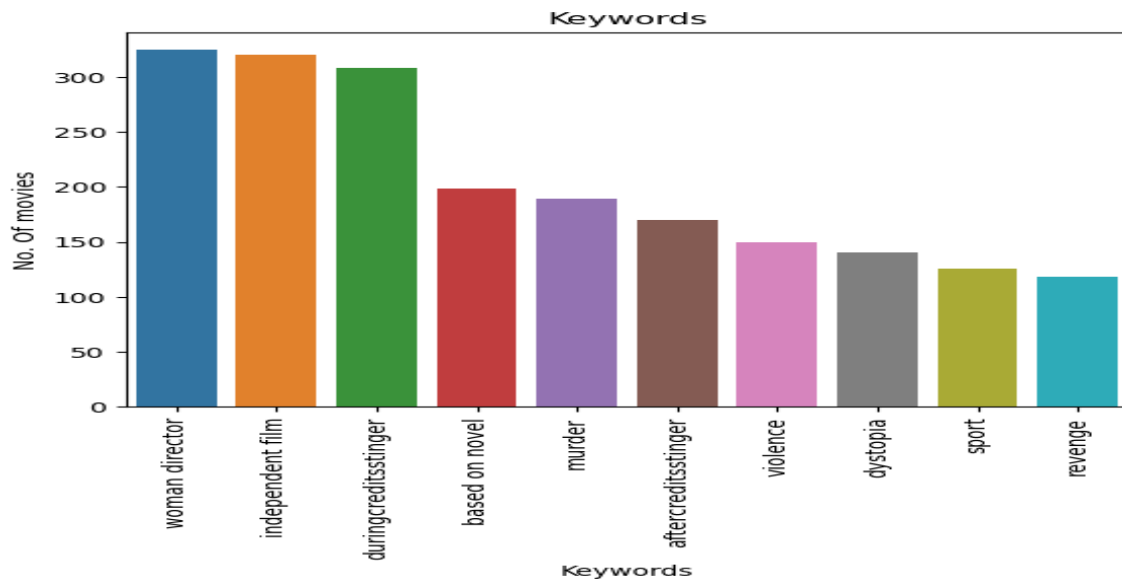


**Figure 4.4 Number of genres per Movie**

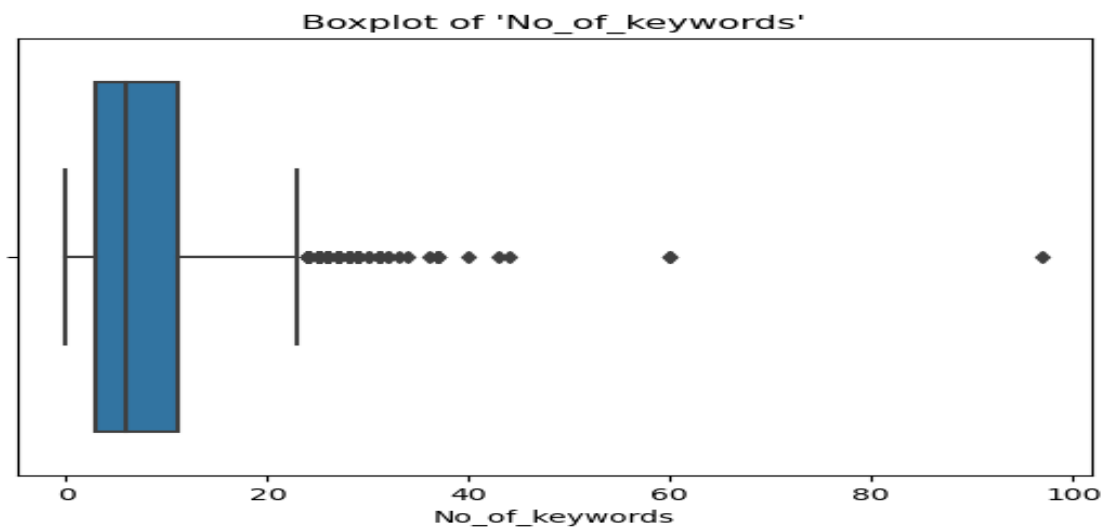
When examining the distribution, it becomes evident that the most common scenario is movies having three genres, with two and four genres per movie being the subsequent prevalent occurrences.

A noteworthy insight from our dataset reveals the prevalence of certain keywords, shedding light on recurring themes within the movie entries. Notably, the most frequently used keywords

include 'Women Director,' 'Independent Film,' and 'Duringcreditsstinger,' hinting at prevalent motifs and cinematic elements across the dataset.



**Figure 4.5 Top Keywords Most number of movies have**



**Figure 4.6 Boxplot of ‘Number of Keywords’**

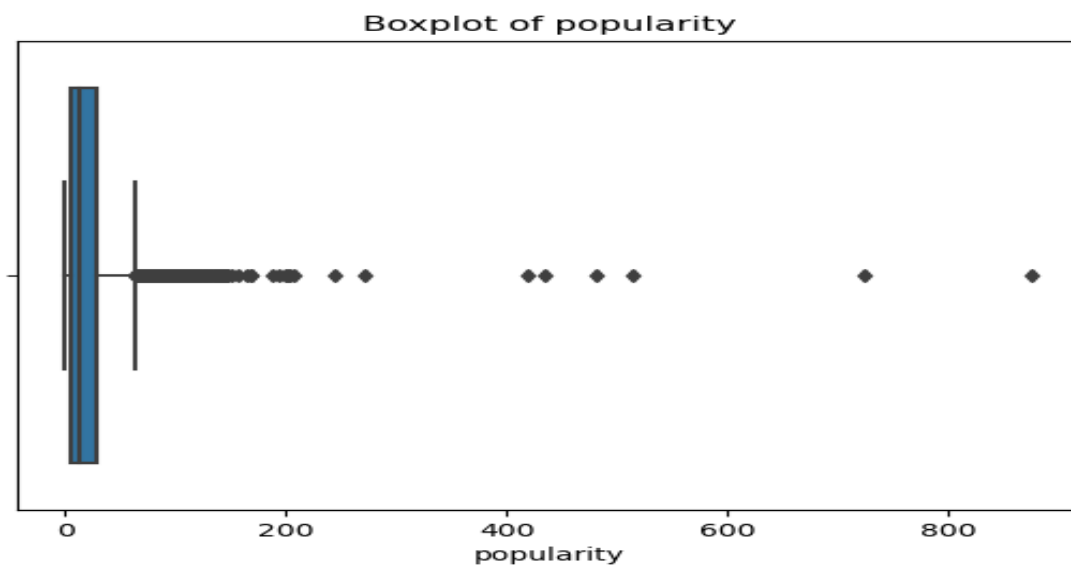
Moreover, the dataset showcases a wide variation in the number of keywords associated with each movie. The keyword count spans a considerable range, from 0 to 95 and densely distributed between 5 to 15 keywords per movie, as illustrated by a boxplot visualization. This visual representation not only emphasizes the diversity in the number of keywords but also highlights the distribution's interquartile range and the presence of potential outliers. The observed variability underscores the complexity and diversity present in our dataset,



emphasizing the need for a nuanced approach in exploring and understanding the textual features associated with each movie.

#### 4.4.2 Popularity & Ratings

Represented as an index, popularity serves as a dynamic measure of a movie's widespread appeal. This metric encapsulates factors such as viewership, social media mentions, and overall public interest, offering insights into the cultural impact and reach of a movie.

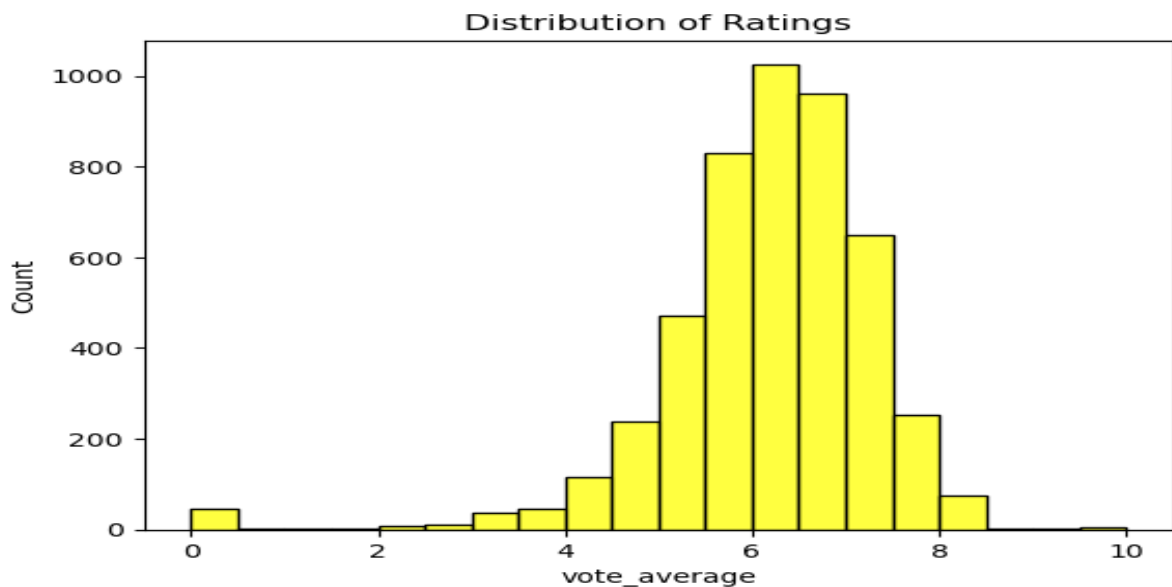


**Figure 4.7 Boxplot of Popularity**

The distribution depicted in the figure suggests that a significant majority of movies in the dataset have a popularity index below 100. However, it is notable that a subset of movies appears as potential outliers, indicating the presence of films with exceptionally high popularity. While these outliers might initially be perceived as exceptional cases, they also point towards the existence of movies that have garnered an extraordinary level of popularity.

As an integral part of the dataset's rating system, the vote average provides a quantitative assessment of a movie's overall quality and viewer satisfaction. Derived from user ratings, this

metric aids in understanding the collective sentiment and appreciation expressed by the audience



**Figure 4.8 Distribution of Ratings**

Figure illustrates a diverse distribution of film ratings within our dataset. Notably, there is a presence of films with extremely low ratings, starting from 0. The central tendency of ratings falls within the range of 5 to 6, suggesting that the majority of films in our dataset have average ratings within this bracket.

Moreover, a notable cluster of films falls within the range of 6 to 8, indicative of movies that are generally better rated. Beyond this range, there exist outliers representing films with exceptional ratings.

#### **4.4.3 Weighted Rating**

To address the anomalies formulated by vote average feature in the dataset, new column named weighted Rating is created. This selection will be based on the True Bayesian estimate weighted rating, a metric widely employed in platforms like IMDb

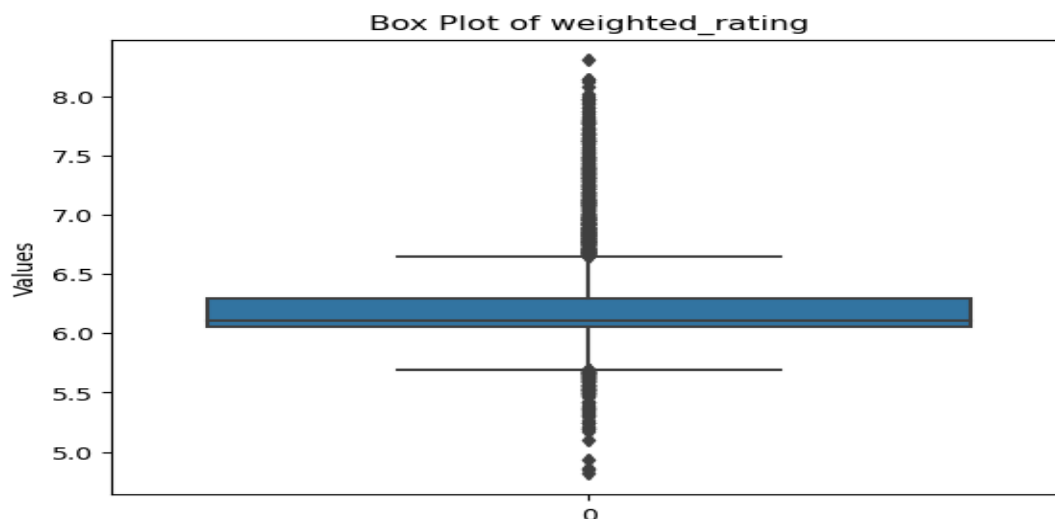
$$\text{Weighted Rating (WV)} = (\text{votes} \div (\text{votes} + \text{min votes})) \times R + (\text{min votes} \div (\text{votes} + \text{min votes})) \times C$$

$R$  = Mean of Movie ratings

Votes = Total no. votes for the movie

min votes = The minimum number of votes needed to secure a place in the Top 250 is presently set at 25,000.

$C$  = The average (mean) vote across the entire report



**Figure 4.9** Boxplot of Weighted ratings

#### 4.4.4 Languages

An analysis of the dataset reveals a predominant presence of movies originally made in English, constituting approximately 94% of the total. Following English, French emerges as the second most prevalent language, albeit with a significantly lower representation. Collectively, movies from all other languages make up the remaining 6%, underscoring the dominance of English-language productions within the dataset.

Distribution of Movie language\_counts

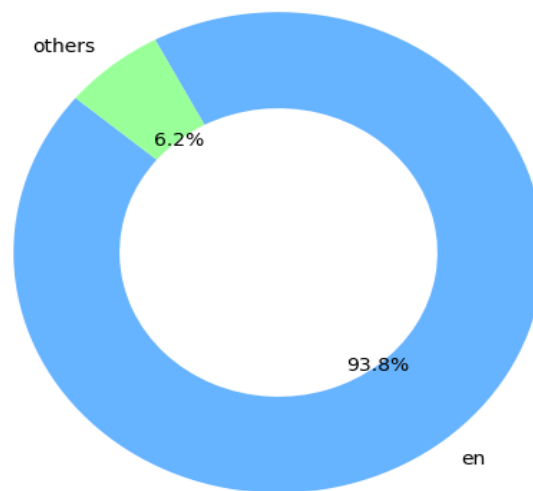


Figure 4.10 Pie - Chart of Languages

#### 4.4.5 Top Grossed movies

Securing the top two spots for highest profits are the cinematic behemoths "Avatar" and "Titanic." These blockbuster successes are closely followed by "Jurassic World," solidifying its position among the highest-grossing movies.

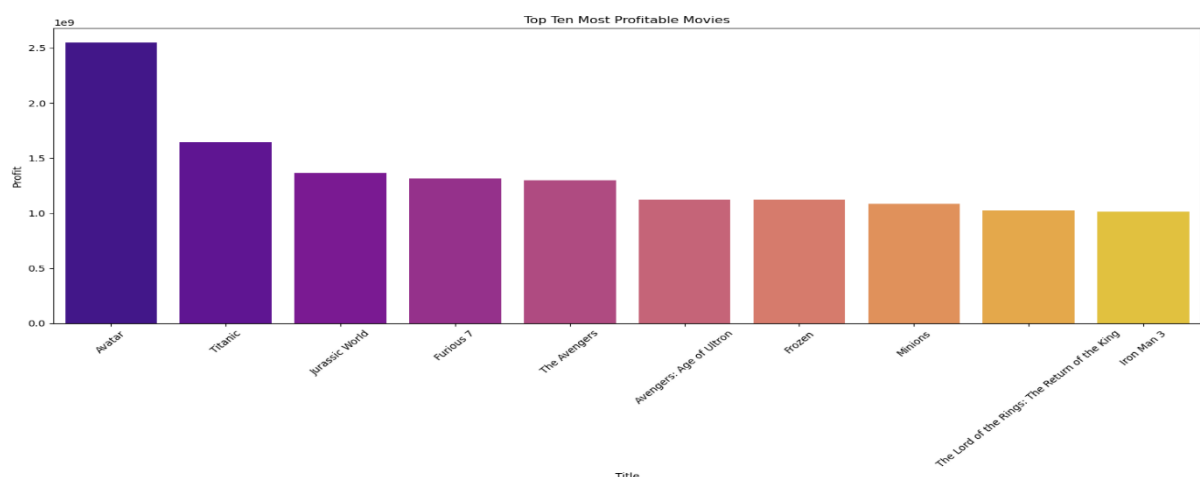
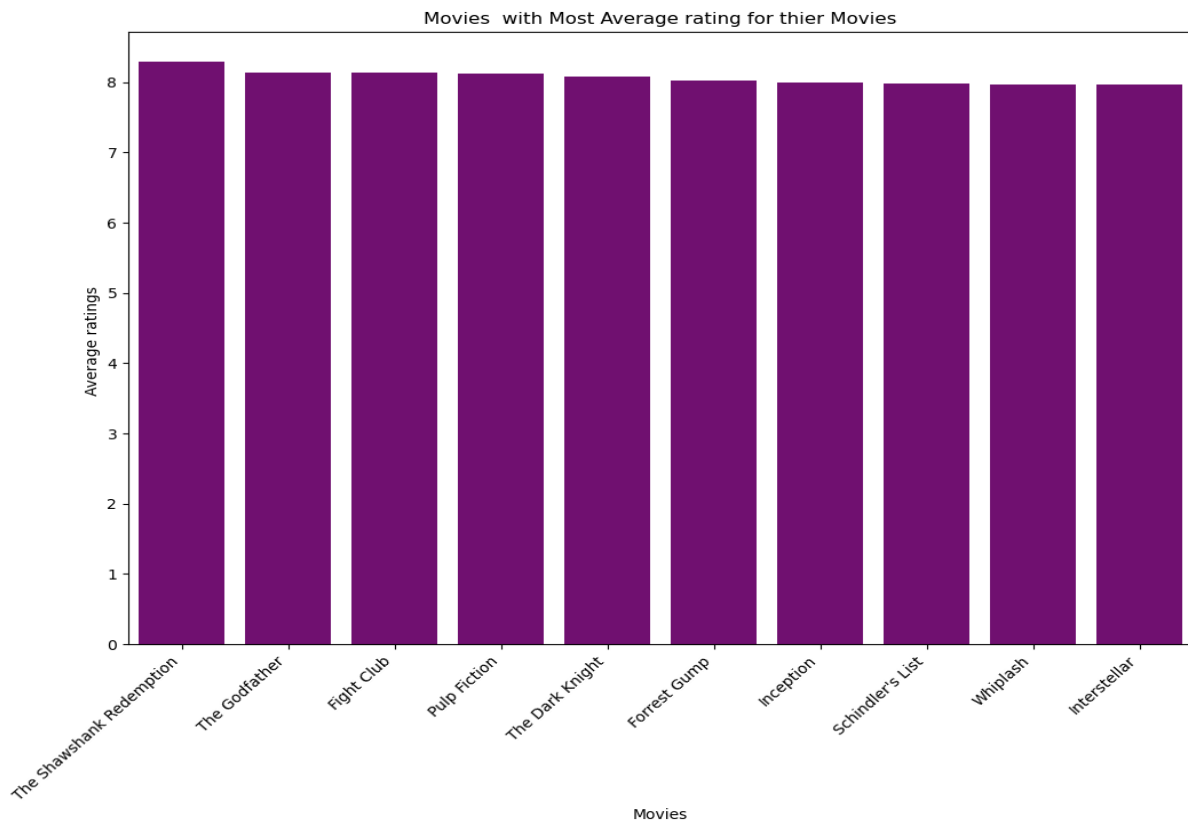


Figure 4.11 Top 10 Most popular Movies

#### 4.4.6 Top Rated Movies

In the world of movies, "The Shawshank Redemption" and "The Godfather" take the lead as the most loved films in our dataset. Right on their heels are "Fight Club" and "Pulp Fiction," making a strong impression and earning high ratings from the audience. These movies stand out for their captivating stories and unique styles, making them crowd favourites in the diverse landscape of cinematic excellence.



**Figure 4.12 Movies with Most average rating of their Movies**

## 4.5 Exploratory Data Analysis (Bivariate analysis)

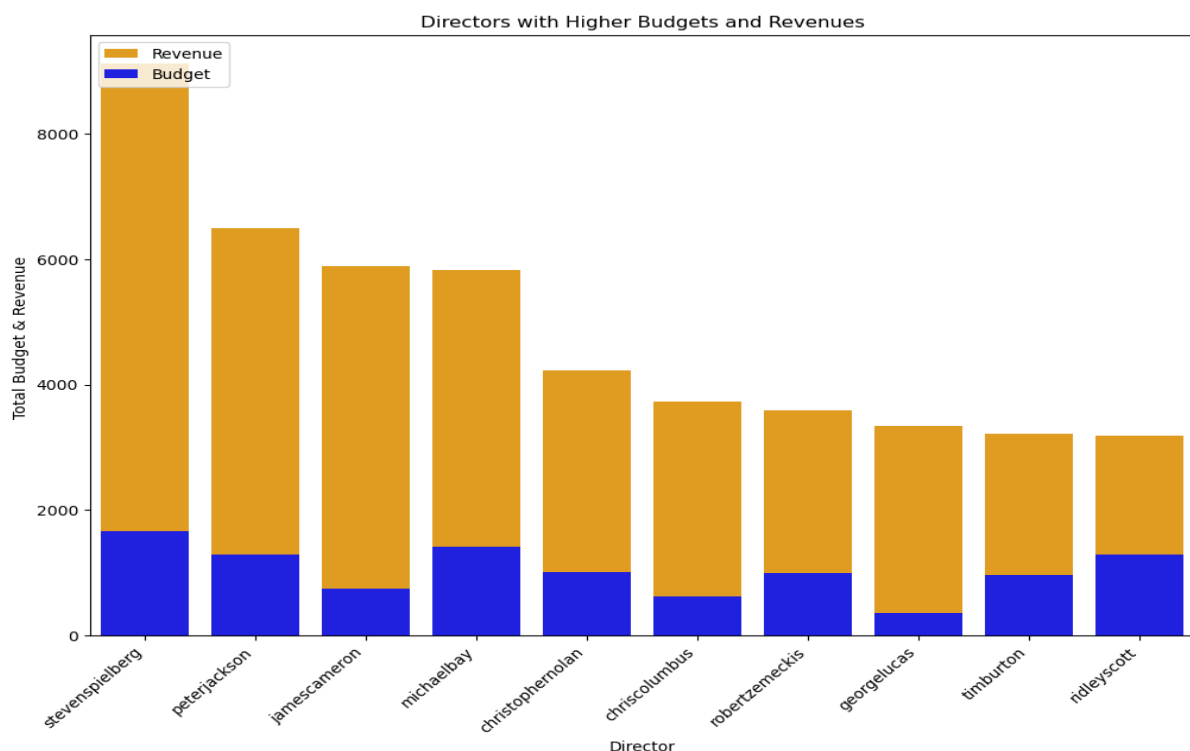
In this section, we explore relationships between pairs of features to gain a deeper understanding of the data. Bivariate analysis allows us to assess how two features interact and influence each other, contributing to a more comprehensive understanding of the dataset.

### 4.5.1 Director Vs Budgets& Revenue

At the forefront of prolific directors stands Steven Spielberg, a luminary in the film industry, credited with an impressive tally of over 25 movies throughout his illustrious career. Not only does Spielberg lead in sheer volume, but he also commands a notable presence in terms of budgets and revenue allocated to his cinematic endeavours.

Closely following Spielberg are icons Woody Allen and Martin Scorsese, each with a substantial repertoire of films that have left an indelible mark on the cinematic landscape.

Securing the second position in both budgets and revenue is Peter Jackson, renowned for helming epic franchises like "Lord of the Rings" and "The Hobbit." James Cameron, with a comparatively smaller filmography, takes the third spot in terms of high revenue.



**Figure 4.13 Director with Higher Budgets and Revenues**

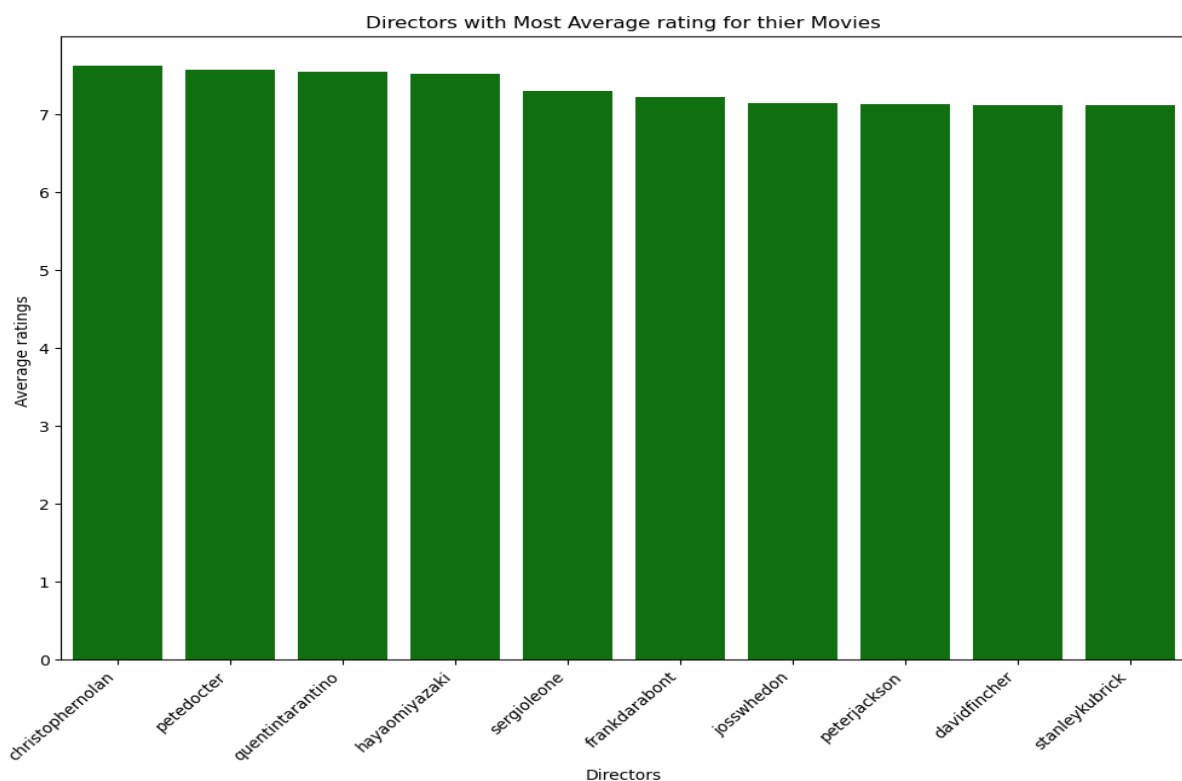
#### 4.5.2 Director vs Ratings

Among directors with a filmography exceeding two entries, the esteemed filmmaker Christopher Nolan emerges as a frontrunner, commanding attention with an impressive weighted average of 7.2 for his internationally acclaimed works.

Following closely in the rankings are Pete Docter and Quentin Tarantino, securing the second and third positions with noteworthy average ratings of 7.2 and 7.1, respectively. This places

them in the level of consistently high-rated directors, a testament to the enduring impact and exceptional quality of their cinematic contributions.

These remarkable figures not only shape narratives but also consistently garner acclaim, signifying a sustained commitment to cinematic excellence. Their films, marked by high audience ratings, stand as a testament to their ability to captivate audiences and leave a permanent mark on the world of cinema.



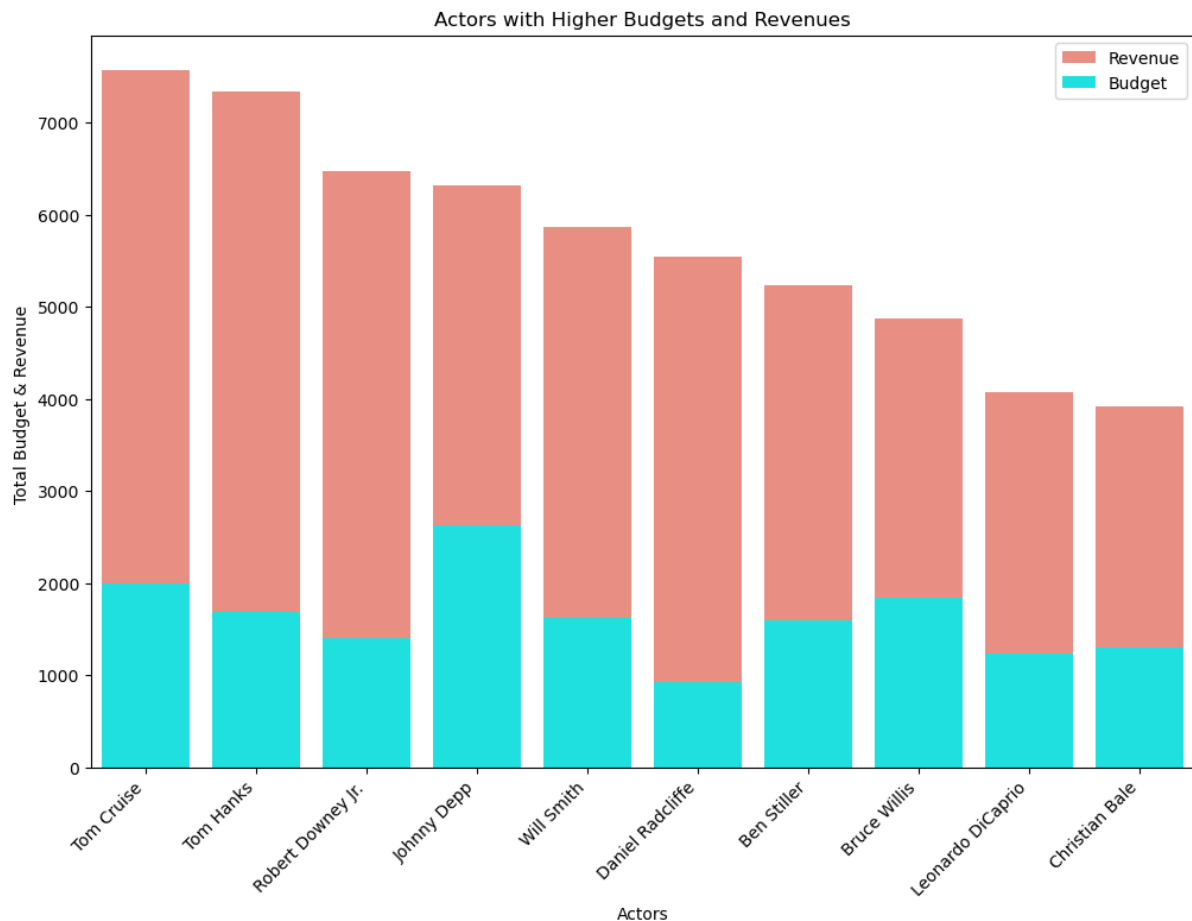
**Figure 4.14 Directors with Most average ratings for their movies**

### 4.5.3 Actor vs Budget & Revenue

As a leading actor, Tom Cruise takes the spotlight with movies boasting both substantial budgets and impressive revenues. Following closely in the second position is Tom Hanks, further solidifying the trend of accomplished actors making a significant impact not only on the cinematic stage but also on the financial success of their films.

Securing the third and fourth positions in this cinematic financial line-up are Robert Downey Jr. and Johnny Depp. Their presence in these slots underscores the enduring financial impact

that these accomplished actors bring to their movies. With Tom Cruise and Tom Hanks leading the pack, the collective influence of these actors highlights the substantial role that star power plays in both budget allocation and revenue generation within the film industry. Figure 4.15 shows the visualization of the Actors with Higher budgets and revenue. Bar with Aqua color represents Budgets and other one represents Revenue



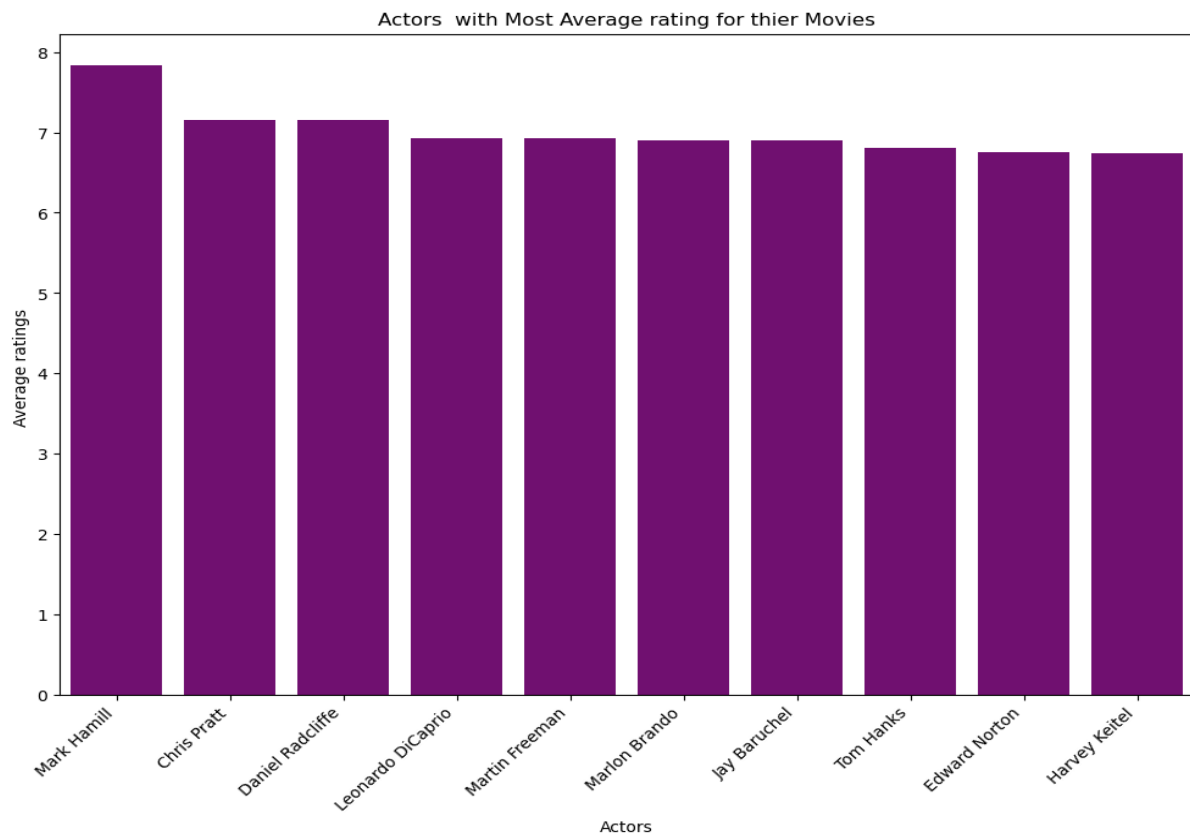
**Figure 4.15 Actors with higher budgets and revenues**

#### 4.5.4 Actor vs Ratings:

When it comes to consistently high-rated movies in our dataset, Mark Hamill and Chris Pratt steal the spotlight as the main actors. Their performances have garnered acclaim, reflecting in the impressive average ratings of their movies.



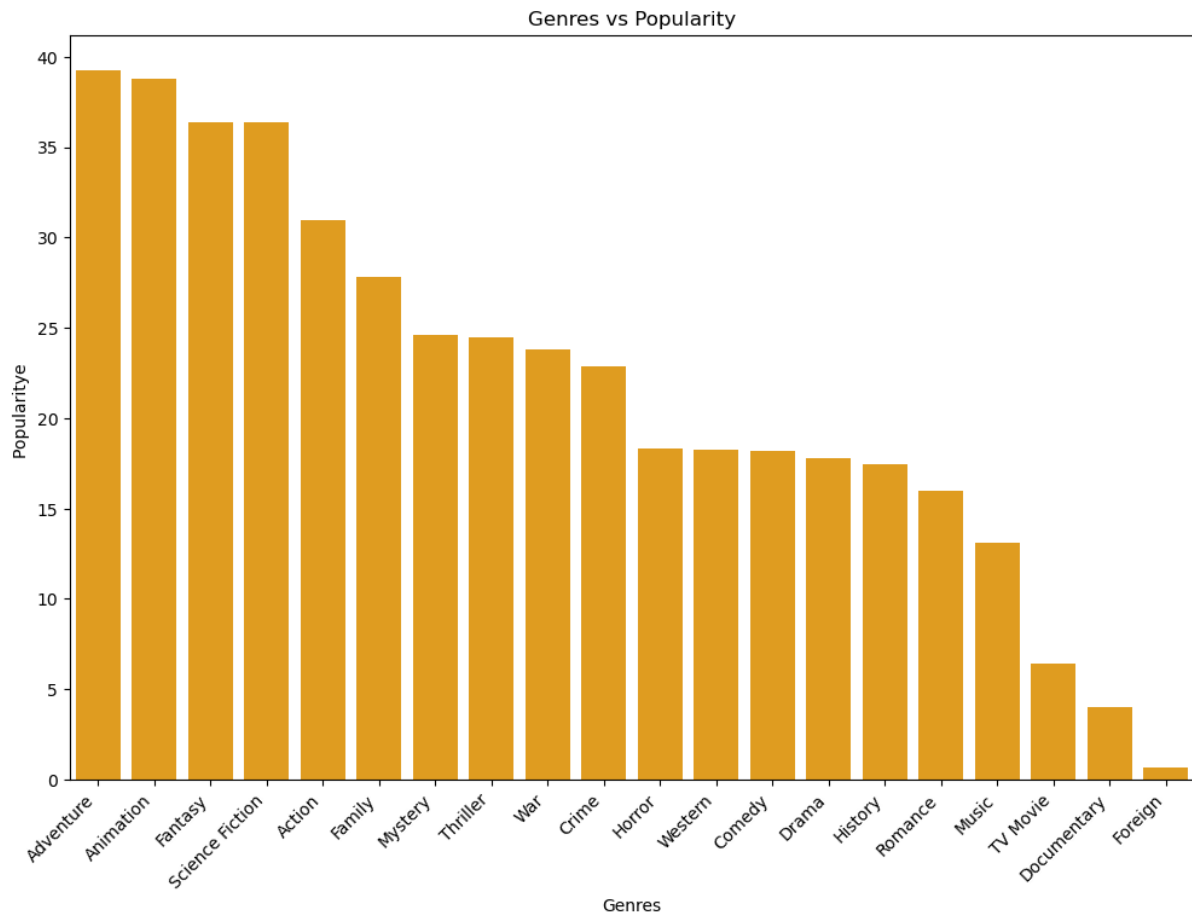
Following closely are Daniel Radcliffe and Leonardo DiCaprio, further solidifying the trend of these actors delivering noteworthy performances that resonate with audiences. Their presence adds to the star power contributing to the high average ratings observed in the dataset. Figure 4.1 shows the visualization of Actors and their average ratings for their movies



**Figure 4.16 Actors with most average rating for their movies**

#### 4.5.5 Genres vs Popularity

Section 4.4.1 might have suggested that Drama, Comedy, Thriller are top genres with most number of movies, but when genres are aligned with popularity and compared, Adventure, Animation and fantasy are the top 3 movies



**Figure 4.17 Visualization representing genre vs popularity**

## 4.6 Chi-Square Test on Categorical columns

The Chi-Square Test serves as a valuable statistical tool when examining the relationship between categorical variables within a dataset. This test has been conducted to evaluate what all categorical columns in the dataset has significant association with Vote average.

### 4.6.1 Hypothesis Testing

Test: Chi-square test.

- Null Hypothesis H0: The ratings range is independent of Given Categorical Column.
- Alternative Hypothesis H1: The ratings Range is associated with Given Categorical column.
- $\alpha = 0.05$

## 4.6.2 Data Preparation

Initial step involves, organising data into a contingency table, a structured representation of the frequencies of each combination of categories for the two variables under consideration. Contingency table is created for all categorical columns except keywords and overview columns. Figure 4.17 is the contingency table created for genre column and in the same way tables are created for remaining categorical columns. A new column by name Rating range is created based on the weighted average.

- Movies with a weighted average rating equal to or above 7 are categorized as "High."
- Films receiving a weighted average rating less than 5 fall into the "Low" category.
- All other movies, falling between the range of 5 to 6.9, are designated as "Medium" rated.

Rating_Range	HIGH_VOTE_AVG	LOW_VOTE_AVG	MEDIUM_VOTE_AVG
new_genres			
action	1	5	15
action,adventure	0	1	7
action,adventure,animation,comedy,family	0	1	1
action,adventure,animation,comedy,family,fantasy,romance	0	0	1
action,adventure,animation,comedy,family,fantasy,sciencefiction	0	0	1
...	...	...	...
western,comedy	1	0	0
western,drama	0	0	2
western,drama,adventure,thriller	1	0	0
western,history	0	0	1
western,history,war	0	0	1

1174 rows × 3 columns

**Figure 4.18 contingency table created for genre column**

For the Chi-Square tests conducted in this study, a predetermined significance level, denoted as  $\alpha$ , has been set at 0.05. This choice signifies that any p-value resulting from a Chi-Square test that falls below  $\alpha$  leads to the rejection of the null hypothesis. In practical terms, this indicates a statistically significant association between the categorical variables under consideration. A p-value less than  $\alpha$  serves as a threshold for rejecting the null hypothesis, signifying that the observed associations between the columns are beyond what would be

expected by chance alone. This Chi-Square test has been specifically designed to explore the relationship between the columns under scrutiny and audience engagement.

## **4.7 Pearson Correlation Coefficient**

This experiment is designed with the specific aim of concluding the relationship between two numerical variables within our dataset. In this investigation, we cast a wide net by considering all numerical columns, including newly created ones, to assess their correlation with the column vote average. Utilizing the Pearson Correlation coefficient, we gauge the strength and direction of linear relationships between these numerical variables and the vote average.

### **4.7.1 Hypothesis Testing**

Test: Person correlation.

- Null Hypothesis  $H_0$ : The ratings range is independent of Given Categorical Column.
- Alternative Hypothesis  $H_1$ : The ratings Range is associated with Given Categorical column.
- $\alpha = 0.05$

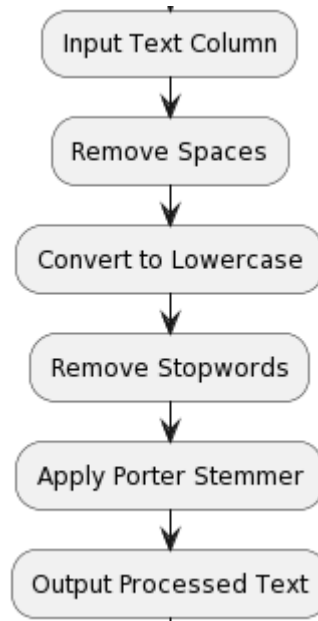
For this analysis, as with the Chi-Square test, the significance level ( $\alpha$ ) has been established at 0.05. This predetermined threshold signifies that any p-value resulting from the Pearson Correlation analysis falling below  $\alpha$  prompts the rejection of the null hypothesis. This strategic choice ensures that only statistically significant relationships are considered, fortifying the reliability and robustness of our conclusions.

## **4.8 Data Pre-processing:**

To clean the text columns and prepare for modelling certain pre-processing steps were applied on each feature to ensure cleanliness, uniformity, and relevance. These transformative steps are as follows:

- The initial step involves the thorough removal of unnecessary spaces within the text.
- To establish consistency in the textual content, all alphabets within the text are uniformly converted to lowercase.
- A crucial phase involves the judicious removal of common stop-words, frequent terms like 'and,' 'the,' or 'is' that contribute minimally to substantive meaning. This process refines the text, focusing analytical efforts on words of significance.

- To distil the essence of the textual content, the Porter Stemmer algorithm is applied. This algorithm systematically reduces words to their root or base form, capturing core semantic meaning and fostering uniformity.



**Figure 4.19 Data Pre-Processing Steps**

These pre-processing steps collectively refine text columns, creating a standardized and meaningful foundation for subsequent analyses. This preparatory phase enables the extraction of nuanced insights from the textual components of the dataset.

#### **4.9 Experiment 3: Feature Combinations**

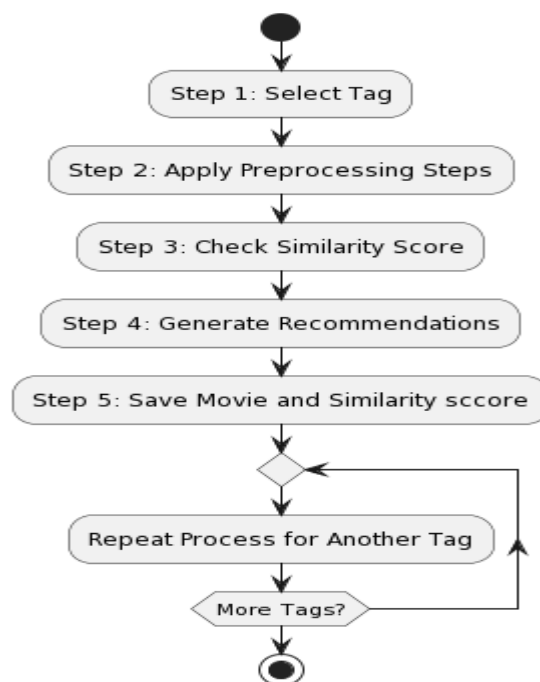
The primary objective of this experiment is to explore diverse feature combinations with a focus on applicability of recommending movies. Goal is to systematically investigate and evaluate different combinations and which feature combination is recommending movies more accurately.

'22nd century, parapleg marin dispatch moon pandora uniqu mission, becom torn follow order protect alien civilization. cultu reclash futur spacewar spacecoloni societi spacetravel futurist romanc space alien tribe alienplanet cgi marin soldier battl loveaffair antiwar powerrel mindandsoul 3d action adventur fantasi sciencefict ingeniousfilmpartners twentiethcenturyfoxfilm corporation duneentertainment lightstormentertainment samworthington zoesaldana sigourneyweaver stephenlang jamescameron eng lish español237000000 4 21 2787965087 162.0'

**Figure 4.20 Tag created by concatenating Categorical and Numerical columns**

In this study, seven distinct combinations were meticulously chosen to create tags, laying the foundation for generating movie recommendations based on these carefully curated sets of features.

- **Tag1:** Comprises all categorical features selected through the Chi-Square Test along with key features Movie overview and keywords.
- **Tag2:** Solely relies on the Movie Overview and Keywords columns.
- **Tag3:** adds another columns genre to Tag2
- **Tag 4:** Incorporates a blend of essential features, including Cast, Movie Overview, Keywords, and Genres.
- **Tag 5:** Is created only with feature keywords
- **Tag6:** is created only with feature Overview
- **Tag7:** includes all the numerical columns added along with first tag. Numerical features are added after converting them to string type and separated by space at the end of first tag.

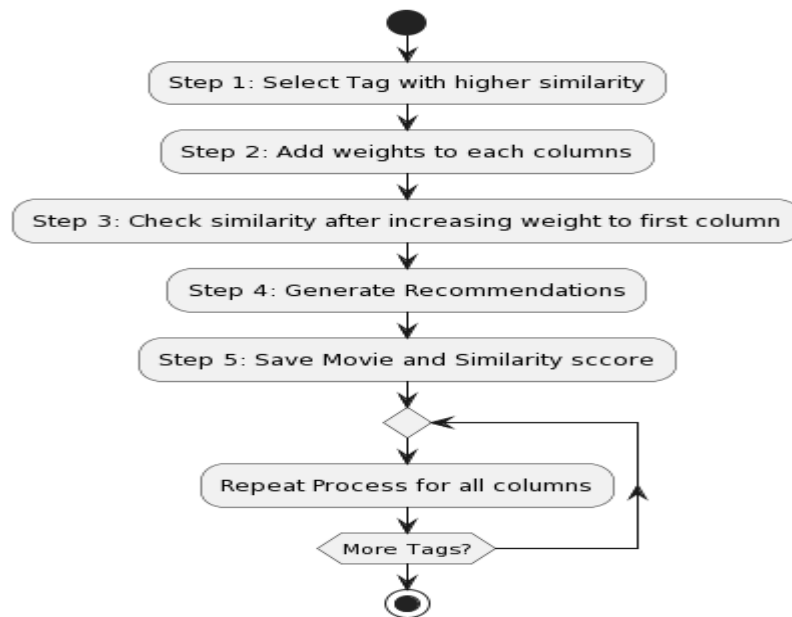


**Figure 4.21 Steps to generate Recommendations**

A comprehensive study was done on above all tags created by concatenating different feature columns. Movie recommendations were generated for all the tags and compared to check which feature combination works well with content based recommendation.

#### 4.10 Fine tuning Movie Recommendations by assigning weights to columns

In this phase of the study, the exploration focuses on assessing the impact of adding weights to columns before their conversion into tags, with the objective of gauging whether this approach contributes value to the movie recommendations. Tags that have generated recommendations with a higher similarity percentage were selected for this part of the study, and weight was added to one column at a time before creating a tag.



**Figure 4.22 Steps to Generate recommendations using Weighted tags.**

Each tag created was then compared for the same movies used in Experiment 3 to check whether adding weights added any value to the recommendations. This phase holds significance as it meticulously dissects the impact of weighted features on the recommendation system. The step-by-step addition of weights to columns allows for a nuanced understanding of how this refinement contributes to the overall quality of movie suggestions.

#### 4.11 Implementation of TF-IDF Vectorization and Cosine Similarity

In this section, we detail the steps taken to implement TF-IDF (Term Frequency-Inverse Document Frequency) vectorization and cosine similarity on movie tags. The goal of this

implementation is to quantify the textual content of movie tags and determine the similarity between different tags. This process is fundamental for generating relevant and personalized movie recommendations in a content-based recommendation system.

## Step 1: TF-IDF Vectorization

TF-IDF is a numerical statistic that reflects the importance of a term in a collection of documents. In our case, the documents are movie tags, and the terms are the individual words within these tags. The TF-IDF vectorization process involves the following steps:

- A set of sample movie tags is collected. These tags serve as the basis for the TF-IDF analysis.
- The scikit-learn library is utilized for TF-IDF vectorization. The TF-IDF Vectorizer class is employed to convert the sample tags into a TF-IDF matrix.
- The resulting TF-IDF matrix represents the importance of each term in the context of the entire set of movie tags.

## Step 2: Cosine Similarity Calculation

Cosine similarity is employed to measure the similarity between pairs of TF-IDF vectors. It computes the cosine of the angle between two vectors, providing a metric for their similarity. The following steps outline the process:

### **Cosine Similarity Function:**

- The scikit-learn library provides a Cosine-similarity function, which is applied to calculate the cosine similarity between the TF-IDF vectors of different tags.
- The result is a cosine similarity matrix, where each entry (i, j) indicates the cosine similarity between tags i and j.

## **4.12 Deployment of Django-Based OTT Application on AWS**

In this section, we detail the deployment process of our Django-based Over-The-Top (OTT) application, enriched with a content-based movie recommendation system, on the AWS Elastic Beanstalk platform. The objective is to ensure accessibility, scalability, and a seamless user experience in alignment with industry standards.

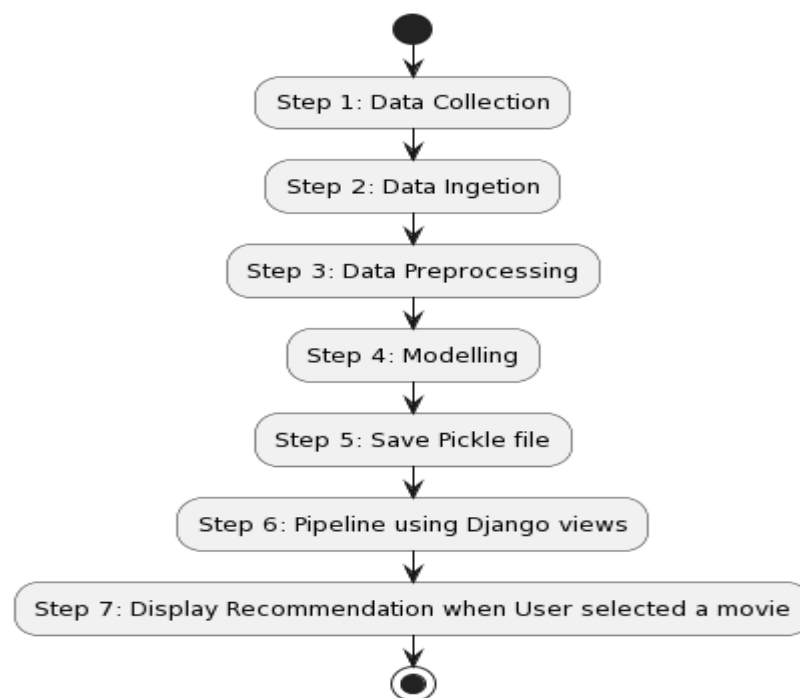
### **4.12.1 Django Application:**



Our Django project, structured to include models, views, templates, and static files, serves as the foundation for our OTT application. Key settings, such as database configurations and static file handling, are configured to suit the project requirements.

#### 4.12.2 Integration of Content-Based Recommendation System:

The integration of the CBRs into the Django application followed a meticulous four-step process, ensuring seamless functionality and personalized movie recommendations for user.



**Figure 4.23 Steps to Integrate of Content-Based Recommendation System**

- A curated movie dataset was procured, emphasizing key features such as genres, keywords, and movie overviews and converted the data into readable format.
- Raw data underwent a comprehensive pre-processing pipeline. This involved handling missing values, transforming textual data, and creating relevant numerical features.
- The TF-IDF vectorization and cosine similarity algorithm were chosen for their suitability in content-based recommendation scenarios.
- The selected model was trained on the pre-processed dataset, learning patterns and relationships between movie features.

- A dedicated API endpoint within Django views was established to handle user input and initiate the recommendation generation process.
- The trained CBRS model was invoked through the Django views, generating personalized movie recommendations based on user queries or preferences.
- The generated recommendations were seamlessly presented to the user through the application's user interface, providing a user-friendly and interactive experience.

The four-step integration process ensures that the Content-Based Recommendation System becomes an integral and user-centric component of the Django application and help install application as a package while deploying application in cloud.

#### **4.12.3 Homepage and Overview page with Recommendations:**

Home page of this OTT app designed leveraging insights derived from Exploratory Data Analysis (EDA). A search bar was integrated on this page with dropdown suggestion of movies from dataset. Upon selecting a movie from the dropdown menu or anywhere on the homepage, users are seamlessly directed to a dedicated Movie Overview page. This page acts as a comprehensive source of information, presenting detailed insights into the selected movie. Users can delve into the movie's synopsis, cast, genres, and other relevant details.

One notable feature of the Movie Overview page is the integration of our Content-Based Recommendation System (CBRS). Based on the user's selected movie, the page not only provides intricate details but also offers a curated list of recommended movies. Leveraging the top 10 similarity scores from the CBRS, users are presented with a personalized selection of movies that align with their preferences.

#### **4.12.4 Deploying OTT Application in AWS**

The deployment phase of our thesis involves the strategic utilization of Amazon Web Services (AWS), specifically leveraging the Elastic Compute Cloud (EC2) free-tier instance. This cost-effective approach aims to deploy our web application seamlessly while utilizing the benefits of Git for version control.

Following the creation of a free-tier EC2 instance on AWS, we initiate the deployment process by connecting to the EC2 instance through Secure Shell (SSH). With the repository successfully transferred to the EC2 instance, we proceed to configure the environment by

installing the necessary dependencies and setting up runtime configurations. This ensures that the environment on the EC2 instance mirrors the development environment, guaranteeing consistency in functionality.

The deployment is executed by launching the application on the EC2 instance, making it accessible through the instance's public IP address or domain. This simple yet effective deployment strategy not only provides a cost-effective hosting solution using the AWS free-tier but also establishes a streamlined process for ongoing development.

#### **4.13 Summary**

In summary, this section comprehensively details the Analysis and Implementation of the Content-Based Recommendation System (CBRS), culminating in the end-to-end deployment on AWS. Initially, a thorough Exploratory Data Analysis (EDA) was undertaken to extract valuable insights instrumental in designing the homepage of an Over-The-Top (OTT) website. Subsequently, Chi-square and Pearson correlation tests were employed to identify features explaining user interaction, shaping the foundation for our recommendation system.

The section further outlines meticulous data pre-processing steps utilizing the NLTK library to ensure data cleanliness and relevance. Following this, seven distinct tags were created based on varying combinations of features. The Text Frequency-Inverse Document Frequency (TF-IDF ) vectorizer was employed to convert textual tags into vectors, and cosine similarity was utilized to determine movies with similar content. Evaluation based on similarity scores ensued, leading to the selection of the best-performing tag.

To enhance the model, weights were assigned to each feature, and fine-tuning was conducted based on the performance of the selected tag. Insights derived from EDA, along with the best-performing model from experiments, informed the final deployment phase. An end-to-end Django application was deployed on AWS EC2, providing users with personalized movie recommendations upon selecting any movie from the website.

## CHAPTER 5: RESULTS AND DISCUSSIONS

### 5.1 Introduction

This chapter explores the results from four experiments that have been conducted from chapter 4. Aim is to evaluate the results generated from Chi-Square and Pearson Correlation tests for checking relation of Categorical and numerical columns respectively to check features that explain Audience interaction vote average. Later 7 different tags were evaluated to check which is recommending movies better based on the cosine similarity index.

### 5.2 Insights from Chi-square test

A chi-square test of independence was conducted on nine categorical columns against the newly created column 'Rating\_Range' to assess their potential association with audience engagement. As discussed in Section 4.6.1, the significance level ( $\alpha$ ) was set at 0.05. For each column, if the p-value resulting from the test is less than 0.05, we reject the null hypothesis. This rejection indicates that there is a statistically significant association between the respective categorical column and audience engagement, as measured by the 'Rating\_Range' variable.

**Table 5.1 Results from Chi-Square test**

Variable	X2	p_value
original language	166.4046	0
new genres	2886.774	0
production_countries	1141.9735	0
production_companies	7365.502	0.5519
Actor1	5433.799	0
Actor2	6280.1959	0
Actor3	6711.761	0
Director	6286.137	0
spoken_languages	1141.556	0.0338

Table 1 presents the p-values resulting from chi-square tests for all columns. Notably, the analysis indicates that we were unable to reject the null hypothesis for the 'production\_companies' column. Therefore, we conclude that 'production\_companies' has no statistically significant association with ratings. In contrast, for the remaining columns, the p-

values are below the 0.05 threshold, leading us to reject the null hypothesis. Consequently, we assert that these columns exhibit a significant association with audience ratings.

### 5.3 Insights from Pearson Correlation test

The Pearson correlation analysis was conducted to examine the relationships between various film attributes and audience ratings. The results, as presented in Table 5.3.1, reveal significant insights into the factors influencing audience engagement.

**Table 5.2 Results from Pearson Correlation Test**

Variable	r	p_value
popularity	0.2756	0
vote_count	0.3189	0
budget_in_mn	0.0838	0
No_of_Genres	0.0459	0.0015
No_of_keywords	0.2819	0
overview_word_count	-0.0036	0.8032
revenue	0.198	0
runtime	0.3488	0

Notably, the variables 'overview\_word\_count' ( $r = -0.0036$ ,  $p = 0.8032$ ) showed a negligible negative correlation, indicating that the length of the movie overview does not significantly influence audience ratings.

These results affirm the influence of factors such as film popularity, vote count, budget, number of genres, number of keywords, revenue, and runtime on audience ratings. The statistically significant positive correlations suggest that movies with higher popularity, larger budgets, and longer runtimes, among other attributes, tend to receive higher audience ratings. However, the negligible correlation for 'overview\_word\_count' implies that the textual length of movie overviews does not play a substantial role in shaping audience perceptions. These findings contribute valuable insights for filmmakers and industry professionals seeking to understand the determinants of audience engagement in the film industry.

### 5.4 Evaluating Recommendations

It was assessed how similar their recommended movies were based on cosine similarity scores. This method allows us to quantify the similarity between the recommendation lists for each movie, providing insights into the effectiveness of the recommendation algorithm.

The selected four movies were subjected to the cosine similarity calculation, comparing the vectors representing their recommended movie lists. The cosine similarity scores range from 0 to 1, with higher values indicating greater similarity. A high cosine similarity suggests that the recommended movies for two films share commonalities, potentially indicating a robust recommendation system.

All the tags and feature combinations are explained in chapter 4.9. Below are the results for all the feature combinations.

**Table 5.3 Recommending Movies for avatar**

TAG	Title	Score	TAG	Title	Score
1	Falcon Rising	0.234822	2	Falcon Rising	0.217152
	Aliens	0.229564		Aliens	0.214149
	Aliens vs Predator: Requiem	0.224796		Battle: Los Angeles	0.187805
	Meet Dave	0.204099		Apollo 18	0.17316
	Battle: Los Angeles	0.198011		Meet Dave	0.162454
TAG	Title	Score	TAG	Title	Score
3	Falcon Rising	0.229226	4	Falcon Rising	0.231814
	Aliens	0.223156		Aliens	0.210539
	Battle: Los Angeles	0.197344		Battle: Los Angeles	0.200523
	Apollo 18	0.177361		Aliens vs Predator: Requiem	0.193071
	Meet Dave	0.17534		Apollo 18	0.180826
TAG	Title	Score	TAG	Title	Score
5	Star Trek Into Darkness	0.243402	6	Apollo 18	0.250244
	A Monster in Paris	0.238914		Tears of the Sun	0.171047
	The Blue Room	0.196232		The Adventures of Pluto Nash	0.160715
	Stranded	0.167489		Bucky Larson: Born to Be a Star	0.136665
	The Astronaut's Wife	0.158843		Life During Wartime	0.134997
TAG	Title	Score			
7	Aliens	0.253438			
	Falcon Rising	0.219417			
	Aliens vs Predator: Requiem	0.213516			
	Meet Dave	0.190659			
	Battle: Los Angeles	0.185273			

Table 5.3 shows the results of recommendations and Similarity scores generated using cosine similarity for the movie ‘Avatar’. Below are few observations.

- Tag7, featuring a combination of features selected from Chi-square and Pearson tests, claims the top spot on the list. In contrast, Tag6, relying solely on the movie overview, secures the second-best position.
- Despite the high scores, the movies recommended by Tag6 stand out as distinctly different from those suggested by other models. In contrast, the remaining tags demonstrate similarities in the recommended movies.
- Tag1, incorporating all features selected from the Chi-square test, occupies the fourth position, closely following Tag5, which secures the third spot in the ranking.

**Table 5.4 Recommended Movies & Similarity score for the Movie Spider-Man 3**

TA G	Title	Similiarity Score	TA G	Title	Similiarity Score
1	Spider-Man	0.553914	2	Spider-Man	0.474276
	Spider-Man 2	0.49759		Spider-Man 2	0.425032
	Arachnophobia	0.326598		Arachnophobia	0.357508
	The Amazing Spider-Man 2	0.311498		The Amazing Spider-Man	0.297067
	The Amazing Spider-Man	0.291256		The Amazing Spider-Man 2	0.294876
TA G	Title	Similiarity Score	TA G	Title	Similiarity Score
3	Spider-Man	0.480508	4	Spider-Man	0.512319
	Spider-Man 2	0.434106		Spider-Man 2	0.465573
	Arachnophobia	0.35376		Arachnophobia	0.338126
	The Amazing Spider-Man	0.30614		The Amazing Spider-Man	0.291952
	The Amazing Spider-Man 2	0.305174		The Amazing Spider-Man 2	0.287542
TA G	Title	Similiarity Score	TA G	Title	Similiarity Score
5	Spider-Man 2	0.373638	6	Spider-Man	0.360956
	Spider-Man	0.290028		Spider-Man 2	0.357898
	The One	0.288827		The Amazing Spider-Man	0.314955
	Two Lovers	0.269547		Arachnophobia	0.289726
	You, Me and Dupree	0.255087		The Amazing Spider-Man 2	0.26169

TA G	Title	Similarity Score
7	Spider-Man	0.541847
	Spider-Man 2	0.494721
	Arachnophobia	0.326543
	The Amazing Spider-Man 2	0.302411
	The Amazing Spider-Man	0.290747

Table 5.4 Demonstrates the recommended movies and similarity scores for the move “Spider-Man 3”. Below are the observations from the comparison.

- Nearly all models showcase strong performance, with Tag1 and Tag7 leading the list in terms of recommendation effectiveness respectively.
- Interestingly, each tag in the table displays similar recommendations, except for Tag5, which stands out with distinct movie suggestions.
- Tag5 and Tag6 are identified as the least performing models based on the similarity score. Despite the overall strong performance of the models, these two tags exhibit comparatively lower effectiveness in generating similar movie recommendations.

**Table 5.5 Recommended Movies for the Movie Pirates of the Caribbean: At World's End**

TA G	Title	Score	TA G	Title	Score
1	Pirates of the Caribbean: Dead Man's Chest	0.455293	2	Pirates of the Caribbean: Dead Man's Chest	0.241139
	Pirates of the Caribbean: The Curse of the Bla...	0.314338		Life of Pi	0.19318
	Pirates of the Caribbean: On Stranger Tides	0.225578		20,000 Leagues Under the Sea	0.185548
	20,000 Leagues Under the Sea	0.172407		Pirates of the Caribbean: On Stranger Tides	0.173371
	Life of Pi	0.156201		The Pirates! In an Adventure with Scientists!	0.168914
TA G	Title	Score	TA G	Title	Score
3	Pirates of the Caribbean: Dead Man's Chest	0.264335	4	Pirates of the Caribbean: Dead Man's Chest	0.416063
	Life of Pi	0.204		Pirates of the Caribbean: The Curse of the Bla...	0.246766
	Pirates of the Caribbean: On Stranger Tides	0.195548		Pirates of the Caribbean: On Stranger Tides	0.203924
	20,000 Leagues Under the Sea	0.190535		Life of Pi	0.192958
	The Pirates! In an Adventure with Scientists!	0.170503		20,000 Leagues Under the Sea	0.173847



TA G	Title	Score	TA G	Title	Score
5	Pirates of the Caribbean: Dead Man's Chest	0.364204	6	What's Love Got to Do with It	0.222033
	Pirates of the Caribbean: The Curse of the Bla...	0.273381		My Blueberry Nights	0.190972
	Cutthroat Island	0.25899		The Chronicles of Narnia: The Voyage of the Da...	0.179136
	Two Lovers	0.223862		The Descendants	0.174796
	You, Me and Dupree	0.211852		Disturbia	0.168073
TA G	Title	Score			
7	Pirates of the Caribbean: Dead Man's Chest	0.429867			
	Pirates of the Caribbean: The Curse of the Bla...	0.280256			
	Pirates of the Caribbean: On Stranger Tides	0.240155			
	20,000 Leagues Under the Sea	0.164164			
	Life of Pi	0.152353			

Table 5.5 shows the 5 Recommended Movies for the Movie Pirates of the Caribbean: At World's End.

- Once again, Tag1 and 7 emerge as the top-performing tags, showcasing consistent effectiveness in generating recommendations.
- On the other hand, Tag6 and 3 are identified as the least performing tags, indicating lower success in delivering relevant and similar movie suggestions.
- Particularly, Tag6 stands out by presenting a completely separate set of recommended movies, contributing to its lower similarity score compared to other tags.

**Table 5.6 Recommended Movies for the Movie Tangled**

TAG	Title	Score	TAG	Title	Score
1	Out of Inferno	0.229737	2	Gulliver's Travels	0.256013
	Gulliver's Travels	0.195604		Out of Inferno	0.217298
	Ant-Man	0.171832		The Walk	0.176592
	Man on Wire	0.1683		Ant-Man	0.172109
	The Walk	0.157372		TRON: Legacy	0.155238
TAG	Title	Score	TAG	Title	Score
3	Gulliver's Travels	0.253737	4	Out of Inferno	0.247771
	Out of Inferno	0.220417		Gulliver's Travels	0.21635
	The Walk	0.173589		Ant-Man	0.175579
	Ant-Man	0.173074		The Walk	0.172469

	TRON: Legacy	0.153294		Man on Wire	0.167494
<b>TAG</b>	<b>Title</b>	<b>Score</b>	<b>TAG</b>	<b>Title</b>	<b>Score</b>
5	The Princess and the Frog	0.353773	1	Gulliver's Travels	0.212124
	Flicka	0.273351		Out of Inferno	0.192199
	Enchanted	0.270699		TRON: Legacy	0.19004
	Fame	0.258633		The Walk	0.174689
				The Lord of the Rings: The Two Towers	0.161028
<b>TAG</b>	<b>Title</b>	<b>Score</b>			
7	Out of Inferno	0.229837			
	Gulliver's Travels	0.189331			
	Ant-Man	0.164804			
	Man on Wire	0.162231			
	The Walk	0.149512			

Table 5.6 shows the recommendation for the movie Tangled.

- In a surprising turn of events, tag5 emerges as the best-performing tag, surpassing all others in recommending movies that resonate more effectively with users.
- Contrarily, tag1, which was previously considered a strong performer, is unexpectedly identified as the least performing tag among all the models. This unexpected result may warrant a closer examination of the features and dynamics incorporated in Tag 1, aiming to understand the factors contributing to its comparatively lower performance.

## 5.5 Evaluating tags with Added weights

Selected Tag and process of adding weight to each column is discussed in chapter 4.10.

Again Avatar, Spider-Man-3 and Tangled are the three movies we are going to evaluate

**Table 5.7 Tag1 with added weights for the movie Avatar**

Added 10 weights to Overview			Added 10 weights to Keywords	
Title	Score		Title	Score
Apollo 18	0.221989		A Monster in Paris	0.24006
Tears of the Sun	0.168576		Star Trek Into Darkness	0.226196
Dolphin Tale 2	0.149704		The Blue Room	0.210632
Life During Wartime	0.146193		Cargo	0.191964

The Adventures of Pluto Nash	0.139951		Aliens	0.172603
Added 10 weights to cast			Added 10 weights to Director	
<b>Title</b>	<b>Score</b>		<b>Title</b>	<b>Score</b>
Backmask	0.47467		Aliens	0.862454
Gods and Generals	0.339771		True Lies	0.846721
Shadow Conspiracy	0.27623		The Abyss	0.842505
Clash of the Titans	0.273395		Terminator 2: Judgment Day	0.829105
Wrath of the Titans	0.272414		The Terminator	0.815264

Added 10 weights to Genres			Added 10 weights to Production	
<b>Title</b>	<b>Score</b>		<b>Title</b>	<b>Score</b>
Jupiter Ascending	0.818902		True Lies	0.67103
Beastmaster 2: Through the Portal of Time	0.787962		Titanic	0.588823
Small Soldiers	0.775771		The Abyss	0.521708
Man of Steel	0.77009		Live Free or Die Hard	0.46726
The Wolverine	0.7634		Fantastic 4: Rise of the Silver Surfer	0.440731

Added 10 weights to languages	
<b>Title</b>	<b>Score</b>
Morvern Callar	0.700762
The Ten	0.691219
Instructions Not Included	0.691081
Mi America	0.687505
Hands of Stone	0.683496

**Table 5.8 Tag1 with added weights for the movie Spider-Man 3**

Added 10 weights to Overview		Added 10 weights to Keywords	
<b>Title</b>	<b>Score</b>	<b>Title</b>	<b>Score</b>
Spider-Man	0.377689	Spider-Man 2	0.436828
Spider-Man 2	0.36529	Spider-Man	0.317146
The Amazing Spider-Man	0.322087	Two Lovers	0.279928
Arachnophobia	0.294222	The One	0.277445
The Amazing Spider-Man 2	0.269742	You, Me and Dupree	0.260012

Added 10 weights to Cast		Added 10 weights to Director	
Title	Score	Title	Score
Spider-Man	0.739748	Spider-Man	0.89699
Spider-Man 2	0.719084	Spider-Man 2	0.87373
Flyboys	0.307386	The Quick and the Dead	0.807635
We Bought a Zoo	0.304024	The Evil Dead	0.806518
Labor Day	0.301475	Oz: The Great and Powerful	0.79484

Added 10 weights to Genres		Added 10 weights to Production	
Title	Score	Title	Score
Spider-Man 2	0.782856	Spider-Man 2	0.939975
Spider-Man	0.716815	Spider-Man	0.691102
Krull	0.706654	The Amazing Spider-Man 2	0.638798
The Amazing Spider-Man 2	0.702198	The Amazing Spider-Man	0.570096
The Amazing Spider-Man	0.666979	The Butler	0.430725
Added 10 weights to languages			
Title	Score		
Pink Narcissus	0.607907		
The House of Mirth	0.599088		
Nighthawks	0.585881		
Not Easily Broken	0.577682		
The Face of an Angel	0.559921		

The introduction of weights to the columns has proven to be a transformative strategy, notably enhancing the similarity scores. In Table 5.7, the published results display the recommended movies and the similarity scores for the movie 'Avatar' after assigning a weight of 10 to each column. Subsequently, Table 5.8 provides detailed insights into the movie 'Spider-man 3.'

In comparison to the various tags tested in Section 5.4, the incorporation of weights notably amplifies the similarity index for recommended movies. Impressively, assigning 10 weights to the 'directors' column yields the highest score among all tested configurations. Following closely, the addition of 10 weights to 'Genres' also produces commendable results.

However, it's noteworthy that assigning weights to 'Overview' or 'Keywords' fails to contribute substantial value when compared to the impact observed with other features. This observation suggests that for the current dataset and recommendation system, emphasizing the director or

genre elements has a more pronounced effect on enhancing movie recommendations, while 'Overview' and 'Keywords' may not significantly influence the similarity scores.

## 5.6 Interpretations from EDA

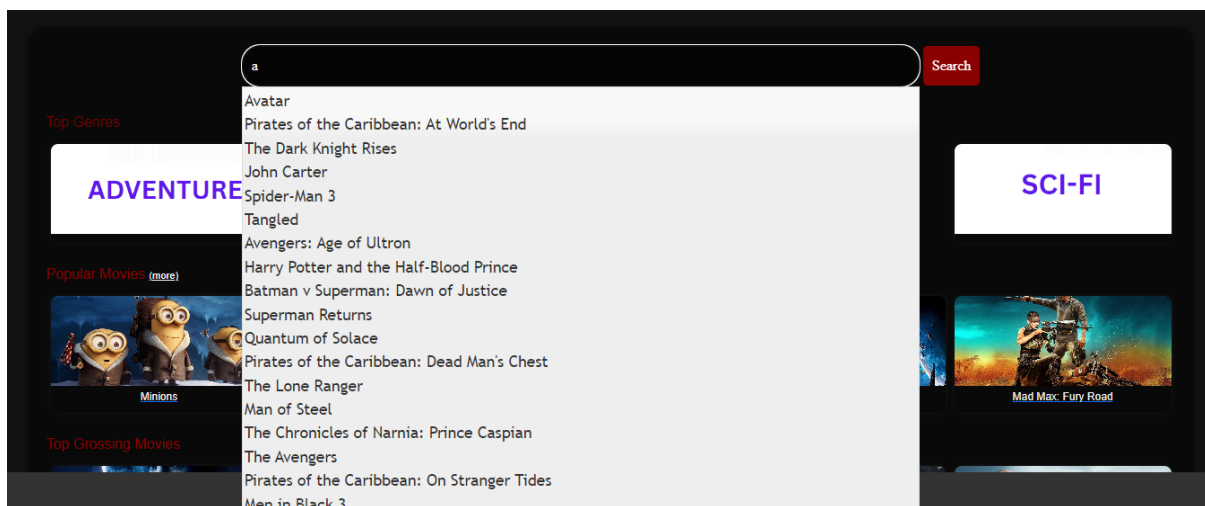
These key observations derived from Exploratory Data Analysis have significantly influenced the design of the homepage for the OTT platform deployed on AWS. The insights below were instrumental in shaping the user interface and content curation:

- **Genre Prioritization:** The top four genres in the dataset with most number of movies, namely Drama, Comedy, Thriller, and Action, but Adventure, Animation, Fantasy and science fiction have been identified as the most popular among users. This information has guided the platform to prominently feature content from these genres on the homepage, ensuring alignment with user preferences.
- **Highlighting Popular Movies:** Leveraging data on top-rated movies from the TMDB dataset, such as 'Minions,' 'Interstellar,' 'Dead pool,', 'Guardians of the Galaxy,' and 'Mad Max: Fury Road,' the platform strategically showcases these popular titles. This approach aims to captivate user interest by presenting well-received and widely acclaimed films.
- **Showcasing Top Grossing Movies:** The identification of top-grossing movies, including 'Avatar,' 'Titanic,' 'The Avengers,' 'Jurassic World,' and 'Furious 7,' has guided the platform to feature financially successful titles prominently. This strategic placement aims to attract users by offering blockbuster content.
- **Highlighting Highly Rated Movies:** The weighted average ratings from the dataset have identified 'The Shawshank Redemption,' 'The Godfather,' 'Fight Club,' 'Pulp Fiction,' and 'The Dark Knight' as top-rated movies. The platform utilizes this information to showcase critically acclaimed and highly rated films, creating an enriched viewing experience for users.
- These observations serve as a foundational framework for the homepage design, ensuring that the OTT platform aligns with user preferences, showcases popular and successful content, and offers a curated selection of top-rated movies.

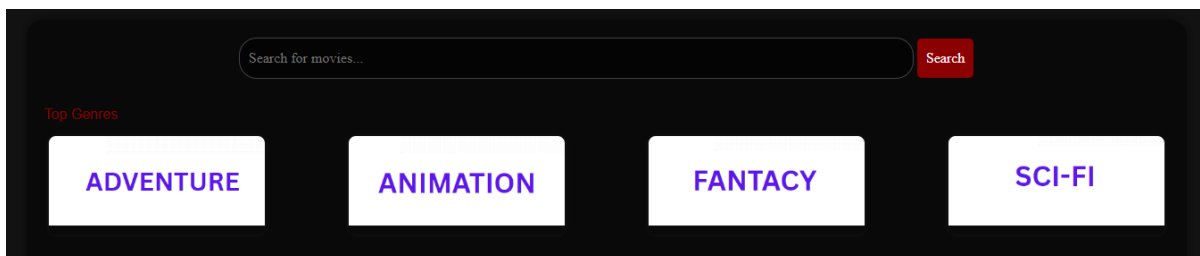
## 5.7 Website Features and Recommendation Options

Website features are completely based on the insights from EDA. EDA has played a pivotal role in understanding user behaviour, preferences, and engagement patterns. By thoroughly examining the data, we have been able to extract meaningful information that directly informs the design and functionality of our website. When a user logged in for the first time there is no user profile to recommend movies to the user. An interesting challenge arises when a user logs in for the first time. At this point, there is no existing user profile to draw from for personalized movie recommendations.

- Firstly, Figure 5.1 shows the search mechanism was implemented using drop down recommendations of movies for user to type and search for a movie.

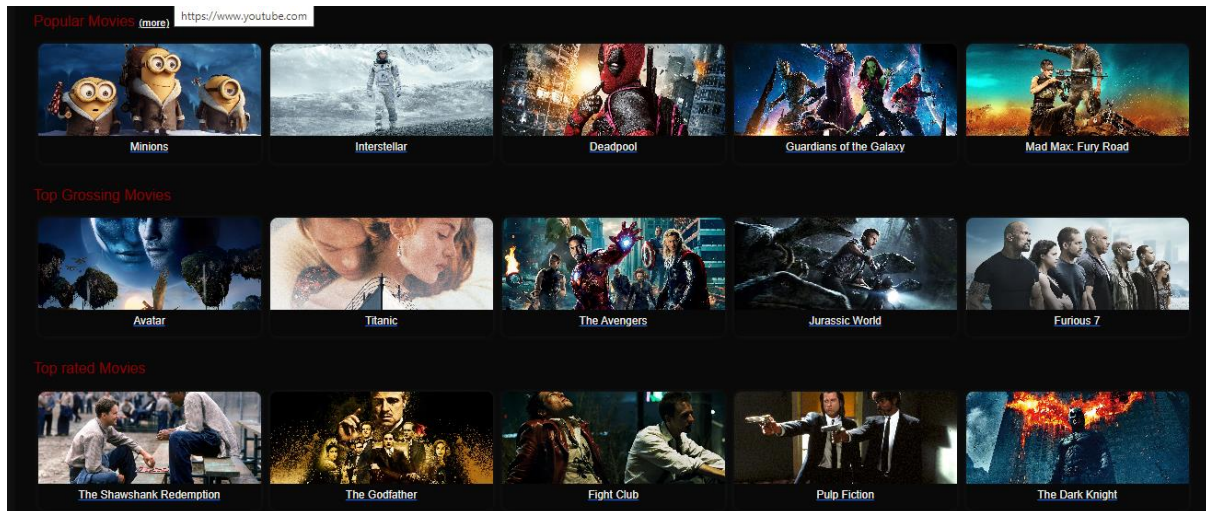


**Figure 5.1 Search Bar with Dropdown suggestions**



**Figure 5.2 Options Displaying Top Genres in the Homepage**

- As highlighted in section 5.6, Figure 5.2 displays the top 4 Genres in the dataset which has movies with highest average popularity.
- Figure 5.3 shows the Popular movies, Top grossing movies and highly rated movie as second, third and fourth sections respectively, placed on the home page of a website



**Figure 5.3 Display of top 5 Popular, Top grossing and highly rated movies**



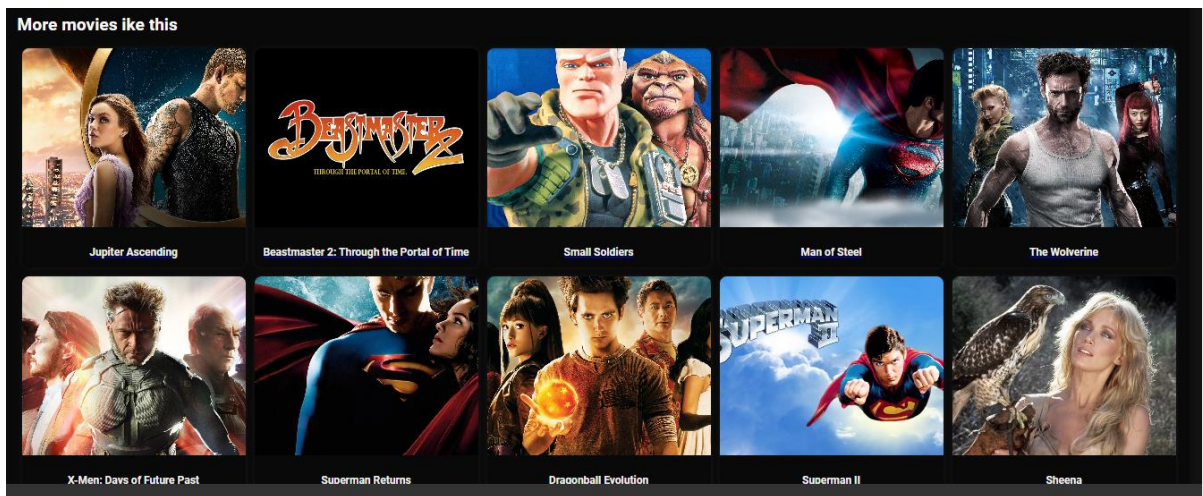
**Figure 5.4 Figure showing overview of Avatar movie**

- Upon selecting any movie, an overview page opens up, providing comprehensive details about the selected movie, including the movie poster. In Figure 5.4, you can

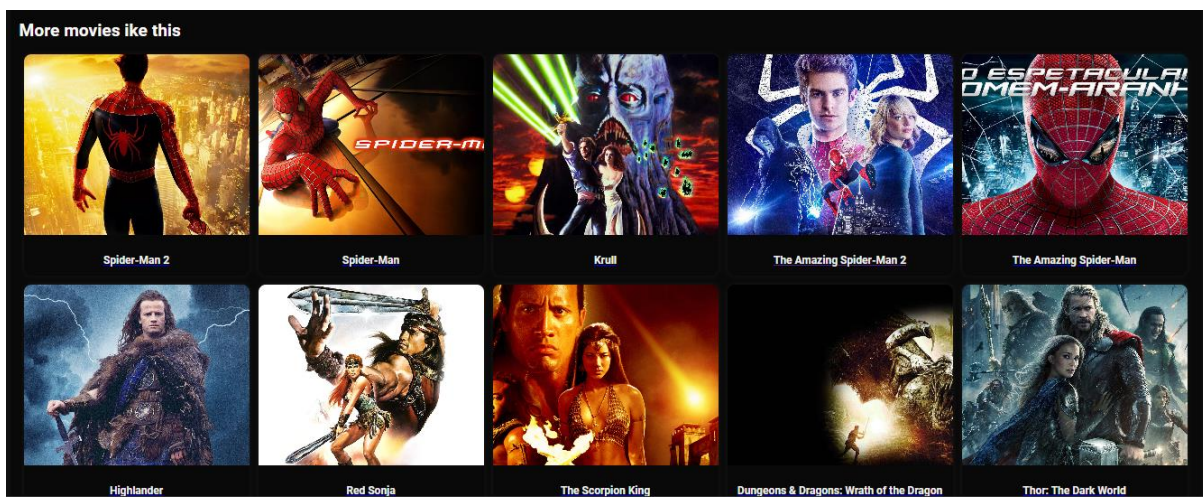


observe the overview page for the movie "Avatar." The details encompass release date, genre, director, cast, and a brief movie overview.

- Figures 5.5 and 5.6 showcase the top 10 recommended movies based on similarity scores for the movies "Avatar" and "Spider-Man 3," respectively. The tags utilized in the final model for recommendations include the incorporation of 10 weights added to the Genre column.



**Figure 5.5 Recommendation for the selection of a movie Avatar**



**Figure 5.6 Recommendations of a selection of Movie Spider-Man 3**



## **5.8 Summary**

In summary, this section explains the significance of employing Chi-square and Pearson correlation tests to identify features that strongly influence audience ratings for movies. The evaluation process involves assessing various tags, each representing a distinct combination of features. Through a thorough analysis, the best-performing tag is identified using cosine similarity. This section also evaluates the weights added to each feature in the movie dataset.

This evaluation process results in the selection of a tag that incorporates the most influential features, ultimately contributing to the creation of an optimal model for movie recommendations. This carefully curated model is poised to be deployed on an end-to-end OTT website hosted on AWS. By leveraging the insights gained from EDA and model evaluations, the platform is strategically designed to enhance the overall user experience by offering personalized and compelling movie recommendations.

## **Chapter 6: Conclusions and Recommendations**

### **6.1 Introduction**

This Chapter covers the key findings from the content based recommendation system, focusing on key insights from the implementation and evaluation process. Through a comprehensive review of achieved objectives and a discussion of system limitations, this chapter will explain the main aim of this research and recommendation for the future work.

### **6.2 Discussion and Conclusion**

This study Aimed to contribute to the specific objectives, below are some conclusions drawn:

#### **6.2.1 Feature Selection Insights:**

The utilization of Chi-square and Pearson-correlation tests for feature selection provided valuable insights into audience interests, particularly in relation to movie ratings. This selection of features might be helpful to get insights on selecting features based on audience interest.

#### **6.2.2 Comparative Analysis of Feature Combinations:**

The content-based filtering model incorporating tags derived from Chi-square and Pearson test have emerged as the most effective when overall similarity scores were compared with other feature combinations. But There is no conclusive evidence from this study that these techniques are helpful for creating tags. Feature tag that used only overview has shown good similarity but the movies recommended are very different from other models.

#### **6.2.3 Weighted Tags**

Adding weights to the features before creating tags outperformed the performance of tag created using features selected by Chi-square test in terms of similarity scores by huge margin, underscoring the nuanced nature of user preferences and the importance of assigning weights to tags. This step served as a fine tuned model of our recommendations based on cosine similarity. Best performing tags are when 10 weights added to Cast, Director and Genre respectively. However, we considered weight added to Genre as best model, mitigating the risk of recommending movies solely based on a single director or actor

#### **6.2.4 Role of EDA**

Exploratory Data Analysis (EDA) played a pivotal role in system development, offering actionable insights that streamlined the design of the OTT platform's homepage, enhancing user experience.

#### **6.2.5 Real-world Applicability and Validation:**

The successful deployment of the recommendation system on AWS validated its real-world applicability, with observed recommendations aligning closely with those generated during testing on Jupyter notebooks.

The convergence of feature selection methodologies, the significance of weighted tags, and the practical impact of EDA collectively reinforce the robustness of our content-based recommendation system.

### **6.3 Contribution to knowledge**

- This study approaches the route of understanding the feature selection methodologies in the context of content-based movie recommendation systems, offering valuable insights for designing user-centric OTT platforms.
- The exploration of weighted tags, especially the identification of the optimal weight distribution, represents a novel contribution and aid as a Fine tuning model for future recommendation systems
- This study underscores the practical integration of Exploratory Data Analysis, not only as a tool for system enhancement but also as a guide for designing user-friendly interfaces in OTT platforms.

### **6.4 Future Recommendations**

Future research could investigate the approach of fine-tuning the mechanisms for assigning weights to tags, exploring dynamic adjustments based on evolving user interactions and preferences to enhance the personalization of recommendations.

Considering the success of content-based approaches, future work could explore the integration of collaborative filtering techniques to develop a hybrid recommendation system, providing a more comprehensive and accurate user experience.

Conducting user-centric evaluation studies, such as surveys or user feedback sessions, could provide deeper insights into the user experience and further guide refinements to the recommendation system.

As the recommendation system scales, exploring optimization strategies for enhanced scalability and improved computational efficiency will be crucial to maintain optimal performance.

Emphasizing the importance of continuous model iteration and adaptation based on evolving user behaviours, preferences, and platform requirements will ensure the sustained relevance and effectiveness of the recommendation system over time.

## REFERENCES

- Albayati, A.N.K. and Ortakci, O.U.Y., (2022) Recommendation Systems on Twitter Data for Marketing Purposes using Content-Based Filtering. In: *HORA 2022 - 4th International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Proceedings*. pp.1–5.
- Ayesha, S., Hanif, M.K. and Talib, R., (2020) Overview and comparative study of dimensionality reduction techniques for high dimensional data. *Information Fusion*, 59, pp.44–58.
- Bahl, D., Kain, V., Sharma, A. and Sharma, M., (2020) A novel hybrid approach towards movie recommender systems. *Journal of Statistics and Management Systems*, 236, pp.1049–1058.
- Batmaz, Z., Yurekli, A., Bilge, A. and Kaleli, C., (2019) A review on deep learning for recommender systems: challenges and remedies. *Artificial Intelligence Review*, 52, pp.1–37.
- Benkessirat, S., Boustia, N. and Nachida, R., (2021) A new collaborative filtering approach based on game theory for recommendation systems. *Journal of web engineering*, 202, pp.303–326.
- Chen, P., Li, F. and Wu, C., (2021) Research on intrusion detection method based on Pearson correlation coefficient feature selection algorithm. In: *Journal of Physics: Conference Series*. p.12054.
- Christakou, C., Vrettos, S. and Stafylopatis, A., (2007) A hybrid movie recommender system based on neural networks. *International Journal on Artificial Intelligence Tools*, 1605, pp.771–792.

- Darban, Z.Z. and Valipour, M.H., (2022) GHRS: Graph-based hybrid recommendation system with application to movie recommendation. *Expert Systems with Applications*, 200, p.116850.
- Deldjoo, Y., Dacrema, M.F., Constantin, M.G., Eghbal-Zadeh, H., Cereda, S., Schedl, M., Ionescu, B. and Cremonesi, P., (2019) Movie genome: alleviating new item cold start in movie recommendation. *User Modeling and User-Adapted Interaction*, 29, pp.291–343.
- Dessi, D., Helaoui, R., Kumar, V., Recupero, D.R. and Riboni, D., (2021) TF-IDF vs word embeddings for morbidity identification in clinical notes: An initial study. *arXiv preprint arXiv:2105.09632*.
- Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J. and Yin, D., (2019) Graph neural networks for social recommendation. In: *The world wide web conference*. pp.417–426.
- Ferrari Dacrema, M., Cremonesi, P. and Jannach, D., (2019) Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In: *Proceedings of the 13th ACM conference on recommender systems*. pp.101–109.
- Furtado, F. and Singh, A., (2020) Movie recommendation system using machine learning. *International journal of research in industrial engineering*, 91, pp.84–98.
- Gunawardana, A. and Meek, C., (2009) A unified approach to building hybrid recommender systems. In: *Proceedings of the third ACM conference on Recommender systems*. pp.117–124.
- Havolli, A., Maraj, A. and Fetahu, L., (2022) Building a content-based recommendation engine model using Adamic Adar Measure; A Netflix case study. In: *2022 11th Mediterranean Conference on Embedded Computing (MECO)*. pp.1–8.
- Herce-Zelaya, J., Porcel, C., Bernabé-Moreno, J., Tejeda-Lorente, A. and Herrera-Viedma, E., (2020) New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests. *Information Sciences*, 536, pp.156–170.
- Idris, N., Foozy, C.F.M. and Shamala, P., (2020) A generic review of web technology: Django and flask. *International Journal of Advanced Science Computing and Engineering*, 21, pp.34–40.
- Javed, U., Shaukat, K., Hameed, I.A., Iqbal, F., Alam, T.M. and Luo, S., (2021) A Review of Content-Based and Context-Based Recommendation Systems. *International Journal of Emerging Technologies in Learning*, 163, pp.274–306.
- Kannikaklang, N., Wongthanavas, S. and Thamviset, W., (2022) A Hybrid Recommender System for Improving Rating Prediction of Movie Recommendation. In: *2022 19th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. pp.1–6.
- Konstan, J. and Terveen, L., (2021) Human-centered recommender systems: Origins, advances, challenges, and opportunities. *AI Magazine*, 423, pp.31–42.
- Koren, Y., Rendle, S. and Bell, R., (2021) Advances in collaborative filtering. *Recommender systems handbook*, pp.91–142.
- Kumaar, H., Srikumaran, S., Veni, S. and others, (2022) Content-based Movie Recommender System Using Keywords and Plot Overview. In: *2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*. pp.49–53.
- Lops, P., Jannach, D., Musto, C., Bogers, T. and Koolen, M., (2019) Trends in content-based recommendation: Preface to the special issue on Recommender systems based on rich item descriptions. *User Modeling and User-Adapted Interaction*, 29, pp.239–249.

- Mohamed, M.H., Khafagy, M.H. and Ibrahim, M.H., (2019) Recommender Systems Challenges and Solutions Survey. In: *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*. pp.149–155.
- Pintas, J.T., Fernandes, L.A.F. and Garcia, A.C.B., (2021) Feature selection methods for text classification: a systematic literature review. *Artificial Intelligence Review*, 548, pp.6149–6200.
- Pradeep, N., Rao Mangalore, K.K., Rajpal, B., Prasad, N. and Shastri, R., (2020) Content based movie recommendation system. *International journal of research in industrial engineering*, 94, pp.337–348.
- Pujahari, A. and Sisodia, D.S., (2022) Item feature refinement using matrix factorization and boosted learning based user profile generation for content-based recommender systems. *Expert Systems with Applications*, 206, p.117849.
- Rahman, A. and Hossen, M.S., (2019) Sentiment analysis on movie review data using machine learning approach. In: *2019 international conference on bangla speech and language processing (ICBSLP)*. pp.1–4.
- Rendle, S., Krichene, W., Zhang, L. and Anderson, J., (2020) Neural collaborative filtering vs. matrix factorization revisited. In: *Proceedings of the 14th ACM Conference on Recommender Systems*. pp.240–248.
- Sahu, S., Kumar, R., Pathan, M.S., Shafi, J., Kumar, Y. and Ijaz, M.F., (2022) Movie Popularity and Target Audience Prediction Using the Content-Based Recommender System. *IEEE Access*, 10, pp.42030–42046.
- Sayassatov, D. and Cho, N., (2020) The analysis of association between learning styles and a model of IoT-based education: Chi-square test for association. *Journal of Information Technology Applications and Management*, 273, pp.19–36.
- Silveira, T., Zhang, M., Lin, X., Liu, Y. and Ma, S., (2019) How good your recommender system is? A survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics*, 10, pp.813–831.
- Singh, A.K. and Shashi, M., (2019) Vectorization of text documents for identifying unifiable news articles. *International Journal of Advanced Computer Science and Applications*, 107.
- Singh, P. and Singh, P., (2019) Natural language processing. *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*, pp.191–218.
- Singla, R., Gupta, S., Gupta, A. and Vishwakarma, D.K., (2020) FLEX: A content based movie recommender. In: *2020 International Conference for Emerging Technology, INCET 2020*. pp.1–4.
- Walek, B. and Fojtik, V., (2020) A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, 158, p.113452.
- Wu, S., Sun, F., Zhang, W., Xie, X. and Cui, B., (2022) Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 555, pp.1–37.
- Xin, X., He, X., Zhang, Y., Zhang, Y. and Jose, J., (2019) Relational collaborative filtering: Modeling multiple item relations for recommendation. In: *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. pp.125–134.
- Xue, F., He, X., Wang, X., Xu, J., Liu, K. and Hong, R., (2019) Deep item-based collaborative filtering for top-n recommendation. *ACM Transactions on Information Systems (TOIS)*, 373, pp.1–25.
- Yogish, D., Manjunath, T.N. and Hegadi, R.S., (2019) Review on natural language processing trends and techniques using NLTK. In: *Recent Trends in Image Processing and Pattern Recognition: Second International Conference, RTIP2R 2018, Solapur, India, December 21–22, 2018, Revised Selected Papers, Part III 2*. pp.589–606.

## **APPENDIX A: RESEARCH PROPOSAL**

To build a Content based Movie Recommendation system based on Movie tags

S NAGACHAITANYA

## Research Proposal

SEPTEMBER 2023



## **Abstract**

In the rapidly evolving landscape of the movie industry, new organizations face challenges in providing personalized movie recommendations to users due to limited initial user profiles and item ratings. This research presents a comprehensive approach to address this challenge by developing an optimized content-based movie recommendation system. By featuring popular movies on the home page of a website and implementing a search bar for users to input movie names, the project aims to tackle the 'cold start' problem. Furthermore, the research explores the impact of different feature combinations on recommendation accuracy and deploys the system on a user-friendly website interface.

We are employing content-based filtering by utilizing Term Frequency-Inverse Document Frequency (TF-IDF) vectorization and recommending movies to users using cosine similarity considering their choices made on the website. This study follows a structured methodology, including feature analysis, model development, optimization, and website integration. Its aim is to offer valuable insights and actionable recommendations to new organizations aiming to improve user engagement and satisfaction through personalized movie suggestions.

## 2. Contents

Abstract .....	2
1. Background .....	99
2. Problem Statement/Related work .....	100
3. Research Questions .....	102
4. Aim and Objectives.....	102
5. Significance of the Study .....	103
6. Scope of the Study .....	104
7. Research Methodology .....	106
8. Requirements Resources .....	109
9. Research Plan.....	109
References.....	110

### Table of Figures

<b>Figure 1: Content Based Recommendation system(Havolli et al., 2022) .....</b>	<b>19</b>
<b>Figure 2: Research Methodology.....</b>	<b>107</b>

### List of Tables

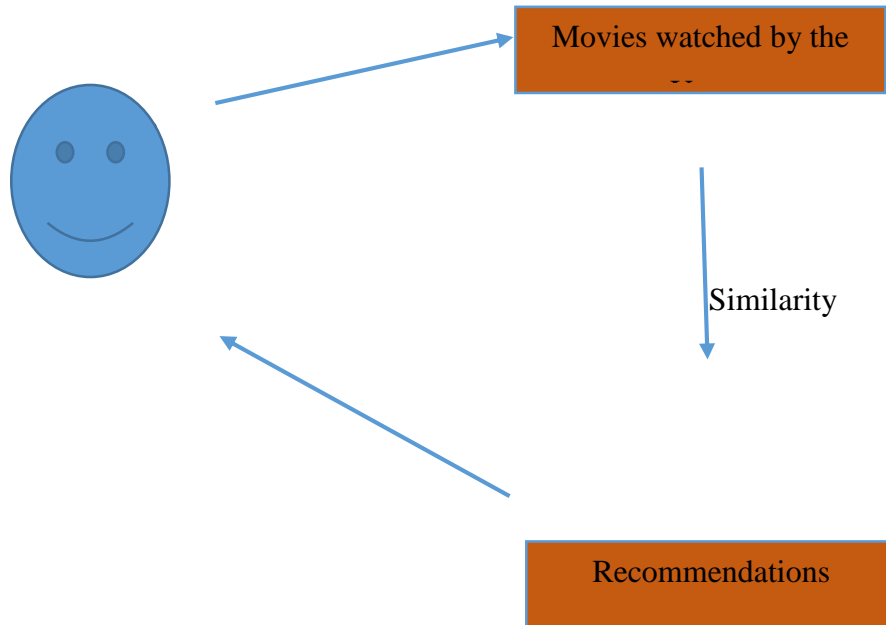
<b>Table 1: Research Plan .....</b>	<b>109</b>
-------------------------------------	------------

## 1. Background

Recommendation systems are everywhere in the Digital world and with emergence of Digital Video Libraries(OTTs) like Netflix, Amazon, they have employed well-tailored recommendations using tons of data to the target audience based on their taste (Kumaar et al., 2022). Traditional recommendation systems often rely on extensive user profile data and item ratings to provide accurate suggestions(Kannikaklang et al., 2022). However, new organizations entering the movie industry often lack this wealth of data, requiring innovative solutions to deliver effective recommendations from the outset. They can fetch data from twitter or other social networking platforms to inject some analysis(Albayati and Ortakci, 2022), but may not have extensive user profiling initially often these big companies have. There are different Hybrid recommendation systems for example (Darban and Valipour, 2022), (Kannikaklang et al., 2022) that have done well to overcome Cold start problem but this research embarks on the ambitious task of developing a content-based movie recommendation system (Figure 1) that not only overcomes the initial cold start problem using popular based recommendations initially but also delves into the intricate nuances of feature combinations that influence recommendation accuracy(Singla et al., 2020).

In this era of data-driven insights, our approach emphasizes the fusion of data science methodologies with user-centric design principles. By extracting meaningful features from the dataset and investigating how their combinations impact recommendation quality, we aspire to empower new organizations with the tools needed to navigate the challenges of the recommendation landscape. Moreover, we envision deploying the optimized recommendation system on a user-friendly website(Furtado and Singh, 2020),(Pradeep et al., 2020), ensuring that users can seamlessly access and benefit from personalized movie suggestions aligned with their preferences.

This research journey will traverse the realms of data pre-processing, model development, optimization, and user experience enhancement.



### **Content Based Recommendation system(Havolli et al., 2022)**

#### **2. Problem Statement/Related work**

This project seeks to develop a content-based movie recommendation system tailored for new organizations, addressing the initial cold start problem using Popular movie recommendation from the dataset and investigating the influence of varying feature combinations on recommendation accuracy. The system will then be optimized based on these findings and seamlessly integrated into a website, allowing users to receive popular recommendations initially and later personalized movie recommendations based on his preferences on the website and enhancing their movie-watching experience.

Content based recommendation systems are one of the popular recommendations systems in the field of study. A study (Havolli et al., 2022) used Netflix dataset and suggests Adamic-

Adar measures are effective for recommending new items. TF-IDF and Word2Vec models are employed to build a content-based recommendation system.

(Pujahari and Sisodia, 2022) Examines the concept of enhancing item features through matrix factorization. Additionally, it emphasizes that the scarcity and disparities in the datasets pose difficulties in collecting an ample amount of item feature data. It underscores the significance of the system adapting to correct inaccurate recommendations

(Kannikaklang et al., 2022) conducted research on the Movielens database to develop a hybrid recommendation system by integrating collaborative and content-based filtering. Their study incorporated models such as User K-NN, Item K-NN, Matrix Factorization, and Biased Matrix Factorization, comparing their performance using metrics like RMSE and MAE. Their findings indicated that Matrix Factorization, Biased Matrix Factorization, and Factor-wise Matrix Factorization are well-suited for collaborative filtering, while suggesting that Content-based filtering may not be optimal for large datasets.

(Darban and Valipour, 2022) Study leverages graph-based features and autoencoders and proposed a recommendation system. The primary objective of their study was to address the cold start problem by establishing relationships between users based on their similarities, represented as nodes in a similarity graph.

(Kumaar et al., 2022) utilized the IMDbpY Python library as their dataset and employed TF-IDF models with cosine similarity, Jaccard recommender models, and word-count cosine similarity models on the 'Keywords' and 'Plot Overview' features separately. Their research concluded that 'Keywords' were the most effective feature, with the Count Vectorizer model providing the best recommendations.

(Pradeep et al., 2020) attempted to merge certain features and employed cosine similarity to offer movie recommendations. They emphasized that their system does not consider other user profiles during the recommendation process.

(Javed et al., 2021) introduced a context-aware recommender system that filters items based on users' interests, in combination with a context-based recommender system for item recommendations. Their study concluded that ontology-based recommendation systems, when combined with other techniques, are widely employed for context-aware resource recommendations

(Albayati and Ortakci, 2022) suggested the application of content-based filtering for marketing purposes on specific social media platforms. Their study employed TF-IDF and cosine similarity for recommendations and found that the system successfully predicted target users for selling and providing services, achieving an accuracy rate of 86.2 based on users' tweets.

(Sahu et al., 2022) utilized IMDb and TMDB data to implement a multiclass classification model with an accuracy rate of 96.8%. Their research employed content-based recommendations to create a new dataset with predicted ratings and voting information. They presented a multiclass model using deep learning with CNN architecture.

### **3. Research Questions**

Question1: How does the selection and combination of different features from the dataset influence the accuracy of a content-based movie recommendation system?

Question 2: How can the optimized system be effectively deployed on a website?

### **4. Aim and Objectives**

The primary objective of this research is to establish a website that initially offers popular-based recommendations and subsequently provides users with item recommendations based on their preferences using content-based filtering.

The research objectives have been developed in alignment with the aim of this study and include the following:

- To recommend popular movies from the dataset using True Bayesian Estimation.
- To assess the accuracy of various models using combinations of different tags within the dataset.

- To deploy the final, accuracy-optimized model on the Heroku platform, employing the Django or Flask framework.

## **5. Significance of the Study**

- Our Study Addresses the "cold start" problem, which is a significant hurdle for new organizations lacking extensive user profiles and item ratings.
- Research provides a roadmap for new organizations to establish a strong foothold in the competitive movie industry. By leveraging content-based recommendation strategies and analysing feature combinations, these organizations can offer relevant and personalized content to users from the outset.
- Personalized recommendations enhance the user experience by helping users discover content aligned with their preferences. This study contributes to creating a more engaging and satisfying movie-watching experience for users, thereby increasing customer retention and loyalty.
- While collaborative filtering and hybrid methods are common, focus on content-based filtering and the examination of different feature combinations adds novelty to the approach. This can lead to innovative techniques that resonate particularly well with new organizations' resource constraints.
- Guiding new organizations in selecting meaningful features from their dataset to enhance recommendation accuracy. This is particularly valuable in domains with limited initial data, as it streamlines the decision-making process.
- The deployment of the optimized recommendation system on a website demonstrates the practical implementation of your research. It serves as a real-world application that showcases the feasibility of your approach and its potential impact on user engagement.
- Exploration of different feature combinations, model development, and optimization contributes to the academic understanding of content-based recommendation systems. Findings from this study can enrich the body of knowledge on recommendation strategies tailored to specific scenarios.

- The methodology can be adapted and extended to other domains beyond movies. Organizations facing similar challenges in various industries can draw insights from this study to develop personalized recommendation systems.

## 6. Scope of the Study

This Research focuses on the development, evaluation and deployment of an optimized content based movie recommendation system for new organizations entering the movie industry. Scope includes below aspects:

**Cold Start Problem:** The study addresses the initial cold start problem faced by new organizations with limited user profiles and item ratings. It concentrates on Popular movie recommendations from dataset using True Bayesian Estimate which is popularly being used in websites like IMDB for the users who have not yet interacted extensively with the system yet.

**Feature Combinations:** The research emphasizes various feature combinations, including movie attributes such as genres, directors, actors, keywords, plot overview etc. The scope includes investigating how different combinations influence the quality and accuracy of recommendations.

**Model Development and Optimization:** The study involves designing and implementing multiple content-based recommendation models, each utilizing a distinct feature combination. The scope extends to optimizing these models based on evaluation metrics to achieve higher recommendation accuracy.

**Website Integration:** The research includes the development of a website using Django/Flask interface that enables users to input their preferences and receive personalized movie recommendations. The study does not emphasize a robust website as this is just a prototype.

**Deployment:** The study culminates in the deployment of the optimized content-based recommendation system on a production environment accessible to users. The scope includes ensuring system functionality and performance on the website.

**Limitations:** The scope acknowledges that while content-based filtering is effective, hybrid methods that combine content-based and collaborative filtering are not within the primary



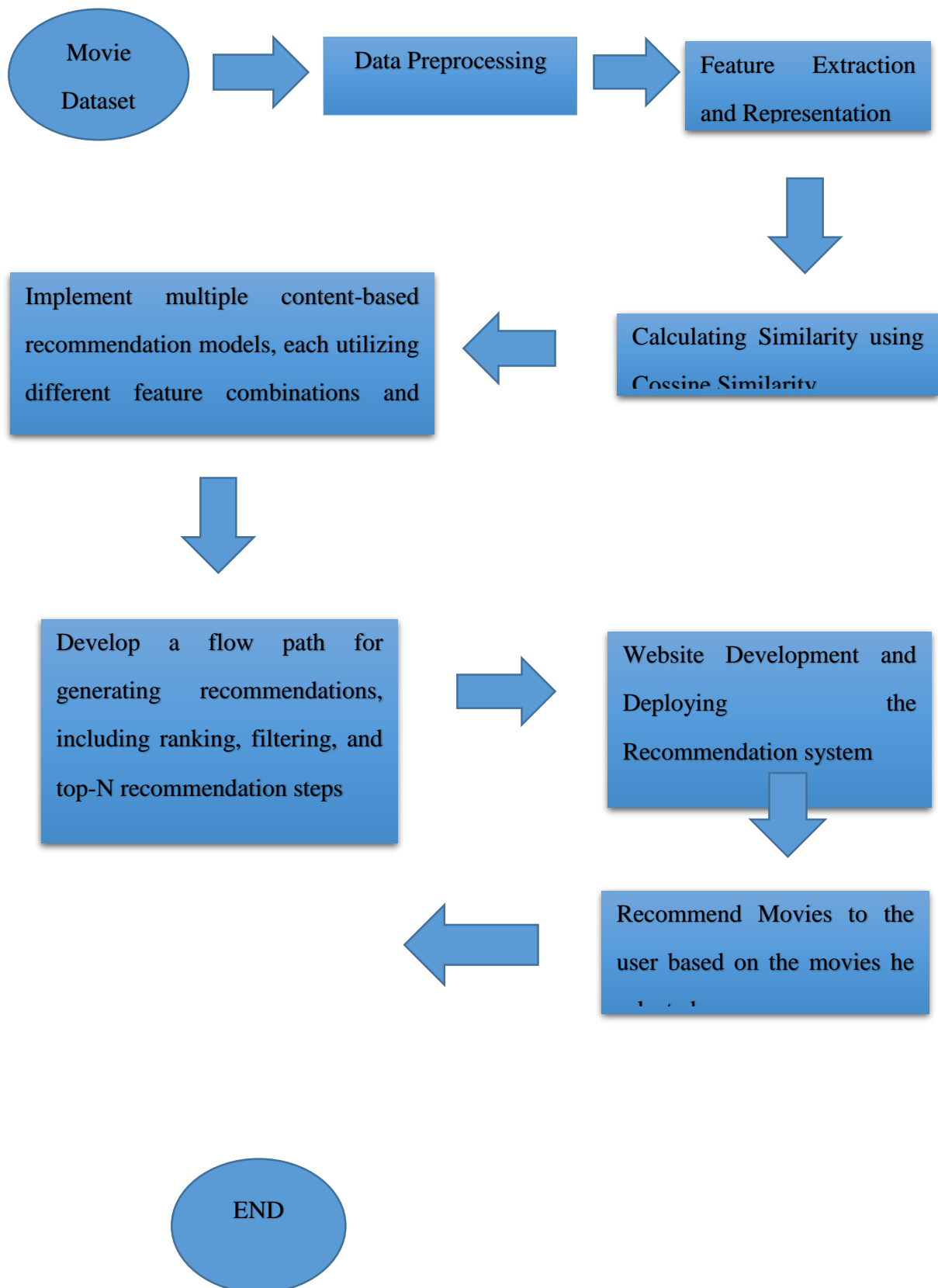
focus of this study. Additionally, while feature combinations are explored, the study does not delve into complex deep learning architectures.

**Dataset:** The study uses a dataset containing movie attributes such as genres, directors, actors, and textual descriptions. The scope does not involve data collection or creation of new datasets.

**Evaluation Metrics:** The study employs standard evaluation metrics such as precision, recall measure recommendation system accuracy. It does not introduce new evaluation metrics.

**Statistical Significance:** While the research aims to provide insights and practical solutions, it may not comprehensively cover statistical significance analysis due to time constraints.

## 7. Research Methodology



## **Research Methodology**

### **Data Collection:**

The selected dataset is sourced from Kaggle - specifically, the TMDB dataset. This dataset includes all the necessary columns, such as cast, crew, keywords, and plot overviews, which are essential for implementing our study.

### **Data Preprocessing:**

- Merge both Datasets and many features from the dataset needs to be extracted as they are in the dictionary format into text format.
- Preprocess the dataset by handling null and duplicate values.

### **Feature Selection and Engineering:**

- Extract key information from textual attributes, such as plot summaries, using NLTK techniques like tokenization, stop-word removal, and stemming or lemmatization.
- Identify relevant features for content-based filtering, including genres, directors, actors, and keywords.
- Textual Feature Representation: Convert the processed text into numerical representations, such as TF-IDF(Havolli et al., 2022) vectors or Word Embeddings (e.g., Word2Vec, GloVe).

### **Model Development:**

- Implement multiple content-based recommendation models, each utilizing different feature combinations.

- Cosine Similarity: Calculate the similarity score between the user selection and all other vectors in the dataset using Cosine Similarity.

### **Model Evaluation:**

- After Splitting the dataset into training and test sets for evaluation, Performance of each recommendation model will be accessed using standard evaluation metrics like precision, recall, and Mean Average Precision (MAP) etc.
- Compare the models to identify the most effective feature combinations.

### **Recommendation generation:**

- Rank the items based on their similarity scores with the user profile. Higher similarity indicates a stronger recommendation.
- Select the top N items with the highest similarity scores as the recommendations to present to the user.

### **Website Development:**

- Create a user-friendly website interface using Django, allowing users to input their preferences and interact with the recommendation system.
- To address the cold start problem, we will feature popular movies from the dataset on the home page using True Bayesian estimate weighted rating which popularly being used in IMDB.

$$\text{Weighted Rating (WV)} = (\text{votes} \div (\text{votes} + \text{min votes})) \times R + (\text{min votes} \div (\text{votes} + \text{min votes})) \times C$$

R = Mean of Movie ratings

Votes = Total no. votes for the movie

min votes = The minimum number of votes needed to secure a place in the Top 250 is presently set at 25,000.

C = The average (mean) vote across the entire report

- Include a search option at the top of the home page with dropdown recommendations. This feature encourages users to either type the movie name or select a popular movie to generate recommendations based on the previously developed model.
- Integrate the content-based recommendation system into the platform, allowing users to input preferences and interact.

## **8. Requirements Resources**

**Dataset: Using TMBD dataset from the kaggle for this study.**

<https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata>

**Software:**

- **Python:** Commonly used language for Machine learning and Web development.
- **Data processing libraries:** Pandas, Numpy,
- **Machine learning Libraries:** Scikit-learn, TF-IDF vectorization, cosine similarity.
- **Web development framework:** Django/Flask
- **Visualization libraries:** Matplotlib, seaborn
- **NLP tools:** word embeddings (Word2Vec/GloVe), stop-word lists, or stemming/lemmatization tools
- **Jupyter Notebook/Google colab :** For Prepressing the data
- **Visual Studio code:** for deploying the mode

## **9. Research Plan**

### **Research Plan**

# Content based Recommendation system

Project Planner

Period Highlight:

1

Plan Duration

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE	PERIODS							
						1	2	3	4	5	6	7	8
Literature search	2	4	2	4	100%								
Literature review	4	4	4	4	100%								
Investigate & Evaluate Recommender systems	2	3	1	3	100%								
Get Movie database	1	1	1	1	100%								
Feature extraction	4	4	0	0	0%								
Evaluatig different feature combinations on the dataset.	3	3	0	0	0%								
Generating recommendations using Top N using best features	1	1	0	0	0%								
Website Development	2	2	0	0	0%								
Deploying Popular and Content based recommendations in to website	1	1	0	0	0%								
Complete report	4	4	0	0	0%								

## References

Albayati, A.N.K. and Ortakci, O.U.Y., (2022) Recommendation Systems on Twitter Data for Marketing Purposes using Content-Based Filtering. In: *HORA 2022 - 4th International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Proceedings*. pp.1–5.

Ayesha, S., Hanif, M.K. and Talib, R., (2020) Overview and comparative study of dimensionality reduction techniques for high dimensional data. *Information Fusion*, 59, pp.44–58.

Bahl, D., Kain, V., Sharma, A. and Sharma, M., (2020) A novel hybrid approach towards movie recommender systems. *Journal of Statistics and Management Systems*, 236, pp.1049–1058.

Batmaz, Z., Yurekli, A., Bilge, A. and Kaleli, C., (2019) A review on deep learning for recommender systems: challenges and remedies. *Artificial Intelligence Review*, 52, pp.1–37.

Benkessirat, S., Boustia, N. and Nachida, R., (2021) A new collaborative filtering approach based on game theory for recommendation systems. *Journal of web engineering*, 202, pp.303–326.

Chen, P., Li, F. and Wu, C., (2021) Research on intrusion detection method based on Pearson correlation coefficient feature selection algorithm. In: *Journal of Physics: Conference Series*. p.12054.

Christakou, C., Vrettos, S. and Stafylopatis, A., (2007) A hybrid movie recommender system based on neural networks. *International Journal on Artificial Intelligence Tools*, 1605, pp.771–792.

Darban, Z.Z. and Valipour, M.H., (2022) GHRS: Graph-based hybrid recommendation system with application to movie recommendation. *Expert Systems with Applications*, 200, p.116850.

Deldjoo, Y., Dacrema, M.F., Constantin, M.G., Eghbal-Zadeh, H., Cereda, S., Schedl, M., Ionescu, B. and Cremonesi, P.,

- (2019) Movie genome: alleviating new item cold start in movie recommendation. *User Modeling and User-Adapted Interaction*, 29, pp.291–343.
- Dessi, D., Helaoui, R., Kumar, V., Recupero, D.R. and Riboni, D., (2021) TF-IDF vs word embeddings for morbidity identification in clinical notes: An initial study. *arXiv preprint arXiv:2105.09632*.
- Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J. and Yin, D., (2019) Graph neural networks for social recommendation. In: *The world wide web conference*. pp.417–426.
- Ferrari Dacrema, M., Cremonesi, P. and Jannach, D., (2019) Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In: *Proceedings of the 13th ACM conference on recommender systems*. pp.101–109.
- Furtado, F. and Singh, A., (2020) Movie recommendation system using machine learning. *International journal of research in industrial engineering*, 91, pp.84–98.
- Gunawardana, A. and Meek, C., (2009) A unified approach to building hybrid recommender systems. In: *Proceedings of the third ACM conference on Recommender systems*. pp.117–124.
- Havolli, A., Maraj, A. and Fetahu, L., (2022) Building a content-based recommendation engine model using Adamic Adar Measure; A Netflix case study. In: *2022 11th Mediterranean Conference on Embedded Computing (MECO)*. pp.1–8.
- Herce-Zelaya, J., Porcel, C., Bernabé-Moreno, J., Tejeda-Lorente, A. and Herrera-Viedma, E., (2020) New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests. *Information Sciences*, 536, pp.156–170.
- Idris, N., Foozy, C.F.M. and Shamala, P., (2020) A generic review of web technology: Django and flask. *International Journal of Advanced Science Computing and Engineering*, 21, pp.34–40.
- Javed, U., Shaukat, K., Hameed, I.A., Iqbal, F., Alam, T.M. and Luo, S., (2021) A Review of Content-Based and Context-Based Recommendation Systems. *International Journal of Emerging Technologies in Learning*, 163, pp.274–306.
- Kannikaklang, N., Wongthanavas, S. and Thamviset, W., (2022) A Hybrid Recommender System for Improving Rating Prediction of Movie Recommendation. In: *2022 19th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. pp.1–6.
- Konstan, J. and Terveen, L., (2021) Human-centered recommender systems: Origins, advances, challenges, and opportunities. *AI Magazine*, 423, pp.31–42.
- Koren, Y., Rendle, S. and Bell, R., (2021) Advances in collaborative filtering. *Recommender systems handbook*, pp.91–142.
- Kumaar, H., Srikumaran, S., Veni, S. and others, (2022) Content-based Movie Recommender System Using Keywords and Plot Overview. In: *2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*. pp.49–53.
- Lops, P., Jannach, D., Musto, C., Bogers, T. and Koolen, M., (2019) Trends in content-based recommendation: Preface to the special issue on Recommender systems based on rich item descriptions. *User Modeling and User-Adapted Interaction*, 29, pp.239–249.
- Mohamed, M.H., Khafagy, M.H. and Ibrahim, M.H., (2019) Recommender Systems Challenges and Solutions Survey. In: *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)*. pp.149–155.
- Pintas, J.T., Fernandes, L.A.F. and Garcia, A.C.B., (2021) Feature selection methods for text classification: a systematic

literature review. *Artificial Intelligence Review*, 548, pp.6149–6200.

Pradeep, N., Rao Mangalore, K.K., Rajpal, B., Prasad, N. and Shastri, R., (2020) Content based movie recommendation system. *International journal of research in industrial engineering*, 94, pp.337–348.

Pujahari, A. and Sisodia, D.S., (2022) Item feature refinement using matrix factorization and boosted learning based user profile generation for content-based recommender systems. *Expert Systems with Applications*, 206, p.117849.

Rahman, A. and Hossen, M.S., (2019) Sentiment analysis on movie review data using machine learning approach. In: *2019 international conference on bangla speech and language processing (ICBSLP)*. pp.1–4.

Rendle, S., Krichene, W., Zhang, L. and Anderson, J., (2020) Neural collaborative filtering vs. matrix factorization revisited. In: *Proceedings of the 14th ACM Conference on Recommender Systems*. pp.240–248.

Sahu, S., Kumar, R., Pathan, M.S., Shafi, J., Kumar, Y. and Ijaz, M.F., (2022) Movie Popularity and Target Audience Prediction Using the Content-Based Recommender System. *IEEE Access*, 10, pp.42030–42046.

Sayassatov, D. and Cho, N., (2020) The analysis of association between learning styles and a model of IoT-based education: Chi-square test for association. *Journal of Information Technology Applications and Management*, 273, pp.19–36.

Silveira, T., Zhang, M., Lin, X., Liu, Y. and Ma, S., (2019) How good your recommender system is? A survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics*, 10, pp.813–831.

Singh, A.K. and Shashi, M., (2019) Vectorization of text documents for identifying unifiable news articles. *International Journal of Advanced Computer Science and Applications*, 107.

Singh, P. and Singh, P., (2019) Natural language processing. *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*, pp.191–218.

Singla, R., Gupta, S., Gupta, A. and Vishwakarma, D.K., (2020) FLEX: A content based movie recommender. In: *2020 International Conference for Emerging Technology, INCET 2020*. pp.1–4.

Walek, B. and Fojtik, V., (2020) A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, 158, p.113452.

Wu, S., Sun, F., Zhang, W., Xie, X. and Cui, B., (2022) Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 555, pp.1–37.

Xin, X., He, X., Zhang, Y., Zhang, Y. and Jose, J., (2019) Relational collaborative filtering: Modeling multiple item relations for recommendation. In: *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. pp.125–134.

Xue, F., He, X., Wang, X., Xu, J., Liu, K. and Hong, R., (2019) Deep item-based collaborative filtering for top-n recommendation. *ACM Transactions on Information Systems (TOIS)*, 373, pp.1–25.

Yogish, D., Manjunath, T.N. and Hegadi, R.S., (2019) Review on natural language processing trends and techniques using NLTK. In: *Recent Trends in Image Processing and Pattern Recognition: Second International Conference, RTIP2R 2018, Solapur, India, December 21--22, 2018, Revised Selected Papers, Part III 2*. pp.589–606.



## **APPENDIX B: Python Code for Content based recommendation system**

[https://github.com/ChaitanyaSimhadri/CBRS/blob/main/MS\\_IN\\_DS\\_LJMU\\_Movie\\_recommendation.ipynb](https://github.com/ChaitanyaSimhadri/CBRS/blob/main/MS_IN_DS_LJMU_Movie_recommendation.ipynb)