

MOVIE RECOMMENDATION USING UNSUPERVISED METHOD

BY

DIBYO CHAKMA

ID: 221-25-089

This Report Presented in Partial Fulfillment of the Requirements for
The Degree of Masters of Science in Computer Science and Engineering

Supervised By

Md. Sadekur Rahman

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

Md. Tarek Habib

Assistant Professor

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JANUARY 2023

APPROVAL

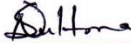
This Project/Thesis titled “**Movie Recommendation Using Unsupervised Method**”, submitted by **Diby Chakma**, ID No: **221-25-089** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfilment of the requirements for the degree of M.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 17-01-2023.

BOARD OF EXAMINERS



Dr. S M Aminul Haque, PhD
Associate Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology Daffodil
International University

Chairman



Ms. Naznin Sultana
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology Daffodil
International University

Internal Examiner



Mr. Md. Sadekur Rahman
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology Daffodil
International University

Internal Examiner



Dr. Mohammad Shorif Uddin, PhD
Professor
Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

We here by declare that, this thesis has been done by us under the supervision of **Md. Sadekur Rahman, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this thesis nor any part of this thesis has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Md. Sadekur Rahman
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:



Md. Tarek Habib
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:



Dibyo Chakma
ID: 221-25-089
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

At first, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final thesis successfully.

We really very grateful and wish our profound our indebtedness to **Md. Sadekur Rahman**, Assistant Professor, Department of CSE Daffodil International University, Dhaka. Deep knowledge and keen interest of our supervisor in the field of “Natural Language Processing and Machine Learning” to carry out this thesis. His scholarly guidance, patience, constructive criticism, motivation, constant and energetic supervision, continual encouragement, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this thesis.

We would also like to thank our co-supervisor **Md. Tarek Habib**, Assistant Professor, Department of CSE Daffodil International University, Dhaka. When we face any problem, she helped us with valuable ideas and suggestions. She motivated us and help us to complete this work.

We would like to express our heartiest gratitude to **Professor Dr. Touhid Bhuiyan, Professor & Head**, Department of CSE, for his motivation and appreciation. We are also very thankful to other faculty members and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we are very thankful to our parents and friends who were always motivate and criticize our work in a manner to improve our work. At least we thank all of them from the core of our heart.

ABSTRACT

Everyone in our hectic world is occupied with their personal and professional lives. There is a maximum number of people who can watch movies in theaters or read books while passing the time. These days, online services like Netflix, Amazon, and others are extremely popular. A recommendation mechanism is necessary for this entire platform. I read some recently released publications to better grasp the current state of recommender systems as well as their potential future directions. The purpose of a movie recommendation system is to make recommendations for films based on the interests of various users. The customers would benefit from time savings when looking for popular movies of their preferred genre. It makes predictions about what movies a user will enjoy based on a data collection and takes into account the qualities of the movies they have already loved. Using a combination of two or more attributes, recommendation systems can suggest movies. Various elements, such as the movie's genre, directors, and performers, are taken into account when developing a movie recommendation algorithm. The cast, keywords, crew, and genres have been used to construct the recommendation system in this work. I start by gathering a data collection from many websites, including Kaggle. Then, using a few strategies, clean up and delete unneeded data. I employ machine learning algorithms like the count vector and cosine similarity algorithm after cleaning datasets. In our datasets, all algorithms perform excellently.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
CHAPTER	
CHAPTER 1: INTRODUCTION	1-4
1.1 Introduction	01-02
1.2 Motivation	02
1.3 Rationale of the Study	02-03
1.4 Research Question	03
1.5 Expected Output	03
1.6 Project Management and Finance	04
1.7 Report Layout	04
CHAPTER 2: BACKGROUND	5-8
2.1 Terminologies	05
2.2 Related Works	05-06
2.3 Comparative Analysis and Summary	06-07
2.4 Scope of the Problem	07-08
2.5 Challenges	08
CHAPTER 3: RESEARCH METHODOLOGY	09-20
3.1 Research Subject and Instrumentation	09
3.2 Data Collection Procedure	09-10
3.3 Statistical Analysis	10-11
3.4 Proposed Methodology	11-20
3.5 Implementation Requirements	20
CHAPTER 4: EXPERIMENTAL RESULTS AND DISCUSSION	21-26
4.1 Experimental Setup	21-23
4.2 Experimental Results & Analysis	23-26
4.3 Discussion	26

CHAPTER 5: IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY	27-28
5.1 Impact on Society	27
5.2 Impact on Environment	27
5.3 Ethical Aspects	27
5.4 Sustainability Plan	28
CHAPTER 6: SUMMARY, CONCLUSION AND IMPLICATION FOR FUTURE RESEARCH	29-30
6.1 Summary of the Study	29
6.2 Conclusions	30
6.3 Implication for Further Study	30
REFERENCES	31-33
PLAGIARISM REPORT	34

LIST OF FIGURES

FIGURES	PAGE NO.
Figure 3.1: Proposed Methodology	12
Figure 3.2: Machine Learning Architecture	12
Figure 3.3: Cosine Similarity orientation (length)	18
Figure 3.4: Euclidean Formula	19
Figure 3.4: Cosine Similarity	19
Figure 4.1: Top Genre	22
Figure 4.2: Actor with highest appearance	22
Figure 4.3: Directors with highest movies	23
Figure 4.4: Recommendations similar to ‘Avatar’ movie using Algorithm 1 and Algorithm 2	24
Figure 4.5: Final Recommendations similar to ‘Avatar’ movie	25
Figure 4.6: Recommendations similar to ‘Titanic’ movies using Algorithm 1 and Algorithm 2	25
Figure 4.7: Final Recommendations similar to ‘Titanic’ movie	26

LIST OF TABLES

TABLE NO.	PAGE NO.
Table 2.1: Summary of Previous Research	07
Table 3.1: 1 st Movie Dataset for analysis	10
Table 3.2: 2 nd Movie Dataset for analysis	10
Table 3.3: Missing Values	17

CHAPTER 1

INTRODUCTION

1.1 Introduction

It has become far too simple for humans and technologies to converse through various methods. The development of science and technology has made living more comfortable than it was in earlier times. Emerging technologies like neural networks, image processing, computer languages, and neutrosophic shortest paths make goods more intelligent and self-healing based. The suggestion method in this publication is based on the actors, keywords, crew, and genres. These days, with the advancement of information technology, getting information via the internet is simple. It is challenging to use all of the online resources correctly because there are so many of them. For this reason, the recommendation system targets the user's interests and suggests things that are similar to what the user wants.

These days, smartphones have surpassed other technologies in importance for our daily lives. Due to this, ordinary activities like viewing movies and reading books are now readily accessible. Recommender systems have advanced significantly over the past ten years, especially in Internet-based applications, mobile devices, and other specialized appliances. The most well-known of them are recommendation systems created for information and entertainment-related areas.

The usage of recommender systems improves already useful applications for both users and service providers. Product recommendations, playlist creation, matchmaking, and many more tasks are all handled by recommender systems. User-item interaction and characteristic information are key components of recommender systems' operation. Characteristic data pertains to the user and the objects, whereas user-item interaction data pertains to ratings, the volume of purchases, user likes, and many other things. This allows collaborative filtering, content-based filtering, or hybrid filtering to be used to create the recommendation system.

Collaborative filtering: The system finds people with comparable preferences and uses their feedback to suggest the same to another user with a comparable interest. It has been incorporated into a variety of programs, including Spotify, Netflix, and YouTube.

Content-Based Filtering: Methods of content filtering are based on user characteristics. In contrast to collaborative approaches, it predicts the items based only on user information and entirely disregards user input.

Hybrid Approach: Combining collaborative filtering with content-based filtering or any other strategy is known as a hybrid approach. It improves the performance and accuracy of recommender systems.

With the help of this movie suggestion system, the user can be provided a list of films that most closely resemble the one they are now watching. Based on an assessment of plot similarity between the target movie's story and a sizable number of plots kept in a movie database, the system determines the list of comparable films.

1.2 Motivation

The internet is reachable from anywhere in the world. Therefore, anybody and everywhere can access it. You can access the internet if you know how to use it effectively. However, there are enough unwanted things on the internet. Our children and other future generations will be impacted by this. We must do this to safeguard it. We must suggest to kids films that are educational, cultural, and social in order to avoid exposing them terrible unrated movies. The quickest and most effective method for discovering quality movies is recommendation. Users find it challenging to research their favorite movies for business information as the number of movies, videos, and movie platforms rises along with the diversity of user tastes. A strong movie recommendation system can aid businesses in improving the movie-watching experience for users and luring more of them to their platform. Consequently, a method for suggesting movies needs to be developed. Everyone wants to spend their time on the internet. Additionally, they conduct impromptu searches for diverse music, film, and book genres. Giving our users the precise recommendations they desire is the aim of this thesis. Time will be saved, and they will learn important knowledge.

1.3 Rationale of the Study

Current tools are updated daily in the age of modern technology. Everyone likes to spend their time watching movies, TV shows, and reading books, if we're talking about how to pass the time. Simple method of phone access. Everyone wants access to all of their favorite television series or

films right away. We need recommendations because searching for movies of any genre requires random searching. Users with similar interests can be assisted by the movie recommendation. Artificial intelligence-based algorithms used for recommendations scan possibilities to compile a unique list of goods that are interesting and pertinent to a particular person. This will enable us to find what we're looking for quickly and generate as much interest as possible. We'll save time this way, and we'll be able to access our favorite movies automatically. Take, for instance, the case of a customer visiting a bakery to purchase his preferred cake, "X." The shopkeeper advises the customer to purchase cake "Y," which is produced with components identical to cake "X," because cake "X" has sadly run out. A recommendation is essentially meant to look for content that will interest a person.

1.4 Research Questions

During the research work some question occurs about this work. The main questions of our work in given below:

- How to collect and preprocess Movie data?
- How to extract features from Movie dataset?
- Do count vector and cosine-similarity work on this dataset for movie recommendation?

1.5 Expected Output

Although this is an experimental endeavor, my primary goal is to write a paper about it. Regarding movie recommendations, I came across a ton of comparable works. Numerous projects are recommended. There are a number different types of move recommendation, including context-based filtering, collaborative filtering, and hybrid filtering. On content-based filtering, I'm concentrating. This approach enables us to obtain accurate user-requested recommendations. This model provides them with accurate movies of the same kind. My main objective is to provide my users with a wide selection of movies.

- Published one or more papers on International Conference
- Make a model Movie Recommendation
- Our main expected output is our system can detect what the user wants

1.6 Project Management and Finance

I have borne all the expenses for my research data collection.

1.8 Research Layout

In our report we have total 6 chapters

- In Chapter 1 we mention our whole research work's outline and divided this chapter into multiple subchapters. For example, introduction, motivation, rational of the study, research question and expected output of our project.
- In Chapter 2 we have discussed about the previous work on Movie Recommendation, the scope of the problem and challenges in this work.
- In Chapter 3 we will talk about our work procedure, methods and techniques to build a Movie Recommendation model.
- In Chapter 4 we will discuss about the Experimental Results and Discussion of our build model.
- In Chapter 5 we will talk about the Impact of Society, Environment, Ethical Aspects and Sustainability plan of our work.
- In Chapter 6 we have discussed about the Summary, Conclusion and Further Study of the work.

CHAPTER 2

BACKGROUND

2.1 Terminologies

Recommendation has become a popular research issue in recent years. Numerous recommendations exist for models. For instance, recommending a music, a movie, or a piece of artwork. Collaborative filtering, content-based filtering, and hybrid filtering are all options for movie suggestion. The focus of my work is content-based filtering. A few papers pertinent to our investigation were discovered. We will categorize Bangla content-based screening in our work. We need to introduce some new terms in order to put our work into practice. I have to gather a lot of data for my job, clean it up, and then turn it into a numeric number so that algorithms can be applied. For this, the terms "count-vectorizer" and "TF-IDF vectorizer" are introduced. Additionally, we'll mention a concept called cosine similarity. We shall go into more detail regarding this word in the following chapter. We analyzed some earlier work that was linked to movie suggestion in order to apply our job flawlessly and to become familiar with this new word. Below, we briefly outline a few of them.

2.2 Related Works

Numerous scholars have studied numerous recommendation techniques [1–9]. Information items that are likely to be of interest to the user are recommended by the recommendation system. Content-based, association-based, demographic-based, and collaborative methods are the four different types of recommendation strategies.

Item-to-item similarity is used in the content-based technique. If a user likes B, we suggest A since it is comparable to B. Item-to-item similarity is used in the association approach as well. With this approach, we do not determine whether they are comparable or not in reality. We determine that two objects are comparable if they have a significant connection with one another. Both the demographic method and the collaborative method make use of inter-person similarities. To determine if persons are similar, the demographic approach requires to know their real characteristics. Correlation between users is used in collaborative methods.

The collaborative filtering-based MOVREC movie recommendation system was introduced by D.K. Yadav [10]. Information submitted by the user is used for collaborative filtering. Following analysis of the data, customers are given recommendations for movies, starting with the highest rated film.

Additionally, the system allows the user to choose the criteria on which he wants the movie to be recommended. Two conventional recommender systems, content-based filtering and collaborative filtering, have been examined by Luis M. Capos et al. [11]. He put forth a new technique that combines collaborative filtering with a Bayesian network because each has disadvantages of its own. The suggested system offers probability distributions to enable relevant conclusions and is optimized for the situation at hand. Harpreet Aur et al. [12] have introduced a hybrid system. The system employs a combination of collaborative and content filtering algorithms. Utkarsh Gupta et al. [13] employ chameleon to group the user- or item-specific information into a cluster. This effective recommender system technique is based on hierarchical clustering. Two method computing cluster representations were shown and assessed. The efficiency of the two proposed methods was compared using a centroid-based solution and memory-based collaborative filtering techniques.

The resulting recommendations were significantly more accurate as a result when compared to the centroid-based technique alone. Movie Recommender is a system that makes movie recommendations based on the user's profile, according to Costin-Gabriel Chiru et al[14] .'s proposal. This system makes an effort to address the issue of unique recommendations that arises from neglecting the user-specific data. Hongli Lin et al. suggested a technique called content-boosted collaborative filtering to forecast the level of difficulty of each case for trainees (CBCF). There are many distinct kinds of recommender systems with various methodologies, some of which are categorized in this document.

In essence, our approach employs the item-based strategy. Later, we'll go over a more thorough explanation of our system for making recommendations.

2.3 Comparative Analysis and Summery

Numerous studies have been published in the field of developing recommendation systems. [15] Systems for recommendations are frequently employed in a variety of applications to suggest

goods for which a consumer is likely to be worried about their preferences. We will contrast one earlier work with later ones in this section. The comparison of prior movie recommendations is shown in Table 2.1.

Table 2.1 Summary of Previous Research

Work Type	Author(s)	Features Considered	Methods Used
Explanatory Research	Litman (1983) [16]	Production cost, critic's rating, genre, distributor, release season, main actor's award history	Regression Analysis
	Prag and Casavant (1994) [17]	Marketing cost, quality, star value, sequel, award, genre, MPAA rating	Regression Analysis
	.Basuroy, Chatterjee and Ravid (2003) [18]	Critical review, star power, budget	Regression Analysis
	Nelson and Glotfelty (2012) [19]	Star power	Regression Analysis
Predictive Research	Sharda and Delen (2006) [20]	MPAA rating, competition, star value, genre, special effects, sequel, number of screens at the initial day of release	Logistic Regression, Discriminant Analysis, Classification and Regression Tree, Neural Networks
	Eliashberg, Hui and Zhang (2007) [21]	Movie script	Classification and Regression Tree
	Zhang, Luo and Yang (2009) [22]	Nation, director, performer, propaganda, content category, month, week, festival, competition, cinema number, screen number	Neural Networks
	Du, Xu and Huang (2014) [23]	Microblog posting counts, microblog posting content	Support Vector Machine, Neural Networks

2.4 Scope of the Problem

When we looked at and reviewed the prior work on movie recommendation, we came across some works, including collaborative filtering movie recommender systems, swarm optimization recommender systems, movie recommendation frameworks, etc. Information-filtering techniques called recommender systems aim to forecast how users and objects will be rated. mostly using big data to support their lies. Movies Systems for recommendations give users a way to group users according to their interests. Consequently, recommender systems become a crucial component of websites and e-commerce software. The industry is currently working to incorporate a variety of cutting-edge recommender systems that focus on group suggestions, POIs, or meta data analysis. We understand the importance of this topic. As we all know, digital technology is developing

swiftly. The internet, mobile devices, and apps are widely used. Hence the exponential surge in recommendations. Everyone desires quick and easy solutions. As a result, everyone is concentrating on suggestions for user goals. We must concentrate on it if we want to make this region more useful. Every sector relies on recommendations. We may use recommendations from websites like Facebook, YouTube, and movies wherever we go. Many people are sluggish and expect everything to be available right away. They place a lot of importance on this issue. For them, we must develop this technology.

2.5 Challenges

Throughout the course of our work, we encounter several difficulties. The dataset is the primary issue. There are a lot of movies online. I must gather them with accurate data. If some data were discovered, but they lacked structure. We experienced the same issue in our instance. I must first arrange them in the proper order and position. I then need to tidy up my dataset. We have a few strategies for cleaning, such as removing redundant or irrelevant data, correcting structural issues, filtering undesired outliers, handling missing data, etc. The greatest issue in our thesis is data cleaning. So, First, we must determine whether any data are duplicates. During data collecting, duplicate observations will frequently occur. Duplicate data can be produced when you merge data sets from several sources, scrape data, or get data from clients or other departments. When you measure or transfer data and find odd naming conventions, kinds, or wrong capitalization, such are structural faults. Because many algorithms won't tolerate missing values, you can't overlook missing data. There are a few options for handling missing data. Both can be thought of, but none is ideal. We need to concentrate on our key point after data cleaning. Next, we must determine whether our data can be handled by the cosine similarity technique and count vector model. Characters and words are not understood by machines. Consequently, in order for a machine to understand text data, it must be represented in numerical form. Text can be turned into numerical data using the count vectorizer technique. The similarity between two vectors in an inner product space is then measured using cosine-similarity. It establishes whether two vectors are roughly pointing in the same direction by calculating the cosine of the angle between them. Due to these restrictions and difficulties, we are unable to gather more data. The result will be more precise and accurate if the dataset is huge.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Research Subject and Instrumentation

In our effort, we want to create a model that can suggest movies. The first step in accomplishing our project's goal is to perform enough background research, which is why a literature review will be done. We need to create a dataset in order to create this model, and we also need to know what kind of work we are doing. We have two classes at work. So, binary classification is what we do. We are essentially addressing a categorization problem here. In machine learning algorithms, there are two techniques to build a model. Learning that is supervised and unsupervised. In supervised learning, we provide input and output to the system, which uses the input and output data to forecast previously unseen data. Contrarily, with unsupervised learning, we are merely provided raw data, which a computer then clusters based on the data pattern. We receive input and output data from the system at work so that we can train the model. Therefore, unsupervised learning is used here. Unsupervised learning uses a variety of classification algorithms to address classification issues. Since the entire project is based on a substantial amount of movie data, we chose a quantitative research approach. A movie recommendation system, also known as a movie recommender system, uses machine learning to anticipate or filter a user's film choices based on past decisions and behavior. The mechanism of sophisticated filtration makes predictions about the potential movie selection. We offer a thorough explanation of the algorithm for quality control to guarantee test validity. We employ machine learning for this, which also covers the topics of count vector and cosine similarity. We will go over all of the algorithms and how they operate in the forthcoming section on the proposed methodology. All concepts in the section describing the suggested methodology will be briefly explained.

3.2 Data collection Procedure

When collecting data is more balanced and dependable to train the system precisely, machine learning algorithms function very well. Therefore, gathering data is crucial to our work. We initially need to create our own movie recommendation dataset for our research. Numerous facts are available on numerous websites. I must therefore gather the relevant information to enable my

user to access their preferences. To offer an excellent service, I must collect all relevant information on well-known movies. I gathered information for this between 2004 and 2020. I merely gathered information on all well-liked movies. These data sets contain a variety of genres, including comedy, action, and adventure. So, it is necessary to gather all possible data. I get 5000 data for this model from two distinct data sets, each of which contains 5000 data. Here, I've attempted to examine a few samples from our data gathering. You may get a sense of the type of data we used by looking at that. There are a large number of rows and columns in the both data set. Some sample data are provided in Table 3.1 and 3.2.

Table 3.1: 1st Movie Dataset for analysis

Original_Title	Original_Language	Release_Date	Runtime
Avatar	en	2009-12-10	162
Pirates of the Caribbean: At World's End	en	2007-05-19	169
Spectre	en	2015-10-26	148
The Dark Knight Rises	en	2012-07-16	165
John Carter	en	2012-03-07	132
Troy	en	2004-05-13	63
Men in Black II	en	2002-07-03	88
Spider-Man	en	2002-05-01	121

Table 3.2: 2nd Movie Dataset for analysis

Movie_id	Title	Cast	Crew
19995	Avatar	Sam Worthington, Zoe Saldana, SigourneyWeaver	James Cameron
285	Pirates of the Caribbean: At World's End	Jhonny Depp, Orlando Bloom, Keira knightley	Gore Verbinsi
206647	Spectre	Daniel Craig, Christoph Waltz, Lea Seydoux	Sam Mendes
49026	The Dark Knight Rises	Christian Bale, Michale Caine, Gary Oldman	Christopher Nolan
49529	John Carter	Taylor Kitsch, Lynn Collins, Samantha Morton	Andrew Stanton

In my data sets there are so many column. I just showed some column to give the idea of my datasets.

3.3 Statistical Analysis

I was able to obtain 5000 data after gathering information from numerous sources. I have 5000 distinct types of movies spanning the years 2004 to 2020 in my dataset. We have two data sets, each of which has 5000 identical records with a different attribute. We have 5000 rows and 20 columns in our first data set, and 4 columns and 5000 rows in our second data set. Budget, genres,

homepage, id, eywords, original language, original title, overview, popularity, production companies, production countries, release date, revenue, runtime, spoken languages, status, tagline, title, vote average, vote count, and movie id are all columns of attributes in my first dataset. The columns of attributes in my second dataset include movie id, title, cast, and crew. Due to the fact that both data sets contain the same column name title, we then marge them on that basis. Then, our final data set, which has 5000 rows and a total of 23 columns. Then, we discover NA values in our sets of data. In homepage column we have 3096 NA values and in overview column we have 3 NA values. We can drop overview NA values but we have to fil our homepage column NA values. Our goal is to deliver correct suggestions, so we'll concentrate on genres, key phrases, the title, an overview, the cast (actor name), and the crew (director name).. On this column attribute, machine learning, count vector, and cosine-similarity will all be used.

- We have 23 columns in our final dataset.
- Our dataset is available in CSV (Comma Separated Value) format which extension is .csv

3.4 Proposed Methodology

In the part that follows, we'll go over our research technique. In my research, count vector and cosine-similarity were also included in the machine learning focus. We create our own dataset and run this algorithm on it. We are able to view our "tmdb 5000 movies.csv" and "tmdb 5000 credits" datasets using the pre-processing approach on kaggle.com. Let's now select the most appropriate models and train them to increase forecast accuracy. We used two different models: Cosine and Vector Count Similarity When determining each model's accuracy and examining them, let's cross predicting some things. For content-based recommender systems in particular, we have a tendency to come up with new ideas to increase the representative's accuracy and provide users with highly relevant films that match their tastes in movies. I broke up my work into a few steps for this. The steps of our methodology are shown in Figure 3.1. The remaining steps of the approach are given below.

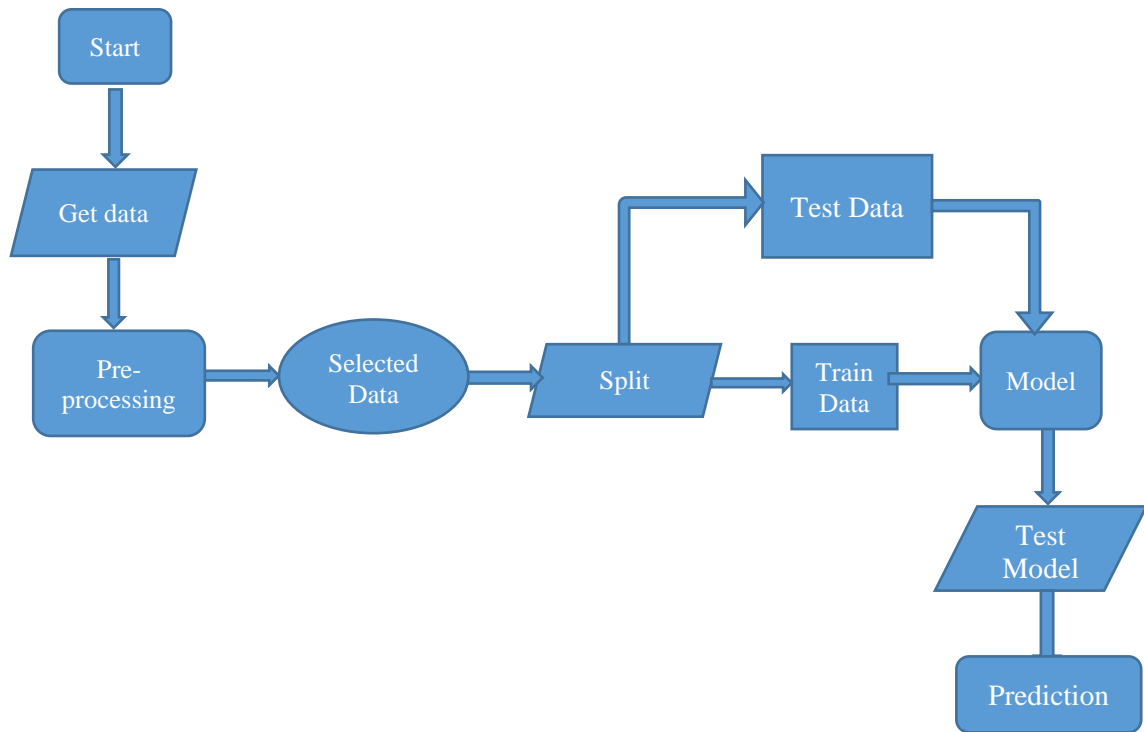


Figure 3.1: Proposed Methodology

3.4.1 Data Preprocessing

Pandas and Numpy libraries are used in machine learning for pre-processing.

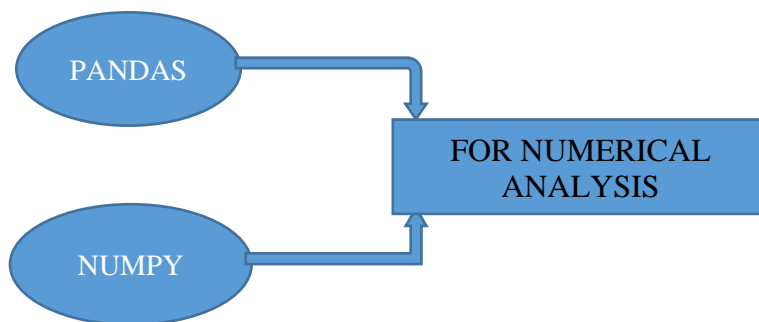


Figure 3.2: Machine Learning Architecture

NUMPY: Numpy is short for either "Numerical Python" or "Numeric Python." Its companion Python package for parsing text files offers rapid mathematical processing on arrays and matrices. The notebook will be foreign to Numpy using:

```
>>>import numpy as np
```

PANDAS: One of the most widely used Python libraries in information science is Pandas[24]. It offers excellent information analysis tools that are simple to use. As a result, pandas square is now capable of offering a number of additional features, such as creating pivot tables, computing columns supported by various columns, and creating graphs. Pandas will be introduced into Python by:

```
>>>import pandas as pd
```

In situations where we are using machine learning techniques such as Scikit Learn, NLTK (Natural Language Toolkit), and the circular function similarity formula, picking a model and strategy is a crucial step.

SCI-KIT LEARN: Scikit-learn (Sklearn) is the most reliable and practical Python package for machine learning. It offers a variety of affordable tools for applied math modeling, classification, regression, clumping, and spatial property reduction using a Python consistency interface. NumPy, SciPy, and Matplotlib are the building blocks of this package, which is primarily written in Python. Scikit-learn library is focused on modeling the data, as opposed to specializing in loading, modifying, and summarizing information. Stop words are merely a list of words that you don't want to use as alternatives. To utilize an integral list, set the stop words='english' argument. As an alternative, you can set stop words that fit within a specific list. Its default value is none [25].

NLTK: The NLTK (Natural Language Toolkit) Library may be a collection of programs and libraries for the use of language in applied mathematics. It's one of the most potent NLP libraries, containing tools for teaching computers to understand human language and respond to it appropriately. In Python's NLTK, text standardization techniques for language processing include stemming and lemmatization. These methods are frequently employed for text preparation. Lemmatization and stemming differ in that lemmatization takes longer since it considers the context of words before cutting them, but stemming is quicker because it does so. Stemming is one possible way for standardizing words used in language processing. It is a technique for shortening a search in which a group of words in a sentence are rearranged into a sequence. With this process, words that have the same meaning but differ slightly depending on the sentence or context are normalized [26].

In Python's NLTK, text normalization techniques for language processing include stemming and lemmatization. These methods are frequently employed for text preparation. Stemming and lemmatization differ in that stemming is speedier since it cuts words without considering context, whereas lemmatization takes longer because it considers context before processing. Stemming may be used to normalize words as part of the linguistic process. This technique involves turning a group of words from a long sentence into a short one in order to make it run more quickly. With this process, words with similar meanings but slight differences depending on the sentence or context are normalized [27].

3.4.2 Feature Extraction

After preprocessing, we must clean our data, but we cannot feed this data to our machine learning model. We need to extract the characteristics from clean data. To extract the features, we must convert the text data into a numeric value. However, the performance of the machine learning model depends on how successfully the characteristics from the text are extracted. To accomplish this, we must use some techniques that convert our text into a vector, or to put it another way, a numerical value. The name for this is one hot encoding. In this instance, the vectorizers count and TF-IDF (term frequency-inverse document frequency) are applied. The count vectorizer is the most useful feature extraction method in a content-based filtering paradigm. The count vectorizer essentially builds a vector from the text data based on the word frequency (count) of each word that appears in the sentences. This sentiment analysis method is quite helpful. The count vectorizer represents unique words as matrix columns, and each row of text data from the dataset is represented by a matrix row. The value obtained after counting the word frequency is then entered into the matrix. TF-IDF is the most complex and popular approach for features extraction from processed text data. The accuracy of the proposed model can occasionally be enhanced by features extraction using the TF-IDF vectorizer method. Because the word weight of the entire document is taken into account when creating a matrix. The TF-IDF formula is [29]:

$$tf - idf(t) = tf(t, d) * idf(t) \text{ ----- (1)}$$

Where,

$$tf(t, d) = \sum_{x \in d} fr(x, t)$$

$$idf(t) = \log \frac{|D|}{1 + |\{d : t \in d\}|}$$

Here,

- *term denoted by 't' and documents denoted by 'd'*
- *fr(x, t) is a function which returns term frequency count*
- *|D| is the number of whole documents*
- *|\{d : t \in d\}| is the number of documents where t appears*

In our work we tried two features extraction methods and when we compare which is suitable and increase our model accuracy based on our dataset. We found that Count-vectorizer get more accuracy than TF-IDF. So, in our work we use Count-vectorizer method to extract feature from our text data. In the upcoming Chapter 4 we will discuss more about this and show the outcome these methods.

3.4.3 Implementation

This paper is related to machine learning that's we will focus on machine learning algorithms. We know machine learning divided into three categories. Three categories are Supervised learning, Unsupervised learning and Reinforcement learning. My paper is based on unsupervised learning. Now we discuss about unsupervised learning down below.

3.4.3.1 Unsupervised Learning: Unsupervised learning uses machine learning techniques to group and evaluate unlabeled datasets. These algorithms locate data clusters or hidden patterns without the aid of a person. Because of its ability to identify informational similarities and contrasts, it is the greatest choice for exploratory data analysis, cross-selling strategies, consumer segmentation, and picture recognition[28].

Clustering, association, and dimensionality reduction are the three basic tasks that unsupervised learning models are used for. We'll talk about each teaching strategy below.

Clustering: Unlabeled data are grouped using the data mining technique of clustering according to their similarities or differences.

Association Rules: A rule-based approach for determining associations between variables in a given dataset is called an association rule. These techniques are commonly used for market basket analysis, giving businesses a better understanding of how various products relate to one another.

Dimensionality Reduction: While having more data typically yields more accurate findings, it can also impact how well machine learning algorithms perform and make dataset visualization more difficult. The dimensionality reduction approach is used when a dataset contains too many features or dimensions.

Nowadays, it's common practice to utilize machine learning techniques to improve the user experience of products and test systems for quality assurance. Unsupervised learning provides an exploratory technique to investigate data, which enables businesses to identify patterns in enormous amounts of data more quickly than manual observation. Watch now to see how unsupervised learning helps our recommendation engine.

Recommendation Engines: Unsupervised learning can assist in the discovery of data trends that can be used to create more effective cross-selling strategies using historical purchase behavior data. When a consumer is checking out with an online retailer, this is utilized to suggest appropriate add-ons.

Let's discuss about missing data before we move on to implementation.

3.4.3.2 Handling Missing Value: 5000 data are included in our data set. The first step in using any approach is to look for any missing values. Because our dataset has a problem with missing values. Few methods exist for dealing with missing values.

1. Eliminating rows with empty values
2. Fill in the gaps for a continuous variable
3. Other imputation Methods
4. Using Algorithms that support missing values
5. Prediction of missing values
6. Imputation using deep Learning Library

We have 23 columns in our data set. From those we get 5 column which have null values. Let's see our dataset of missing values down below

Table 3.3: Missing Values

Column Name	Missing Data
Homepage	3096
Overview	3

There are a few NA values in this summary column, so we can ignore them. However, there are 3096 NA values in the homepage column. If we remove these numbers, our dataset will shrink, which will affect how we implement things. Therefore, we must fill them. In our dataset, we therefore populate this NA value with the name of their film.

Our dataset is now prepared for the deployment of the following phase. We employ methods like count vectorization and cosine-similarities, which enable us to obtain the appropriate recommendation.

3.4.4 Count vector

Count vector: Count By executing preprocessing operations, such as changing all words to lowercase and deleting special characters, vector refers to breaking down a sentence or any text into words. Characters and words cannot be understood by machines. Consequently, in order for a machine to understand text data, it must be represented in numerical form. Text can be converted to numerical data using the counter vectorizer technique [29].

We must merge overview, genres, keywords, cast, and crew into a new column called "all tags" in order to transform numerical data to numbers. All tag values must be converted into lower case after the margin for improved accuracy.

We are now going to apply count vectorization to our dataset after changing the case of all tag values. Many of the same words, such as dance, dancing, and danced, were used before this. To get more precision, we must merge this word into a single word dance. We employ the nltk function for this. After completing this step, we are now prepared to use counter vectorization.

For my dataset, we currently have a 5000 by 5000 matrix. Each and every row in this represents a movie, and each and every column a word's vector.

We'll discover similar videos after converting to a vector format. For instance, if we enter the name of a specific movie in the avatar field, our algorithm would suggest five films that are comparable. Cosine-similarity must be used to recommend five films.

3.4.5 Cosine Similarity

Cosine Similarity: The angle between two non-zero inner product spaces is cosined when measured by cosine similarity. The direction rather than the quantity of resemblance is given greater weight in this comparison. In conclusion, the similarity between two cosine vectors aligned perpendicularly and in the same direction is 0, while the similarity between them is 1. When two vectors are diametrically opposed, or pointing in completely opposite directions, the similarity index is equal to -1. However, cosine similarity is widely used in positive space, in the region between 0 and 1. Cosine Similarity does not consider or evaluate changes in magnitude; it just represents similarities in orientation (length) [30].

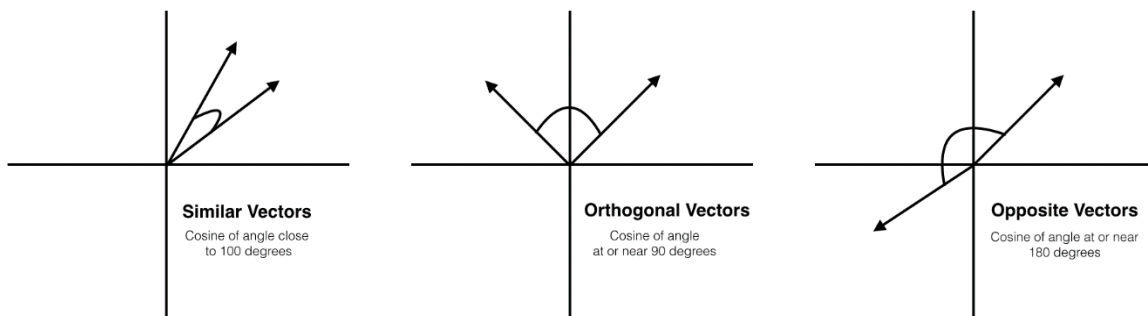


Figure 3.3: Cosine Similarity orientation (length)

3.4.5.1 Cosine Similarity Work

The Cosine Similarity measurement begins by finding the cosine of the two non-zero vectors. This can be derived using the Euclidean dot product formula which is written as:

$$\mathbf{A} \cdot \mathbf{B} = \|\mathbf{A}\| \|\mathbf{B}\| \cos \theta \quad \text{-----} \quad (2)$$

Figure 3.4: Euclidean Formula

Then, given the two vectors and the dot product, the cosine similarity defined as:

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}, \quad \text{----- (3)}$$

Figure 3.5: Cosine Similarity

The output will produce a value ranging from -1 to 1, indicating similarity where -1 is non-similar 0 is orthogonal, and 1 represent total similarity [30].

3.4.5.2 Applications of Cosine Similarity

Beyond the realm of abstract mathematics, applications for cosine similarity exist. The measurement is employed in the text matching, information retrieval, and data mining processes. When the qualities of a variable are allocated to the vectors, the measurement transforms into an important tool for identifying similarities across things.

3.4.5.3 Cosine Similarity and Machine Learning

Machine learning and cosine similarity are used in applications like data mining and information retrieval. In order to process a database of documents, for example, it is feasible to assign each phrase a dimension and a corresponding vector that correlate to the frequency of that term in the text. As a result, it is easy to distinguish distinct publications depending on how much their subject content resembles or overlaps.

Let's now discuss the benefits of cosine similarity for our system.

We will use Cosine Similarities to identify related movies after converting all tags into vectors. How can we obtain 5 comparable movies when we search for a specific movie called "Avatar"? By applying the cosine similarity function, we may obtain five recommended movies. The value of in the cosine similarity is more closely related to the search movie. We now have a 5000 by 5000 dataset with cosine values ranging from 0 to 1 as a result of applying cosine similarities. We will now convert all of the rows' probability values, which are listed in ascending order and

decreasing order, respectively. After sorting everything into decreasing orders, we'll pull out the top 5 films and give our recommendations based on their names.

Now that we have applied count vectorization and cosine similarity, we will have the suggested films.

3.5 Implementation Requirements

Our research title is “Movie Recommendation using data science”. Using the count-vector and cosine-similarity techniques, we can extract feature from the text data. We collected movie data from various source and I want to make a system which can recommend movie. To process and evaluate the entire work we need a high configuration Computer setup with GPU and others necessary instrument. In below we mention the all hardware, software and advance tools which we need to complete this work.

Hardware and Software:

- Intel Core i5 8th gen integrated with 8GB ram
- 1 TB Hard Disk
- Google Collab with 12GB GPU and 350GB ram
- High Speed Internet Connection

Advance Libraries and Tools:

- Windows 10
- Python 3.8
- Pandas
- NumPy
- Regular Expression (RE Library)
- NLTK
- Scikit-Learn

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experimental Setup

This section will detail the model performance that we used with our movie dataset. I employed unsupervised machine learning for my research. We need to tidy up our data set first. Where we will make adjustments for our missing values. After the data cleaning procedure is complete, we must merge all the columns that will enable us to identify the recommended course of action. We provide recommendations for this method using some of the column names, including genre, keywords, title, cast, and crew. Thus, we created a new column called "all tags" to hold this margin column. Then comes the major objective. Count vectorizer may now be used to translate text into a numerical number between 0 and 1. because text is not understood by machines. We must therefore turn them into numbers. Now that we have numerical numbers that are 5000 by 5000, we can move on to the next procedure. Next, we conducted data analysis using the cosine-similarity technique. A measure of how similar two number sequences are is called similarity. The cosine similarity is defined as the cosine of the angle between the sequences when it comes to defining it. The sequences are then viewed as vectors in an inner product space. Now that cosine similarities have been applied, we have 5000 by 5000 cos values that range from 0 to 1. These factors enable us to find similar recommended movies to the ones we were looking for. If the cos values in this case are smaller, the movie is more likely to be the searched movie. That is how the top 5 movies are determined.

All this discussed on chapter 3.

Down below we discuss about the Exploratory Data Analysis (EDA) has been inspired by Heeral Dedhia's blog on medium.com[31].

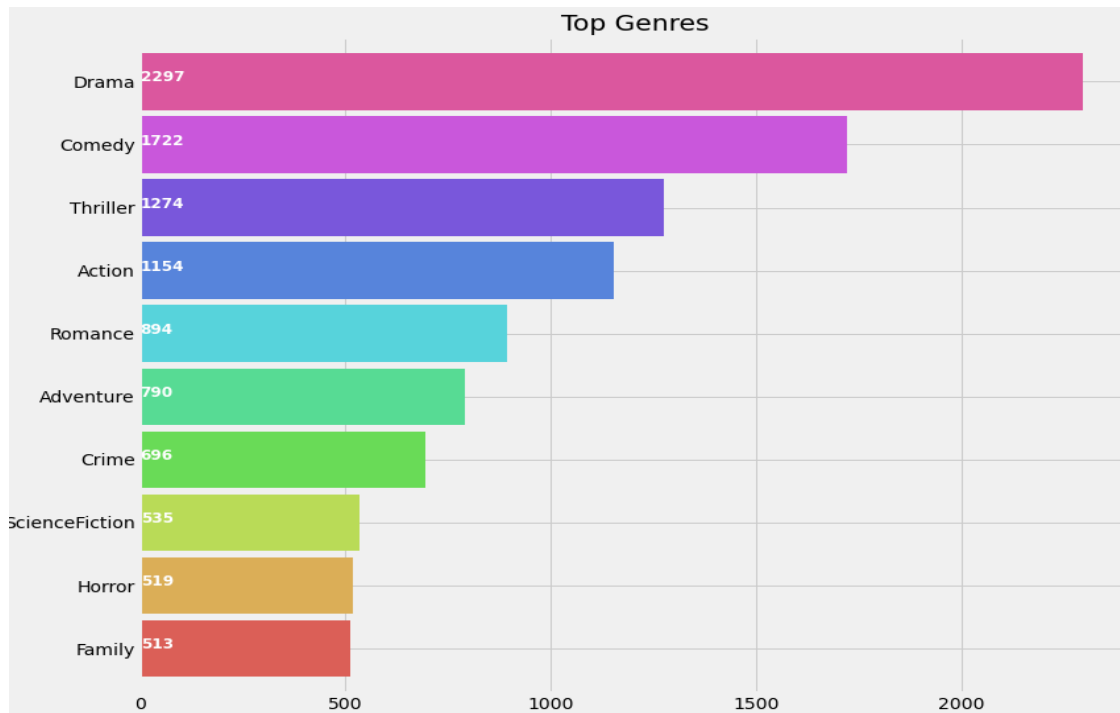


Figure 4.1: Top Genres

In comparison to Family and Horror films, the most drama-themed films are produced. A film may be in more than one genre.

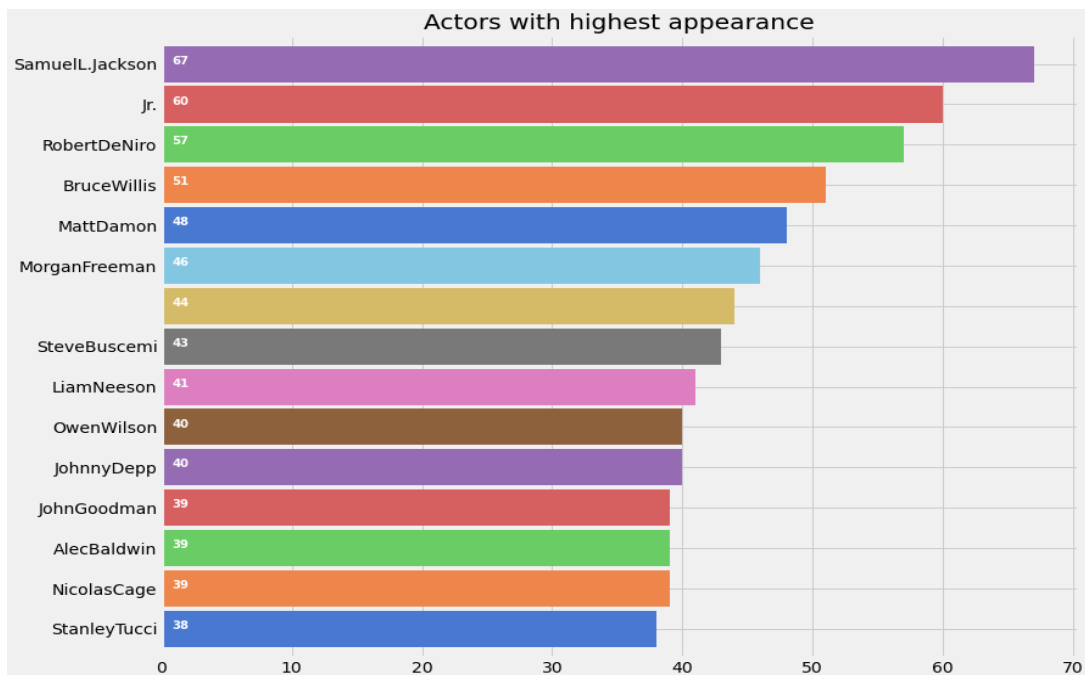


Figure 4.2: Actor with highest appearance

The above figure indicates the actors with the highest appearance in the decreasing order.

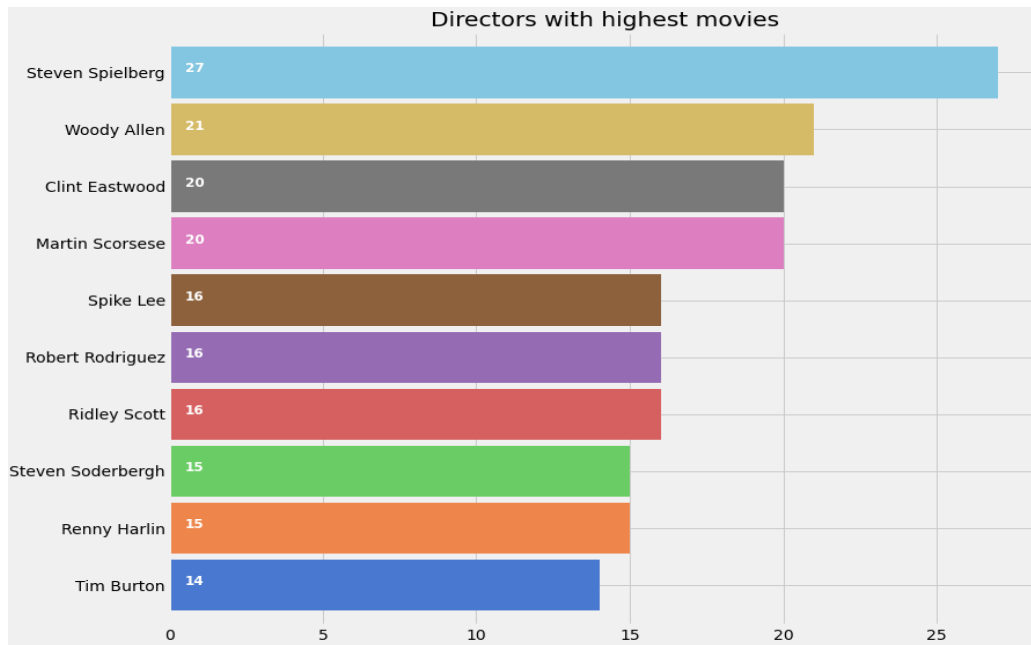


Figure 4.3: Directors with highest movies

The above figure indicates with the highest appearance in the decreasing order.

4.2 Experimental Result and Analysis

After training, the model needs to be put to the test to check if it will work effectively on Earth. A subset of the analysis-ready data set is utilized to gauge the model's efficacy. As a result, the model is forced to test itself against data that wasn't part of its training. To put it another way, we use our tested knowledge and model to determine whether or not it is suitable for analysis. Machine learning incorrectly leverages knowledge to deliver answers to queries. We typically reach certain conclusions through reasoning, also known as recommendation. This is typically the desired result of machine learning when it is fully exploited. To construct vectors from the text, we can use CountVectorizer, TfidfVector, Glove, or Word2Vec. After vectorizing the text, we must ascertain how similar the vectors are to one another. The similarity between the vectors can be discovered using a variety of methods, including cosine similarity, sigmoid kernel, and others. Here, we employ two algorithms. For the first algorithm, we employ CountVectorizer and Cosine Similarity with Content-based Recommendation. In this instance, we'll use CountVectorizer to turn the preprocessed text from the 'combine feature' attribute into vectors. We will use Cosine Similarity to determine how similar the vectors are after we have them. The second approach uses

TfidfVectorizer and cosine similarity for content-based recommendation. In this instance, we'll make vectors using the preprocessed text provided in the 'combine feature' attribute using TfidfVectorizer. We will use cosine similarity to determine how similar the vectors are after we have them. After receiving recommendations from algorithms 1 and 2, choose the most popular movies from each of the recommendations first. The remaining films should then be added to the standard films. Lastly, we can use our model to predict whether or not the user would be offered a comparable film based on his or her preferences based on the similarity of the films. My main objective is to suggest movies that are comparable to what customers have been looking for. The final advice is marginally superior to the separate recommendations of Algorithms 1 and 2 mentioned in this study effort, as shown in the results below. Therefore, it is usually preferable to alter the output of various algorithms to produce a final product that contains the benefits of each method separately.

```
get_recommendations('Avatar', cosine_similarity_cv)
```

```
206                Clash of the Titans
 71                The Mummy: Tomb of the Dragon Emperor
 786                The Monkey King 2
 103                The Sorcerer's Apprentice
 131                G-Force
 215                Fantastic 4: Rise of the Silver Surfer
 466                The Time Machine
 715                The Scorpion King
 1                Pirates of the Caribbean: At World's End
 5                Spider-Man 3
 9                Batman v Superman: Dawn of Justice
 10                Superman Returns
 12                Pirates of the Caribbean: Dead Man's Chest
 14                Man of Steel
 17                Pirates of the Caribbean: On Stranger Tides
Name: title, dtype: object
```

```
get_recommendations('Avatar', cosine_similarity_tv)
```

```
2403                Aliens
 206                Clash of the Titans
 587                The Abyss
 43                Terminator Salvation
 132                Wrath of the Titans
 282                True Lies
 1448                Sabotage
 47                Star Trek Into Darkness
 3439                The Terminator
 3184                The Ice Pirates
 4114                Subway
 2827                Crossroads
 812                Pocahontas
 94                Guardians of the Galaxy
 279                Terminator 2: Judgment Day
Name: title, dtype: object
```

Figure 4.4: Recommendations similar to 'Avatar' movie using Algorithm 1 and Algorithm 2

```

get_final_recommendations('Avatar')[0:15]
['Clash of the Titans',
 'The Mummy: Tomb of the Dragon Emperor',
 'Aliens',
 'The Monkey King 2',
 'The Abyss',
 "The Sorcerer's Apprentice",
 'Terminator Salvation',
 'G-Force',
 'Wrath of the Titans',
 'Fantastic 4: Rise of the Silver Surfer',
 'True Lies',
 'The Time Machine',
 'Sabotage',
 'The Scorpion King',
 'Star Trek Into Darkness']

```

Figure 4.5: Final Recommendations similar to 'Avatar' movie

```

get_recommendations('Titanic', cosine_similarity_cv)
1081          Revolutionary Road
4247          Me You and Five Bucks
49            The Great Gatsby
872           All the King's Men
1311          Angel Eyes
1492          The Reader
2449          Sense and Sensibility
2661          Romeo + Juliet
2701          Little Children
2946          What's Eating Gilbert Grape
4589          Fabled
297           Blood Diamond
351           The Departed
439           Shutter Island
622           Body of Lies
Name: title, dtype: object

get_recommendations('Titanic', cosine_similarity_tv)
1081          Revolutionary Road
609           Escape Plan
282           True Lies
3097          Swept Away
3439          The Terminator
818           Captain Phillips
2403          Aliens
984           Into the Blue
3695          The Blue Lagoon
587           The Abyss
872           All the King's Men
279           Terminator 2: Judgment Day
49            The Great Gatsby
622           Body of Lies
351           The Departed
Name: title, dtype: object

```

Figure 4.6: Recommendations similar to 'Titanic' movies using Algorithm 1 and Algorithm 2

```
get_final_recommendations('Titanic')[0:15]
['Revolutionary Road',
 'The Great Gatsby',
 "All the King's Men",
 'The Departed',
 'Body of Lies',
 'Me You and Five Bucks',
 'Escape Plan',
 'Angel Eyes',
 'True Lies',
 'The Reader',
 'Swept Away',
 'Sense and Sensibility',
 'The Terminator',
 'Romeo + Juliet',
 'Captain Phillips']
```

Figure 4.7: Final Recommendations similar to 'Titanic' movie

That's how we are going to give the recommendation for the users.

4.3 Discussion

The model must be tested after training to see if it would function properly on Earth. To assess the model's effectiveness, a portion of the data set prepared for analysis is used. This places the model in a situation where it runs against items that weren't covered by its training. In other words, we tend to use outdated, tested information and models for analysis purposes to determine whether or not the model is working well. Machine learning uses knowledge improperly to provide answers to questions. Recommendation, or reasoning, is the process through which we usually arrive at particular conclusions. When machine learning is fully utilized, this is frequently its intended outcome. As a result of the similarity of the films, we can finally utilize our model to forecast whether or not the user would be recommended a comparable movie based on his or her interests.

CHAPTER 5

IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

5.1 Impact on Society

Outstanding movies and television have a big impact on any civilization. Children grow as a result of their experiences. Along with portraying our many cultures, the film has long influenced our beliefs and ideals. People who copy the fashions of musicians and actors are a good example. Today's civilizations regularly use rhetorical strategies that are influenced by the motion picture business. A great movie will leave the viewer feeling inspired, informed, and entertained in different ways. Think about how music affects listeners, for example. They compel us to think. They help us develop compassion. They motivate us to help others and carry out gestures of kindness for the greater benefit. There are a number of movies whose narratives motivate us to get out of bed every morning and face the world with positivity. They encourage us to triumph over our own challenges and improve the lives of others. For instance, movies like *The Bucket List* (2007) and *The Pursuit of Happiness* (2006) inspired their fans to make the world a better place for everyone.

5.2 Impact on Environment

From vintage black and white silent movies to the box office hits of today, the motion picture business has gone a long way. Movie productions began to have more of an impact on the environment during that procedure. Although each film production has a unique impact, there are a few underlying problems that lead to environmental deterioration.

Filmmaking has changed in recent years to become more environmentally friendly. Studios in Vancouver, Los Angeles, and London are implementing sustainable production techniques that take into account social and economic as well as environmental challenges.

5.3 Ethical Aspects

The only component that might pose an ethical dilemma is the movie information. However, all of the material we use for our research comes from open sources like Wikipedia and our own movie

database. Therefore, there are no issues with user privacy or data confidentiality. Down below I am going to share some ethical issue about recommendation engine.

1. **Addictiveness:** The fact that recommendation systems are designed to be addictive is one moral dilemma. They are designed to pique and hold users' interest for extended periods of time. Consider the auto play options on Netflix and YouTube. Both provide content that is personalized based on your data profile and automatically plays it to keep you interested.
2. **Extreme Content:** Another crucial ethical problem with recommendation engines has arisen as a result of the competition to attract and hold users' attention: the delivered content may not truly be in the users' best interests and promotes polarization. Recommendation algorithms are "broken" and have turned into "The Great Polarizer," as Renee Diresta writes in Wired. On YouTube, for instance, the algorithm frequently serves up increasingly controversial video in an effort to keep you interested and increase Google's revenue from advertising.

Personal Concern: Personal data must typically be collected in order to analyze personalized recommendations, which puts consumers at risk for privacy violations. The data "undesirably betrays the users' particular interests to the recommender," claims a 2018 study article that was published in Engineering. Additionally, without user permission, the data may be sold to third parties. Consider the Facebook/Cambridge Analytica incident as an illustration. The possibility of platform hacks and user data leaks, which have happened to Facebook (and other platforms) several times, is the third privacy concern.

5.4 Sustainability Plan

Our goal is to support the next generation in their efforts to build a just society, and we believe that doing so will lead to a significant upsurge in both the environment and society as a whole. A solid movie recommendation can have a significant positive impact on our society and the future generation. Because a good movie can improve a child in all ways. This algorithm suggests family-friendly movies that everyone can watch. The filter will remove all terrible movies. Our culture can be entertained by a good film, and children can learn things as well. Our society may focus on positive things and prevent bad things by having knowledge. We must therefore create a sound model that will offer us accurate and engaging films.

CHAPTER 6

SUMMARY, CONCLUSION AND IMPLICATION FOR FUTURE RESEARCH

6.1 Summary

Our employment involves recommending movies. Several recommendation models exist, including content-based filtering, collaborative filtering, and hybrid filtering. However, we'll use content-based screening. For research projects to be completed, the dataset is crucial. We are developing a machine learning model for movie recommendation in this effort. This model is excellent for suggesting movies. As a result, there are a few issues we run into while creating this model, in addition to other issues. Each phase and task summary is provided in detail.

Step 1: Planning about this work

Step 2: Problem Formulation

Step 3: Data collection from various websites

Step 4: Data Labeling

Step 5: Data cleaning

Step 6: Data Vectorization

Step 7: Model Selection

Step 8: Model evaluation and performance testing

After execute all of the steps we are finally able to make our model for movie recommendation. Our work contributes in the domain of recommendation research. Working with movie recommendation is pride and in world wide this model can give entertaining and education movie recommendation that can help our society, kids and whole world.

6.2 Conclusion

The importance of recommender systems is growing as a result of the information overload. We specifically look for a novel way to enhance the representation of the movie accuracy for content-based recommender systems.

First off, we employ a content-based recommender algorithm to address the issues we stated at the outset, thus there is no cold start issue. We have cleaned up our data sets before that. use a count vector next. The cosine similarity, which is widely utilized in industry, was then presented.

A content-based recommender model is presented in this master's thesis for everyone.

6.3 Implication for Further Research

Our work is subject to some restrictions and shortcomings. For instance, in our model, we only employ machine learning algorithms. Only cosine similarity and Count Vectorized text modification techniques are used in addition to this. Additionally, our data are insufficient. We cannot improve the accuracy of the deep learning model without more data.

I intend to use this model on a website. Any user can get the right resources through this website to get the movie set of their choice.

REFERENCES

- [1] S. Bangale, A. Haspe, B. Khemani, and S. Malave, "Recipe Recommendation System Using Content-Based Filtering," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4102283.
- [2] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 116–142, Jan. 2004, doi: 10.1145/963770.963775.
- [3] Ishikawa, "Recommendation System: A Hybrid Approach for Cold Start and Privacy problem.," *Strad Research*, vol. 8, no. 5, May 2021, doi: 10.37896/sr8.5/047.
- [4] W. ZHANG, W. HUANG, and L. XIA, "Recommendation research based on general content probabilistic latent semantic analysis model," *Journal of Computer Applications*, vol. 33, no. 5, pp. 1330–1333, Oct. 2013, doi: 10.3724/sp.j.1087.2013.01330.
- [5] D. Alemayehu, Y. Chen, and M. Markatou, "A comparative study of subgroup identification methods for differential treatment effect: Performance metrics and recommendations," *Statistical Methods in Medical Research*, vol. 27, no. 12, pp. 3658–3678, Jun. 2017, doi: 10.1177/0962280217710570.
- [6] D. C. Wilson, B. Smyth, and D. O. Sullivan, "Sparsity Reduction in Collaborative Recommendation: A Case-Based Approach," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, no. 05, pp. 863–884, Aug. 2003, doi: 10.1142/s0218001403002678.
- [7] C. Hall, R. Naughton, and X. Lin, "Identifying, evaluating and recommending digital collections: A social community approach," *Proceedings of the American Society for Information Science and Technology*, vol. 46, no. 1, pp. 1–4, 2009, doi: 10.1002/meet.2009.1450460369.
- [8] R. J. K. R. J. Kuo and Z. W. R. J. Kuo, "Applying Evolutionary-based User Characteristic Clustering and Matrix Factorization to Collaborative Filtering for Recommender Systems," *網際網路技術學刊*, vol. 23, no. 4, pp. 693–708, Jul. 2022, doi: 10.53106/160792642022072304005.
- [9] J. Bobadilla, S. Alonso, and A. Hernando, "Deep Learning Architecture for Collaborative Filtering Recommender Systems," *Applied Sciences*, vol. 10, no. 7, p. 2441, Apr. 2020, doi: 10.3390/app10072441.
- [10] M. Kumar, D. K. Yadav, A. Singh, and V. Kr., "A Movie Recommender System: MOVREC," *International Journal of Computer Applications*, vol. 124, no. 3, pp. 7–11, Aug. 2015, doi: 10.5120/ijca2015904111.
- [11] L. M. de Campos, J. M. Fernández-Luna, J. F. Huete, and M. A. Rueda-Morales, "Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks," *International Journal of Approximate Reasoning*, vol. 51, no. 7, pp. 785–799, Sep. 2010, doi: 10.1016/j.ijar.2010.04.001.

- [12]Harpreet Kaur Virk, Er. Maninder Singh, “Analysis of Movie Recommender System using Collaborative Filtering,” *International Journal of Recent Trends in Engineering and Research*, vol. 3, no. 5, pp. 338–346, May 2017, doi: 10.23883/ijrter.2017.3232.zpbxj.
- [13]Utkarsh Gupta1 and Dr Nagamma Patil2,” “Resource Allocation of Distributed System Based on Hierarchical Clustering Algorithm,” *Distributed Processing System*, vol. 3, no. 3, Jul. 2022, doi: 10.38007/dps.2022.030307.
- [14]Costin-Gabriel Chiru, Vladimir-Nicolae Dinu , Ctlina Preda, Matei Macri, “Analysis of Movie Recommender System using Collaborative Filtering,” *International Journal of Recent Trends in Engineering and Research*, vol. 3, no. 5, pp. 338–346, May 2017, doi: 10.23883/ijrter.2017.3232.zpbxj..
- [15]P. Sharma and L. Yadav, “MOVIE RECOMMENDATION SYSTEM USING ITEM BASED COLLABORATIVE FILTERING,” *International Journal of Innovative Research in Computer Science & Technology*, vol. 8, no. 4, Jul. 2020, doi: 10.21276/ijircst.2020.8.4.2.
- [16]S. Li and D. Zhou, “Research on Application of Collaborative Filtering Algorithm in Digital Movie Recommendation,” *Journal of Physics: Conference Series*, vol. 1651, no. 1, p. 012091, Nov. 2020, doi: 10.1088/1742-6596/1651/1/012091.
- [17]Q. Chen and U. Aickelin, “Movie Recommendation Systems Using an Artificial Immune System,” *SSRN Electronic Journal*, 2004, doi: 10.2139/ssrn.2832022.
- [18]S. Bangale, A. Haspe, B. Khemani, and S. Malave, “Recipe Recommendation System Using Content-Based Filtering,” *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4102283.
- [19]J. Son and S. B. Kim, “Content-based filtering for recommendation systems using multiattribute networks,” *Expert Systems with Applications*, vol. 89, pp. 404–412, Dec. 2017, doi: 10.1016/j.eswa.2017.08.008.
- [20]Basu, C., Hirsh, H., & Cohen, W, “Automatic Hashtag Recommendation in Social Networking and Microblogging Platforms Using a Knowledge-Intensive Content-based Approach,” *International Journal of Engineering*, vol. 32, no. 8, Aug. 2019, doi: 10.5829/ije.2019.32.08b.06.
- [21]Feature weighting in content based recommendation system using social network analysis, “Social Network Friend Recommendation System Using Semantic Web,” *International Journal of Science and Research (IJSR)*, vol. 5, no. 1, pp. 1156–1161, Jan. 2016, doi: 10.21275/v5i1.nov152976.
- [22]R. Burke, “Recommender Systems: An Introduction, by Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich,” *International Journal of Human-Computer Interaction*, vol. 28, no. 1, pp. 72–73, Jan. 2012, doi: 10.1080/10447318.2012.632301.

- [23]M. Jalali, H. Gholizadeh, and S. A. Hashemi Golpayegani, “An improved hybrid recommender system based on collaborative filtering, content based, and demographic filtering,” *International Journal of Academic Research*, vol. 6, no. 6, pp. 22–28, Nov. 2014, doi: 10.7813/2075-4124.2014/6-6/a.3.
- [24]H. Shubham, “Location-Aware Personalized News Recommendation Based on Behavior and Popularity Technique,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 8, no. 8, pp. 1638–1648, Aug. 2020, doi: 10.22214/ijraset.2020.27921.
- [25]P. Sharma and L. Yadav, “MOVIE RECOMMENDATION SYSTEM USING ITEM BASED COLLABORATIVE FILTERING,” *International Journal of Innovative Research in Computer Science & Technology*, vol. 8, no. 4, Jul. 2020, doi: 10.21276/ijircst.2020.8.4.2.
- [26]P. Sharma and L. Yadav, “MOVIE RECOMMENDATION SYSTEM USING ITEM BASED COLLABORATIVE FILTERING,” *International Journal of Innovative Research in Computer Science & Technology*, vol. 8, no. 4, Jul. 2020, doi: 10.21276/ijircst.2020.8.4.2.
- [27]Y. Deldjoo, M. Elahi, P. Cremonesi, F. Garzotto, P. Piazzolla, and M. Quadrana, “Content-Based Video Recommendation System Based on Stylistic Visual Features,” *Journal on Data Semantics*, vol. 5, no. 2, pp. 99–113, Feb. 2016, doi: 10.1007/s13740-016-0060-9.
- [28]M. Chaudhary, “TF-IDF Vectorizer scikit-learn,” *Medium*, Sep. 17, 2020. <https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a>
- [29]“sklearn.feature_extraction.text.CountVectorizer — scikit-learn 0.20.3 documentation,” *Scikit-learn.org*, 2018. https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html
- [30]Wikipedia Contributors, “Cosine similarity,” *Wikipedia*, Mar. 03, 2019. https://en.wikipedia.org/wiki/Cosine_similarity
- [31]H. Dedhia, “Movie Recommendation and Rating Prediction Using K-Nearest Neighbors,” *The Startup*, Aug. 20, 2020. <https://medium.com/swlh/movie-recommendation-and-rating-prediction-using-k-nearest-neighbors-704ca8ccaff3> (accessed Jan. 14, 2023).

When we looked at and reviewed the prior work on movie recommendation, we came across some works, including collaborative filtering movie recommender systems, swarm optimization recommender systems, m

ORIGINALITY REPORT

18%	10%	6%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	3%
2	Afrin Jaman Bonny, Puja Bhowmik, Md. Shihab Mahmud, Abdus Sattar. "Detecting Fake News in Benchmark English News Dataset Using Machine Learning Classifiers", 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2022 Publication	1%
3	pure.coventry.ac.uk Internet Source	1%
4	Submitted to University of Greenwich Student Paper	1%
5	Submitted to Queen Mary and Westfield College Student Paper	1%

Sudh
17.1.23