

# **MOVIE RECOMMENDATION SYSTEM**

## **A MINI PROJECT REPORT**

*Submitted by*

<i>DEEPIKA E</i>	<i>211419205036</i>
<i>MOHANA PRIYA B</i>	<i>211419205105</i>
<i>MONISHA D</i>	<i>211419205111</i>

*In partial fulfilment for the award of the degree*

*Of*

**BACHELOR OF TECHNOLOGY**

*In*

**INFORMATION TECHNOLOGY**

**PANIMALAR ENGINEERING COLLEGE, POONAMALLEE**

**ANNA UNIVERSITY: CHENNAI 600 025**

**JUNE 2022**



# **MOVIE RECOMMENDATION SYSTEM**

## **A MINI PROJECT REPORT**

*Submitted by*

***DEEPIKA E*** ***211419205036***

***MOHANA PRIYA B*** ***211419205105***

***MONISHA D*** ***211419205111***

*In partial fulfilment for the award of the degree*

*Of*

**BACHELOR OF TECHNOLOGY**

*In*

**INFORMATION TECHNOLOGY**

**PANIMALAR ENGINEERING COLLEGE, POONAMALLEE**

**ANNA UNIVERSITY: CHENNAI 600 025**

**JUNE 2022**

## **BONAFIDE CERTIFICATE**

Certified that this project report “**MOVIE RECOMMENDATION SYSTEM**” is the bonafide work of **DEEPIK.E (211419205036), MOHANA PRIYA.B (211419205105), MONISHA D (211419205111)** Who carried out the project under my supervision.

### **SIGNATURE**

**Dr. M. HELDA MERCY, M.E., Ph.D.,**  
**HEAD OF THE DEPARTMENT**

Department of Information Technology

Panimalar Engineering College

Poonamallee, Chennai - 600 123

### **SIGNATURE**

**Mrs. S. UMA, M.Tech., (Ph.D.,)**  
**SUPERVISOR**

Associate Professor

Department of Information Technology

Panimalar Engineering College

Poonamallee, Chennai - 600 123

Submitted for the project and viva-voce examination held on\_\_\_\_\_

### **SIGNATURE**

**INTERNAL EXAMINER**

### **SIGNATURE**

**EXTERNAL EXAMINER**

## **DECLARATION**

We hereby declare that the project report entitled “**MOVIE RECOMMENDATION SYSTEM**” which is being submitted in partial fulfilment of the requirement of the course leading to the award of the ‘Bachelor of Technology in Information Technology’ in **Panimalar Engineering College An Autonomous Institution, Affiliated to Anna University- Chennai** is the result of the project carried out by me under the guidance and supervision of **Mrs. S.UMA, M.Tech.,(Ph.D.,) Associate Professor in the Department of Information Technology**. I further declared that I or any other person has not previously submitted this mini project report to any other institution/university for any other degree/ diploma or any other person.

**(DEEPIKA.E)**

**(MOHANA PRIYA.B)**

**(MONISHA.D)**

Date:

Place: Chennai

It is certified that this project has been prepared and submitted under my guidance.

**Mrs. S.UMA, M.Tech., (Ph.D.,)**

**(Associate Professor / IT)**

Date:

Place: Chennai

## ACKNOWLEDGEMENT

A project of this magnitude and nature requires kind co-operation and support from many, for successful completion. We wish to express our sincere thanks to all those who were involved in the completion of this project.

Our sincere thanks to **Our Honourable Secretary and Correspondent, Dr. P. CHINNADURAI, M.A., Ph.D.**, for his sincere endeavour in educating us in his premier institution.

We would like to express our deep gratitude and **Our Dynamic Directors, Mrs. C. VIJAYA RAJESHWARI and Dr. C. SAKTHI KUMAR, M.E., Ph.D.**, and **Dr. SARANYA SREE SAKTHI KUMAR, B.E., M.B.A., Ph.D.**, for providing us with the necessary facilities for completion of this project.

We also express our appreciation and gratefulness to **Our Principal Dr. K. MANI, M.E., Ph.D.**, who helped us in the completion of the project. We wish to convey our thanks and gratitude to our head of the department, **Dr. M. HELDA MERCY, M.E., Ph.D.**, Department of Information Technology, for her support and by providing us ample time to complete our project.

We express our indebtedness and gratitude to our staff in charge, **Mrs. S. UMA, M.Tech., (Ph.D.)** Associate Professor Department of Information Technology for his guidance throughout the course of our project. Last, we thank our parents and friends for providing their extensive moral support and encouragement during the course of the project.

## **ABSTRACT**

Now-a-days people are consuming content in form of movies, series, etc. for entertainment. In this modern era, people always look up to entertainment and in that process, they waste their time in searching for movies. Everyone wants to watch good films that have great content. It takes lot of time to search for a movie they like. Recommendation system comes into play in such situations. It helps to people by recommending movies. This paper develops a Movie Recommendation System to recommend movies based on different parameters. The principal objective of the project is to construct a movie recommendation framework to prescribe pictures to users. There are many algorithms that help to build a recommendation system. Here, the Content-based algorithm has been employed to recommend movies based on the similarity with other films by analyzing the content of the movie. To find the similarity, the cosine similarity method has been used. Here, the cosine similarity has been computed by using linear kernel, where the parameters are taken by the result of TF-IDF vectorization. Then the most similar movies are recommended

## **LIST OF FIGURES**

<b>FIGURE NO</b>	<b>NAME</b>	<b>PAGE.NO</b>
3.1	Movie Recommendation System	17
3.2	Data Set	20
3.3	Collection of Movies	20
3.4	Data Preprocessing	21
3.5	Cosine Similarity	25
4.1	Laptop	29
4.2	Ram 8Gb	30
4.3	Core I5	30
4.4	Window 11	31
4.5	Google Colab	32
4.6	Language Python	32
5.1	Flow of Diagram of Movie	34
5.2	Sample Output	41
6.1	Black Box Testing	46
6.2	White Box Testing	47
6.3	White Bos Testing Coverage	48
6.4	Types of White Box Testing	49

## **LIST OF ABBREVIATION**

<b>ABBREVIATION</b>	<b>NAME</b>
RS	Recommendation System.
GIS	Geographic Information System.
PDF	Portable Document Format.
NERS	Neural Engine-Based Recommender System.
TF	Term Frequency.
IDF	Inverse Document Frequency.
MRS	Movie Recommendation System.
ML	Machine Learning.
CBA	Content Based Algorithm.
CBRS	Content Based Recommendation System.
HCI	Human Interaction Computer.



# TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	<b>ABSTRACT</b>	<b>v</b>
	<b>LIST OF FIGURES</b>	<b>vi</b>
	<b>LIST OF ABBREVIATION</b>	<b>vii</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 OVERVIEW OF THE PROJECT	<b>2</b>
	1.2 NEED FOR THE PROJECT	<b>3</b>
	1.3 OBJECTIVE OF THE PROJECT	<b>4</b>
<b>2</b>	<b>LITERATURE SURVEY</b>	<b>5</b>
	2.1 A SURVEY ON MOVIE RECOMMENDATION SYSTEM	<b>6</b>
	2.2 MOVIE RECOMMENDATION SYSTEM USING MACHINE LEARNING	<b>7</b>
	2.3 HYBRID RECOMMENDATIONS AND CONTENT BASED RECOMMENDATION.	<b>8</b>
	2.4 HYBRID MOVIE RECOMMENDATION SYSTEM USING MACHINE LEARNING	<b>9</b>
	2.5 A MOVIE RECOMMENDER SYSTEM: MOVREC USING MACHINE LEARNING TECHNIQUES.	<b>10</b>
	2.6 MOVIE RECOMMENDATION SYSTEM BASED ON USERS' SIMILARITY	<b>11</b>
	2.7 FEASABILITY STUDY	<b>12</b>

<b>3</b>	<b>SYSTEM DESIGN</b>	<b>15</b>
	3.1 EXISTING SYSTEM ARCHITECTURE DESIGN	16
	3.2 PROPOSED SYSTEM ARCHITECTURE DESIGN	17
	3.3 ARCHITECTURE DIAGRAM	17
	3.4 ARCHITECTURE DIAGRAM DESCRIPTION	18
	3.4.1 DATA SET	18
	3.4.2 DATA PREPROCESSING	21
	3.4.3 FEATURE EXTRACTION	23
	3.4.4 USER INPUT	24
	3.4.5 COSINE SIMIARITY	24
	3.4.6 LIST OF MOVIES	26
<b>4</b>	<b>REQUIREMENTS SPECIFICATION</b>	<b>28</b>
	4.1 HARDWARE REQUIREMENTS	29
	4.2 HARWARE REQUIREMENTS DESCRIPTION	29
	4.2.1 LAPTOP	29
	4.2.2 RAM 8Gb	30
	4.2.3 CORE i5	30
	4.2.4 WINDOWS 11	31
	4.3 SOFTWARE REQUIREMENTS	31
	4.4 SOFTWARE REQUIREMENTS DESCRIPTION	32
	4.4.1 GOOGLE COLAB	32
	4.4.2 LANGUAGE-PYTHON	32

<b>5</b>	<b>IMPLEMENTATION</b>	<b>33</b>
	5.1 FLOW DIAGRAM	34
	5.2 FLOW DIAGRAM EXPLANATION	34
	5.3 CODE	35
	5.4 SAMPLE CODE	41
<b>6</b>	<b>TESTING AND MAINTENANCE</b>	<b>42</b>
	6.1 UNIT TESTING	43
	6.1.1 ADVANTAGES OF UNIT TESTING	44
	6.2 BLACK BOX TESTING	45
	6.3 WHITE BOX TESTING	47
<b>7</b>	<b>CONCLUSION AND FUTURE ENHANCEMENT</b>	<b>52</b>
	7.1 CONCLUSION	53
	7.2 FUTURE ENHANCEMENT	53
	<b>REFERNCES</b>	<b>54</b>
	<b>APPENDICES</b>	<b>57</b>

# **CHAPTER – 1**

## **INTRODUCTION**

## 1.1 OVERVIEW OF THE PROJECT

Recommender systems are more popular and increase the production costs for many service providers. Today the world is an over-crowded so that the recommendations are required for recommending products or services. However, recommender systems minimize the transaction costs and improves the quality and decision-making process to users. It is applied in various neighboring areas like information retrieval or human computer interaction (HCI). A RS is a software tool designed to make and deliver suggestions for things or content a user would like to purchase. Using machine learning techniques and various data about individual products and individual users, the system creates an advanced net of complex connections between those products and those people. These are a collection of algorithms used to recommend items to users based on information taken from the user. These systems have become ubiquitous, and can be commonly seen in online stores, movies databases and job finders. It gathers huge amount of information about user's preferences of several items like online shopping products, movies, taxi, TV, tourism, restaurants, etc. It stores information of different ways either positive or negative manner. It captures users review for watched movies, traveled places, and purchased products.

Movie recommendation system design a big problem since other recommendation systems require fast computation and processing service from service providers and product distributors. To recommend movies, first collects the ratings for users and then recommend the top list of items to the target user in addition to this, users can check reviews of other users before watching movie. A different recommendation schemes have been presented includes collaborative filtering, content-based recommender system, and hybrid recommender system. However, several issues are raised with users posted reviews. There are 3 types of recommendation systems

- Popularity based recommendation engine
- Content based recommendation engine
- Collaborative filtering-based recommendation engine.

## 1.2 NEED FOR THE PROJECT

People are often confronted with very large amounts of data, for instance through the internet in an information society. We are asked to make choices that are almost impossible to make without additional information or guidance. Now-a-days people are consuming content in form of movies, series, etc. for entertainment. In this modern era, people always look up to entertainment and in that process, they waste their time in searching for movies. Everyone wants to watch good films that have great content. It takes lot of time to search for a movie they like. Recommendation system comes into play in such situations. It helps to people by recommending movies.

Movie recommendation in portable environment is significantly important for users. A movie recommender has proven to be a powerful tool on providing useful movie suggestions for users. The content-based engine recommends personalized content based on certain predefined parameters. Content-Based methods (or cognitive filtering) on the other hand, use information and metadata about the content to find similarities among them, without incorporating user behaviour in any way. Items similar to those 'accessed 'or 'searched 'by the user are recommended here.

Some approaches analyze the audio and visual features as in using image and signal processing techniques while some analyze textual features via Natural Language Processing methods like TF-IDF, as in, and word2vec, as in. The major difference between collaborative filtering and content-based recommender systems is that the former only uses the user item ratings to make recommendations, while the latter relies on the features of users and items for predictions. Content based recommendation engine takes in a movie that a user currently likes as input. Then it analyzes the contents of the movie to find out other movies which have similar content. Then it ranks similar movies according to their similarity scores and recommends the most relevant movies to the user.

### **1.3 OBJECTIVE OF THE PROJECT**

The main purpose of our recommender system is to provide suggestions and recommendations which are truly based on customer's preference and choice. Recommendation system (RS) helps customers not only finding appropriate movies but also it is benefitting in all domains such as hostels, books, and all sorts of other different products and items. Different types of data i.e., hotels, movies, and music, can be processed by RS.

The main idea here is to develop a recommender system which helps users to find movies according to their preference and choices. When we are dealing with the recommending a movie to the user, we mainly focus on the movie given by the user he/she was interested in. Based on the preferences we select which type of genre is mostly liked by the user. Using this user's genre choice, we will recommend movies which contain genre the user is interested.

The project focuses on a movie recommendation engine that filters the data using different algorithms and recommends the most relevant items to users. We have created our model by reusing the saved file to get recommendations and users have to search the movie name for a year and he/she will get the movie recommendations. The principal objective of the project is to construct a movie recommendation framework to prescribe suggestions to users. There are many algorithms that help to build a recommendation system. Here, the Content-based algorithm has been employed to recommend movies based on the similarity with other films by analyzing the content of the movie. The cosine similarity method has been used; it has been computed where the parameters are taken by TF-IDF vectorization. Then the most similar movies are recommended.

## **CHAPTER - 2**

### **LITERATURE SURVEY**



## **2.1 A Survey on Movie Recommendation System**

**Author:** Kim Mucheol, Noguera, Nanou, Ruotsalo.

**Year:**2019

In this paper, dimensional GIS architecture is designed and implemented in the RS. This RS grants tourists to take advantage from novel characteristics such as a 3D map-based interface. Evaluation of user experience is also presented in this work. This paper projected a theoretical model and sample of the projected interactive movie RS. This proposed method constructs adapted suggestion of movies in online community systems. The presented method develops the grouping conscious community network model which is considered as the approach that is capable to confine the dynamics of socially-mediated information communicated in communal networks. This proposed replica may examine the fondness of user methodically and which can show the quick and constant change in social network. This paper described some problems associated to the presentation of recommendations in movie domain.

The present work shows the survey of former techniques and other popular RS which are focused on user opinion and approval. In this paper, different methods have been compared. The most effective method related to the user opinion and approval is “planned outline” and the “textbook and videotape” interfaces, as a strong constructive association was also originated between client opinion and approval in all experimental situations. This system employed semantic network speech in the form of facts illustration. Ontologies are basically used to make connection between semantic break, sensor inputs, and user profiles. An information retrieval framework is used in the RS to get suitable content for mobile user. Its result indicate that the system is capable to meets the user requirement.

## **2.2 Movie Recommendation System Using Machine Learning**

**Author:** Sang-Min Choi, Muyeed Ahmed, S. Rajarajeswari, Jiang Zhang.

**Year:**2020

This paper mentioned about the shortcomings of collaborative filtering approach like sparsity problem or the cold-start problem. In order to avoid this issue, the authors have proposed a solution to use category information. The authors have proposed a movie recommendation system which is based on genre correlations. The authors stated that the category information is present for the newly created content. Thus, even if the new content does not have enough ratings or enough views, still it can pop up in the recommendations list with the help of category or genre information. The proposed solution is unbiased over the highly rated most watched content and new content which is not watched a lot. Hence, even a new movie can be recommended by the recommendation system. This paper proposed a solution using K-means clustering algorithm. Authors have separated similar users by using clusters. Later, the authors have created a neural network for each cluster for recommendation purpose.

The proposed system consists of steps like Data Pre-processing, Principal Component Analysis, Clustering, Data Pre-processing for Neural Network, and Building Neural Network. User rating, user preference, and user consumption ratio have been taken into consideration. After clustering phase, for the purpose of predicting the ratings which the user might give to the unwatched movies, the authors have used neural network. Finally, recommendations are made with the help of predicted high ratings. This paper discussed about Simple Recommender System, Content-based Recommender System, Collaborative Filtering based Recommender System and finally proposed a solution consisting of Hybrid Recommendation System.

## **2.3 Hybrid Recommendations and Content Based recommendation.**

**Author:** Sharma, Tekin, Beel et al, Chapphannarungsri and Maneero.

**Year:**2021

This paper reviewed the several approaches used for RS. Approaches may be categorized into three parts CF, Hybrid Recommendations and Content Based recommendation. Also, this paper describes the merits and demerits of the recommendation approaches. proposed distributed online learning in social RSs. In this work, things which are suggested to the user depends their query. Also, item issuggested according to the background history of things which was bought earlier, its gender, and age. In this work decentralized sequential decision making is considered. introduced the architecture of the RS and four datasets.

The components like crawling PDF, generating use model etc are included in the architecture of the system. Moreover, the architecture including content-based recommendation for calculating purpose. an advanced RS has been presented which gives good quality of recommendations. Additionally, a method has been proposed for Multiple Criteria approach which can change the means of weighting to be more appropriate and also unease about the occurrence of the selection film features. The Multiple Linear Regression is applied to do Multidimensional technique which can study the techniques. This system may have other troubles such as sparsity and over specialization. Cold start problem is that issue in which client cannot able to draw the inferences for items for which it does not have adequate information. This paper presented some novel concepts which helps in choosing the information to use. It also helps to choose the technique for recognizing which appropriate information give the discrepancy ratings rely on the statistical rating.

## **2.4 Hybrid Movie Recommendation System Using Machine Learning**

**Author:** Md. Akter Hossain, Harper, V. Subramaniaswamy, Debashis Das.

**Year:**2021

This paper proposed NERS which is an acronym for neural engine-based recommender system. The authors have done a successful interaction between 2 datasets carefully. Moreover, the authors stated that the results of their system are better than the existing systems because they have incorporated the usage of general dataset as well as the behaviour-based dataset in their system. The authors have used 3 different estimators in order to evaluate their system against the existing systems. This paper has proposed a solution of personalized movie recommendation which uses collaborative filtering technique. Euclidean distance metric has been used in order to find out the most similar user. The user with least value of Euclidean distance is found. Finally, movie recommendation is based on what that particular user has best rated. The authors have even claimed that the recommendations are varied as per the time so that the system performs better with the changing taste of the user with time. This was a survey paper on recommendation systems.

The authors mentioned about Personalized recommendation systems as well as non-personalized systems. User based collaborative filtering and item based collaborative filtering was explained with a very good example. The authors have also mentioned about the merits and demerits of different recommendation systems. This paper mentioned the details about the MovieLens Dataset in their research paper. This dataset is widely used especially for movie recommendation purpose. There are different versions of dataset available like MovieLens 100K / 1M / 10M / 20M / 25M /1B Dataset. The dataset consists of features like user id, item id / movie id, rating, timestamp, movie title, IMDb URL, release date, etc. along with the movie genre information.

## **2.5 A Movie Recommender System: MOVREC using Machine Learning Techniques.**

**Author:** Rattanajit, Odic, Li and Yamada, Christakou.

**Year:**2020

In this paper, inductive learning algorithm has been proposed and this algorithm is then applied to the recommendation process. In this work, decision tree has been constructed instead of using user user similarity. This decision tree shows the user preference. It can be said that suggestions are performed by decision tree categorization. The results show that the presented technique is appropriate for the explanation of very large-scale issues and high-quality suggestions can be estimated. presented pseudo ratings which is relying on multi criteria and also focuses on the related information as multidimensional. The multi regression is useful to examine the appropriate information of client in order to include multidimensional.

According to the evaluation of experiment, the RS is created on movie domain known as Modernize Movie and demonstrates that the multi criterion pseudo ratings and multi-dimensional client report increase the worth and accurateness of recommender outcome. presented comparisons between two techniques. These two techniques are: the pertinent evaluation from the consumer assessment and the pertinent finding with statistical testing on the rating data. By utilizing these two methods, it can be observed that is it possible for the user to forecast which circumstance manipulates their choice and which technique guides to superior recognition of the pertinent appropriate information. a clustering technique has been proposed which is depending on semi supervised learning. In this paper, presented approach is utilized to create a system for recommending movies which merge collaborative and content-based information. The presented system is checked on the Movie Lens DS, providing recommendations of high precision.

## **2.6 MOVIE RECOMMENDATION SYSTEM BASED ON USERS' SIMILARITY**

**Author:** Gaurav Arora, Ashish Kumar, Gitanjali Sanjay Devre, Prof. Amit Ghumare

**Year:**2014

Most of the online recommendation systems for a variety of items use ratings from previous users to make recommendations to current users with similar interests. One such system was designed by Jung, Harris, Webster and Herlocker (2004) for improving search results. The system encourages users to enter longer and more informative search queries, and collects ratings from users as to whether search results meet their information need or not. These ratings are then used to make recommendations to later users with similar needs.

In particular, we've chosen to explore the movie niche as this is an area where our project can provide significant improvements compared to existing products and systems. Traditional movie websites (IMDB, AOL Movies) function by proving global user ratings on movies in their database. Movies are categorized by metadata such as genre, era, directors, and so on. Users can search for movies, browse lists and read reviews written by critics or other users. However, most of these services lack any personal recommendation system and haven't taken advantage of social-networking communities or crowd wisdom. Some websites, such as Blockbuster, do provide individualized recommendations based on a user's ratings but do not include any social networking component. Yahoo! Movies goes further and uses personal ratings to suggest movies currently playing in theatre, on TV, and out on DVD. It also draws upon its vast user base to give lists of similar movie fans, their ratings, and reviews. Other movie sites, like Flixster, take a different approach. Flixster forms web-based communities around movies and suggests movies to watch based on what your friends have rated.

## **2.7 FEASIBILITY STUDY**

A feasibility study is carried out to select the best system that meets performance requirements. The main aim of the feasibility study activity is to determine that it would be financially and technically feasible to develop the product. The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Economic feasibility, technical feasibility and Operational feasibility.

### **Economic Feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased. The fund required by the company for the research and development of the system. The expenditures faced by the company. For the developed system to be within the budget, the technologies which are freely available and the customized products are to be used.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

The system is economically feasible because it considers all the pros and cons of the construction and the implementation of the project and hence it is considered as feasible economically.

## **Technical Feasibility**

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed. The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Through the technology may become obsolete after some period of time, due to the fact that

never version of same software supports older versions, the system may still be used. So, there are minimal constraints involved with this project. As the system has been developed using Java the project is technically feasible. This study is carried out to check the technical feasibility, that is, the technical Requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands being placed on the client. The developed system must have modest requirements, as only minimal or null changes are required for implementing this system.

## **Social Feasibility**

This study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. The training process for the users becomes so easier while this system is into the use and so the acceptance level also is higher and the system will be familiar and so this project is also socially feasible.



## **Operational Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

## **CHAPTER – 3**

### **SYSTEM DESIGN**

### 3.1 EXISTING SYSTEM ARCHITECTURE DESIGN

- Recommendation systems are software applications that suggest or recommend movies or product to users.
- TYPES:

**CONTENT BASED:** It is also known as cognitive filtering. It provides recommendation by comparing representation of content describing an item or a product to representation of the content describing the interest to the user. It is suitable in situation or domains where items are more than users. The goal behind content-based filtering is to classify products with specific keywords, learn what the customer likes, look up those terms in the database, and then recommend similar things

**POPULARITY BASED:** It is used to check movies that are in current trend or most popular among the users and it directly recommended it.

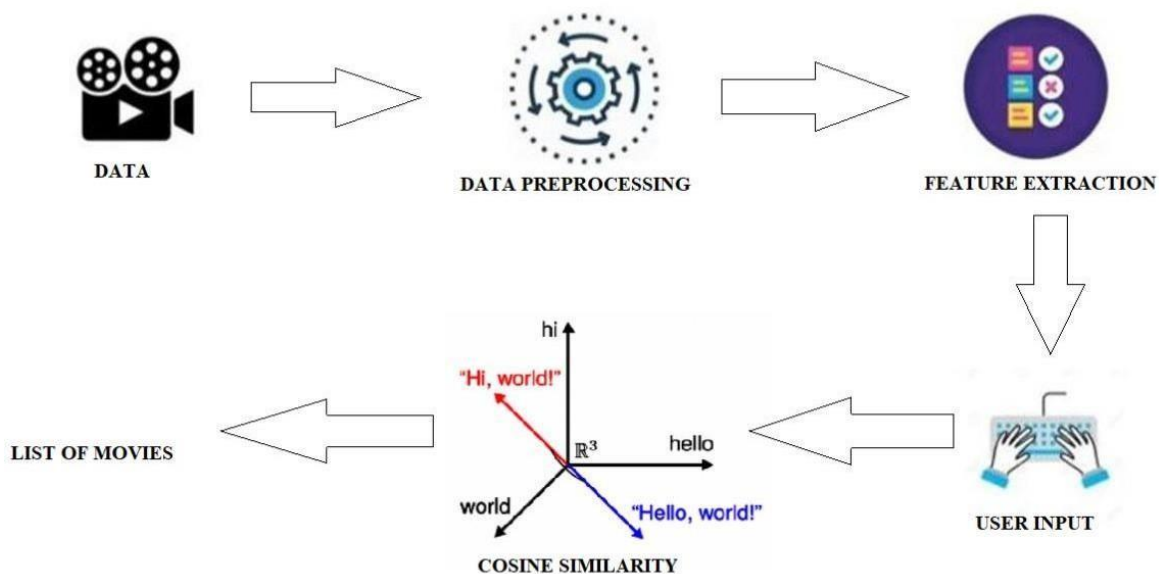
**COLLABORATIVE BASED:** It is a family of algorithms where there are multiple ways to calculate rating based on ratings of similar users i.e., previous data collected from other individuals. The basic premise of such systems is that the users 'previous data should be sufficient to generate a prediction.

- We are Creating a system in python where user can give name of this favourite movies and based on this input, we are going to recommend certain movies to them.
- In this we are going to do content based and certain kind of popularity based.

## 3.2 PROPOSED SYSTEM ARCHITECTURE DESIGN

With our proposed system we aim at building a system that gives more accurate results. Using Python language, we aim at creating a source code that is also compatible with our GUI. With the help Machine learning library like NumPy and Panda, we are making the system that can do mathematical dimensional array and matrices calculation on its own. In our project we use Python language as the main source code. The database which we are going to use contains Movie information and Ratings and as per that given information the system is going to give a recommendation, our system is going to start giving recommendations to the users which will be the final phase of our system.

## 3.3 ARCHITECTURE DIAGRAM



**Fig 3.1: Movie Recommendation system**

## 3.4 ARCHITECTURE DIAGRAM DESCRIPTION

The basic block diagram of the “**MOVIE RECOMMENDATION SYSTEM**” is shown in the above figure. This block diagram consists of the following essential blocks:

- Data Set
- Data Preprocessing
- Feature Extraction
- User Input
- Cosine-Similarity
- List of Movies

### 3.4.1 DATASET

Data collection is considered as the foundation of the Machine Learning model building. Without data, the concept of building a Machine Learning model is futile. The more data we have the better predictive model we can build out of it. But remember, ‘more data’ does not mean a bunch of irrelevant data.

We cannot add any data just to increase the quantity. So, we can say that any effort that is directed toward ‘finding the right data’ is well invested—that way after putting the collected data through a cleansing process, we will have ‘more data’ to build the model with. Now, I am sure that you must be wondering how we can find dataset for machine learning operations.

Dataset for machine learning can be found in two formats—structured and unstructured. Where the structured datasets are in tabular format in which the row of the dataset corresponds to record and column corresponds to the features, and unstructured datasets corresponds to the images, text, speech, audio, etc.

The dataset contains list of movies there will be around 5000+ movies we need to compare movie names and find which movie closet to the one given by user. There are several datasets available to build a movie recommendation system. But for this project, we are going to use a dataset that contains the metadata (index, genre, budget, keyword, id, title etc..) of the movie.

Data imbalance-Some classes or categories in the data may have a disproportionately high or low number of corresponding samples. As a result, they risk being under-represented in the model.

## **Limitation of Datasets**

Finding a quality dataset is a fundamental requirement to build the foundation of any real-world AI application. However, the real-world datasets are complex, messier, and unstructured. The performance of any Machine Learning or Deep Learning model depends on the quantity, quality, and relevancy of the dataset. It's not an easy task to find the right balance.

## **Types of Datasets**

In Machine Learning while training a model we often encounter the problem of overfitting and underfitting.

In order to overcome the situation, we need to divide our dataset into 3 different parts:

- Training Dataset
- Validation Dataset
- Test Dataset

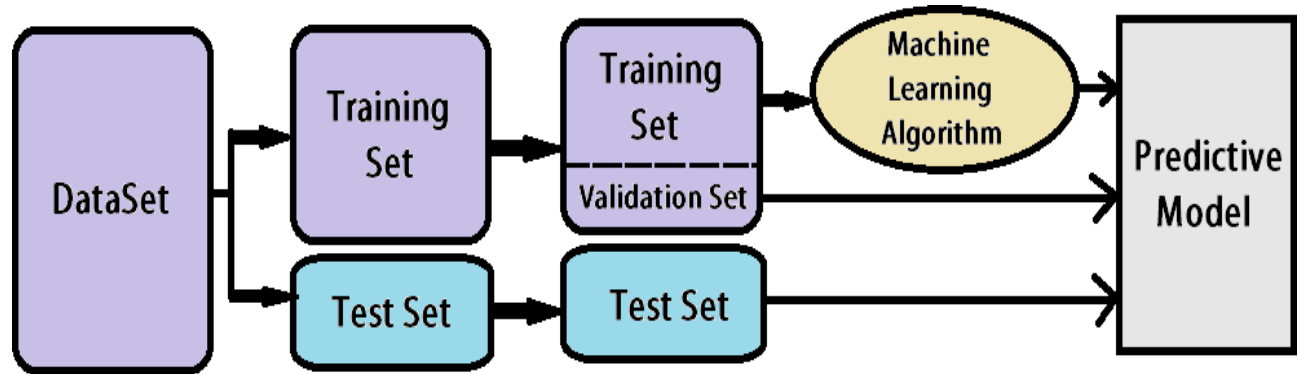


Fig 3.2: Data set

## Movies Dataset (CSV File)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	index	budget	genres	homepage	id	keywords	original_lang	original_title	overview	popularity	production_budget	production_release_date	revenue	runtime	spoken_lang	status	tagline	title	vote_average	vote_count	
2	0	237000000	Action Adventure	http://www...	19995	culture clash	en	Avatar	In the 22nd	150.43758	1012300000	2009-12-10	2788000000	162	iso_639_3	Released	Enter the W Avatar		7.2	11	
3	1	300000000	Adventure Fantasy	http://disne...	285	ocean drug	en	Pirates of the Captain Jack		139.08262	1012300000	2007-05-19	961000000	169	iso_639_3	Released	At the end of the world	Pirates of the Caribbean: The Curse of the Black Pearl	6.9	4	
4	2	245000000	Action Adventure	http://www...	206647	spy based on	en	Spectre	A cryptic me	107.37679	1012300000	2015-10-26	880674609	148	iso_639_3	Released	A Plan No 0 Spectre		6.3	4	
5	3	250000000	Action Crime	http://www...	49026	dc comics c	en	The Dark Knight Following the		112.31295	1012300000	2012-07-16	1085000000	165	iso_639_3	Released	The Legend The Dark Knight		7.6	9	
6	4	260000000	Action Adventure	http://movi...	49529	based on nc	en	John Carter John Carter		43.926995	1012300000	2012-03-07	284139100	132	iso_639_3	Released	Lost in our John Carter		6.1	2	
7	5	258000000	Fantasy Action	http://www...	559	dual identity	en	Spider-Man The seeming		115.69981	1012300000	2007-05-01	890871626	139	iso_639_3	Released	The battle v Spider-Man		5.9	3	
8	6	260000000	Animation Fantasy	http://disne...	38757	hostage ma	en	Tangled	When the ki	48.681969	1012300000	2010-11-24	591794936	100	iso_639_3	Released	They're taki Tangled		7.4	3	
9	7	280000000	Action Adventure	http://marv...	99861	marvel com	en	Avengers: A When Tony		134.27923	1012300000	2015-04-22	1405000000	141	iso_639_3	Released	A New Age I Avengers: A		7.3	6	
10	8	250000000	Adventure Fantasy	http://harry...	767	witch magic	en	Harry Potter: As Harry bej		98.885637	1012300000	2009-07-07	933959197	153	iso_639_3	Released	Dark Secret: Harry Potte		7.4	5	
11	9	250000000	Action Adventure	http://www...	209112	dc comics v	en	Batman v Sl Fearing the		155.79045	1012300000	2016-03-23	873260194	151	iso_639_3	Released	Justice or re Batman v Sl		5.7	7	
12	10	270000000	Adventure Fantasy	http://www...	1452	saving the v	en	Superman R Superman r		57.925623	1012300000	2006-06-28	391081192	154	iso_639_3	Released	Superman R		5.4	1	
13	11	200000000	Adventure Fantasy	http://www...	10764	killing under	en	Quantum of Quantum of		107.92881	1012300000	2008-10-30	586090727	106	iso_639_3	Released	For love, fo Quantum of		6.1	2	
14	12	200000000	Adventure Fantasy	http://disne...	58	witch fortu	en	Pirates of the Captain Jack		145.84738	1012300000	2006-06-20	1066000000	151	iso_639_3	Released	Jack is back Pirates of th		7	5	
15	13	255000000	Action Adventure	http://disne...	57201	texas horse	en	The Lone Ri: The Texas R		49.046956	1012300000	2013-03-07	89289910	149	iso_639_3	Released	Never Take The Lone Ri		5.9	2	
16	14	225000000	Action Adventure	http://www...	49521	saving the v	en	Man of Steel A young boy		99.398009	1012300000	2013-06-12	662845518	143	iso_639_3	Released	You will bell Man of Steel		6.5	6	
17	15	225000000	Adventure Family Fantas		2454	based on nc	en	The Chronic One year af		53.978602	1012300000	2008-05-15	419651413	150	iso_639_3	Released	Hope has a The Chronic		6.3	1	
18	16	220000000	Science Fict	http://marv...	24428	new york sh	en	The Avenger: When an un		144.44863	1012300000	2012-04-25	1520000000	143	iso_639_3	Released	Some assen The Avenge		7.4	11	
19	17	380000000	Adventure Fantasy	http://disne...	1865	sea captain	en	Pirates of the Captain Jack		135.41386	1012300000	2011-05-14	1046000000	136	iso_639_3	Released	Live Forever Pirates of th		6.4	4	
20	18	225000000	Action Com	http://www...	41154	time travel	en	Men in Blac Agents J Wi		52.035179	1012300000	2012-05-23	624026776	106	iso_639_3	Released	They are ba Men in Blac		6.2	4	
21	19	250000000	Action Adventure	http://www...	122917	corruption	en	The Hobbit: Immediate		120.96574	1012300000	2014-12-10	956019788	144	iso_639_3	Released	Witness the The Hobbit:		7.1	4	
22	20	215000000	Action Adventure	http://www...	1930	loss of fath	en	The Amazing Peter Parke		89.866276	1012300000	2012-06-27	752215857	136	iso_639_3	Released	The untold s The Amazin		6.5	6	
23	21	200000000	Action Adventure	http://www...	20662	robin hood	en	Robin Hood When soldie		37.668301	1012300000	2010-05-12	310669540	140	iso_639_3	Released	Rise and ris Robin Hood		6.2	1	
24	22	250000000	Adventure Fantasy	http://www...	57158	elves dwarv	en	The Hobbit: The Dwarve		94.370564	1012300000	2013-12-11	958400000	161	iso_639_3	Released	Beyond darl The Hobbit:		7.6	4	
25	23	180000000	Adventure Fantasy	http://www...	2268	england cor	en	The Golden After overh		42.990906	1012300000	2007-12-04	372234864	113	iso_639_3	Released	There are w The Golden		5.8	1	
26	24	207000000	Adventure Drama Action		254	film busin	en	King Kong In 1933 Nev		61.22601	1012300000	2005-06-14	550000000	187	iso_639_3	Released	The eighth v King Kong		6.6	2	
27	25	200000000	Drama Rom	http://www...	597	shipwreck	ic	Titanic	84 years lat	100.0259	1012300000	1997-11-18	1845000000	194	iso_639_3	Released	Nothing on Titanic		7.5	7	
28	26	250000000	Adventure Fantasy	http://marv...	271110	civil war wa	en	Captain Am Following th		198.37239	1012300000	2016-04-27	1153000000	147	iso_639_3	Released	Divided We Captain Am		7.1	7	
29	27	209000000	Thriller Action Adventur		44833	fight u.s. na	en	Battleship When mark		64.928382	1012300000	2012-04-11	303025485	131	iso_639_3	Released	The Battle f Battleship		5.5	2	
30	28	150000000	Action Adventure	http://www...	135397	monster dvi	en	Jurassic Wo Twenty-tw		418.70855	1012300000	2015-06-09	1514000000	124	iso_639_3	Released	The park is Jurassic Wo		6.5	8	

Fig 3.3 Collection of movies

### 3.4.2 DATA PREPROCESSING

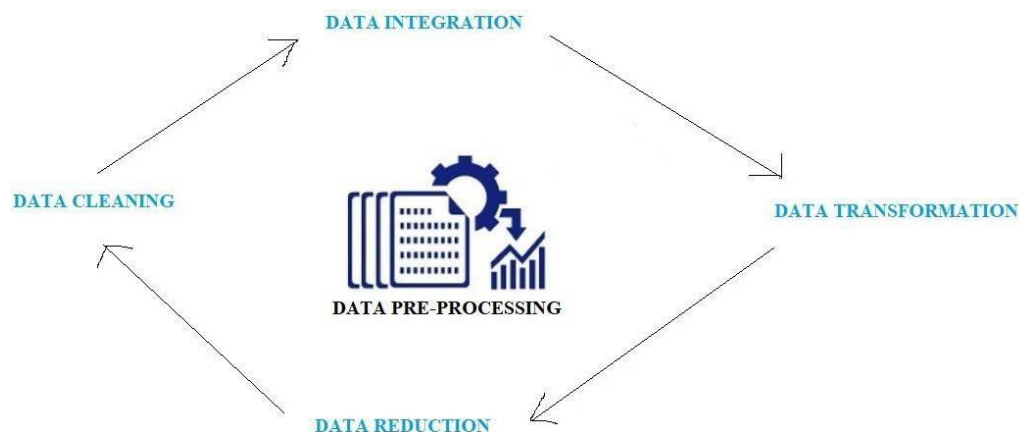
Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data preprocessing task.

Why do we need Data Preprocessing?

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

#### 4 Steps in Data Preprocessing



**Fig 3.4: Data Preprocessing**



## **Data Cleaning**

Data Cleaning is particularly done as part of data preprocessing to clean the data by filling missing values, smoothing the noisy data, resolving the inconsistency, and removing outliers.

- Missing values
- Noisy Data
- Removing outliers

## **Data Integration**

Data Integration is one of the data preprocessing steps that are used to merge the data present in multiple sources into a single larger data store like a data warehouse.

Data Integration is needed especially when we are aiming to solve a real-world scenario like detecting the presence of nodules from CT Scan images. The only option is to integrate the images from multiple medical nodes to form a larger database

## **Data Transformation**

Once data clearing has been done, we need to consolidate the quality data into alternate forms by changing the value, structure, or format of data using the below-mentioned Data Transformation strategies.

- Generalization
- Normalization
- Attribute Selection
- Aggregation

## **Data Reduction**

The size of the dataset in a data warehouse can be too large to be handled by data analysis and data mining algorithms. One possible solution is to obtain a reduced representation of the dataset that is much smaller in volume but produces the same quality of analytical results.

Here is a walkthrough of various Data Reduction strategies.

- Data cube aggregation
- Dimensionality reduction
- Data compression

### **3.4.3 FEATURE EXTRACTION**

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. It yields better results than applying machine learning directly to the raw data. It is used to convert textual data into numerical values (feature vectors). we have textual data (movie description) in form of text. we need to convert into meaningful numbers for that we need this TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a widely used statistical method in natural language processing and information retrieval. It measures how important a term is within a document relative to a collection of documents (i.e., relative to a corpus). Words within a text document are transformed into importance numbers by a text vectorization process.

There are many different text vectorizations scoring schemes, with TF-IDF being one of the most common. As its name implies, TF-IDF vectorizes/scores a word by

multiplying the word's Term Frequency (TF) with the Inverse Document Frequency (IDF).

Term Frequency: TF of a term or word is the number of times the term appears in a document compared to the total number of words in the document.

Inverse Document Frequency: IDF of a term reflects the proportion of documents in the corpus that contain the term. Words unique to a small percentage of documents (e.g., technical jargon terms) receive higher importance values than words common across all documents (e.g., a, the, and).

The TF-IDF of a term is calculated by multiplying TF and IDF scores

$$\text{TF-IDF} = \text{TF} * \text{IDF}$$

#### **3.4.4 USER INPUT**

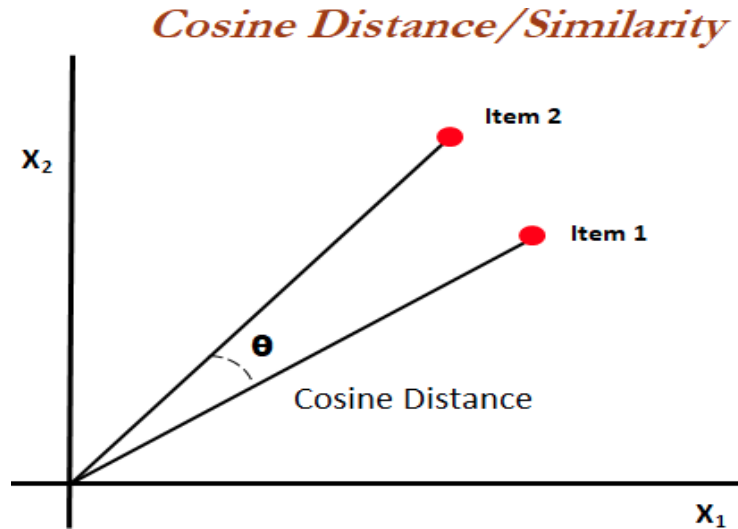
The user should enter the “NAME OF THE MOVIE”.

EXAMPLE:

Enter your favourite movie name: ironman

#### **3.4.5 COSINE SIMILARITY**

Cosine Similarity measures the cosine of the angle between two non-zero vectors of an inner product space. This similarity measurement is particularly concerned with orientation, rather than magnitude. In short, two cosine vectors that are aligned in the same orientation will have a similarity measurement of 1, whereas two vectors aligned perpendicularly will have a similarity of 0. If two vectors are diametrically opposed, meaning they are oriented in exactly opposite directions (i.e. back-to-back), then the similarity measurement is -1.



**Fig 3.5: Cosine Similarity**

### How does 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:

$$\mathbf{A} \cdot \mathbf{B} = \|\mathbf{A}\| \|\mathbf{B}\| \cos \theta$$

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

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

The output will produce a value ranging from -1 to 1, indicating similarity where -1 is non-similar, 0 is orthogonal (perpendicular), and 1 represents total similarity.

Machine learning uses Cosine Similarity in applications such as data mining and information retrieval. For example, a database of documents can be processed such that each term is assigned a dimension and associated vector corresponding to the frequency of that term in the document. This allows for a Cosine Similarity measurement to distinguish and compare documents to each other based upon their similarities and overlap of subject matter.

We need to find similarity of all movies we will do that by cosine similarity it will give some similarity score for all different movies compared to other movies we use this to recommended. Similarity score is numeric value which range from 0 to 1.

### **3.4.6 LIST OF MOVIES**

The data that filter the movies given by the user.

- Iron Man
- Iron Man 2
- Iron Man 3
- Avengers: Age of Ultron
- The Avengers
- Captain America: Civil War
- Captain America: The Winter Soldier
- Ant-Man
- X-Men
- Made
- X-Men: Apocalypse
- X2
- The Incredible Hulk

- The Helix... Loaded
- X-Men: First Class
- X-Men: Days of Future Past
- Captain America: The First Avenger
- Kick-Ass 2
- Guardians of the Galaxy
- Deadpool
- Thor: The Dark World
- G-Force
- X-Men: The Last Stand
- Duets
- Mortdecai
- The Last Airbender
- Southland Tales
- Zither: A Space Adventure
- Sky Captain and the World of Tomorrow.

## **CHAPTER-4**

### **REQUIREMENTS SPECIFICATION**

## 4.1 HARDWARE REQUIREMENTS

- Laptop
- Ram 8Gb
- Core i5
- Windows 11

## 4.2 HARDWARE REQUIREMENTS DESCRIPTION:

### 4.2.1 LAPTOP



**Fig 4.1 Laptop**

**Laptops** combine all the input/output components and capabilities of a desktop computer, including the display screen, small speakers, a keyboard, datastorage device, sometimes an optical disc drive, pointing devices with an operating system, a processor and memory into a single unit. Most modern laptops feature integrated webcams and built-in microphones, while many also have touchscreens. Laptops can be powered either from an internal battery or by an external powersupply from an ACadapter



### 4.2.1 Ram 8Gb



**Fig 4.2: Ram 8Gb**

**Random-access memory (RAM)** is a form of computer memory that can be read and changed in any order, typically used to store working data and machine code. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory.

### 4.2.1 CORE I5



**Fig 4.3 Core I5**

**Intel Core** is a line of streamlined midrange consumer, workstation and enthusiast computer central processing units (CPUs) marketed by Intel Corporation. These processors displaced the existing mid- to high-end Pentium processors at the time of their introduction, moving the Pentium to the entry level. Identical or more capable versions of Core processors are also sold as Xeon processors for the server and workstation markets.

### 4.2.2 WINDOWS 11



**Fig 4.4: Windows 11**

**Windows 11** is the latest major release of Microsoft's Windows NT operating system, released in October 2021. It is a free upgrade to its predecessor, Windows 10 (2015), available for any Windows 10 devices that meet the new Windows 11 system requirements. Upon release, it was praised for its improved visual design, window management, and a stronger focus on security, but was criticized for various modifications to aspects of its user interface which were seen as worse than its predecessor.

## 4.3 SOFTWARE REQUIREMENTS

- Google Colab
- Language-Python

## **4.3 SOFTWARE REQUIREMENTS DESCRIPTION**

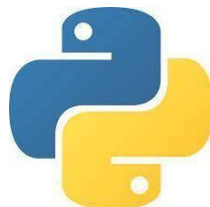
### **4.4.1 GOOGLE COLAB**



**Fig 4.5: Google Colab**

Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

### **4.4.2 LANGUAGE-PYTHON**



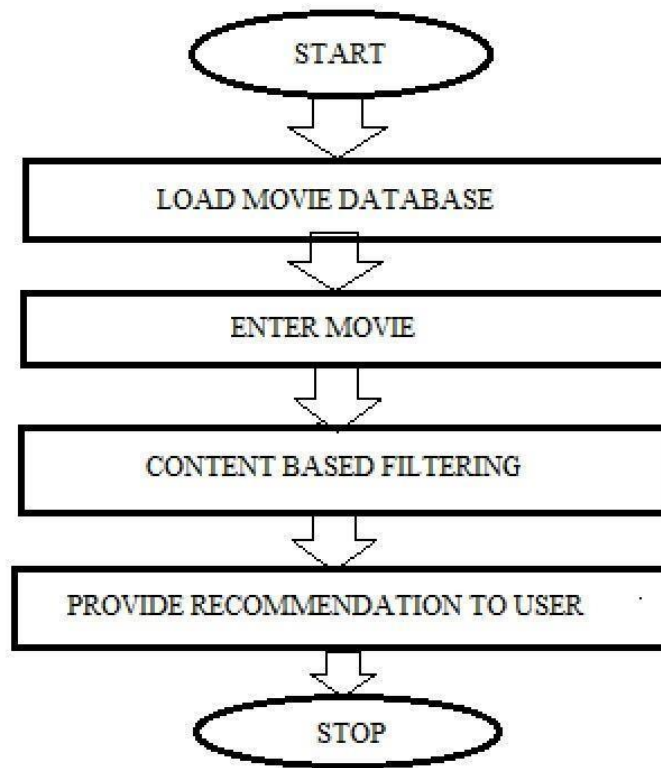
**Fig 4.6: Language Python**

Python is a high-level general-purpose programming language which is used for creating programs with mathematical source codes with easy readability and faster usability, its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

## **CHAPTER – 5**

### **IMPLEMENTATION**

## 5.1 FLOW DIAGRAM



**Fig 5.1 Flow Diagram of Movie Recommendation system**

## 5.2 FLOW DIAGRAM EXPLANATION:

- First, we have to start the process
- Then we have to load the movies in database and by using our moviesdataset file we will compare movie names and find which movie is closest to the one given by user.
- After that, the user should give the name of his/her favourite movie.
- Then the content-based filtering starts its process by finding the movies.
- Finally, by using cosine similarity function we have provided similar movies which was given by the user
- Stop the process.

## 5.3 CODE

### Importing the dependencies

```
import numpy as np
import pandas as pd
import difflib
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
```

### Data Collection and Pre-Processing

```
# loading the data from the csv file to apandas dataframe
movies_data = pd.read_csv('/content/movies.csv')
```

```
# printing the first 5 rows of the dataframe
movies_data.head()
```

index	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_companies	production_countri	
0	0	237000000	Action Adventure Fantasy Science Fiction	http://www.avatarmovie.com/	19995	culture clash future space war space colony so...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "Ingenious Film Partners", "id": 289...}	[{"iso_3166_1": "U", "name": "United States of America"}]
1	1	300000000	Adventure Fantasy Action	http://disney.go.com/disneypictures/pirates/	285	ocean drug abuse exotic island east india trad...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"name": "Walt Disney Pictures", "id": 2}, {"name": "United States of America"}]	[{"iso_3166_1": "U", "name": "United States of America"}]
2	2	245000000	Action Adventure Crime	http://www.sonypictures.com/movies/spectre/	206647	spy based on novel secret agent sequel mi6	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	[{"name": "Columbia Pictures", "id": 5}, {"name": "United Kingdom"}]	[{"iso_3166_1": "GB", "name": "United Kingdom"}]
3	3	250000000	Action Crime Drama Thriller	http://www.thedarkknighttrises.com/	49026	dc comics crime fighter terrorist secret	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"name": "Legendary Pictures", "id": 923}, {"name": "United States of America"}]	[{"iso_3166_1": "U", "name": "United States of America"}]

production_countries	release_date	revenue	runtime	spoken_languages	status	tagline	title	vote_average	vote_count	cast	crew	director
[{"iso_3166_1": "US", "name": "United States o..."}]	2009-12-10	2787965087	162.0	[{"iso_639_1": "en", "name": "English"}, {"iso_639_1": "en", "name": "English"}]	Released	Enter the World of Pandora.	Avatar	7.2	11800	Sam Worthington Zoe Saldana Sigourney Weaver S...	[{"name": "Stephen E. Rivkin", "gender": "0", "departm..."}]	James Cameron
[{"iso_3166_1": "US", "name": "United States o..."}]	2007-05-19	961000000	169.0	[{"iso_639_1": "en", "name": "English"}]	Released	At the end of the world, the adventure begins.	Pirates of the Caribbean: At World's End	6.9	4500	Johnny Depp Orlando Bloom Keira Knightley Stel...	[{"name": "Dariusz Wolski", "gender": "2", "departm..."}]	Gore Verbinski
[{"iso_3166_1": "GB", "name": "United Kingdom"...}]	2015-10-26	880674609	148.0	[{"iso_639_1": "fr", "name": "Fran\u00e7ais"}, {"iso_639_1": "en", "name": "English"}]	Released	A Plan No One Escapes	Spectre	6.3	4466	Daniel Craig Craig Christoph Waltz L\u00e9a Seydoux ...	[{"name": "Thomas Newman", "gender": "2", "departm..."}]	Sam Mendes
[{"iso_3166_1": "US", "name": "United States o..."}]	2012-07-16	1084939099	165.0	[{"iso_639_1": "en", "name": "English"}]	Released	The Legend Ends	The Dark Knight Rises	7.6	9106	Christian Bale Michael Caine Gary Oldman Anne ...	[{"name": "Hans Zimmer", "gender": "2", "departm..."}]	Christopher Nolan



# number of rows and columns in the data frame

```
movies_data.shape
```

```
(4803, 24)
```



# selecting the relevant features for recommendation

```
selected_features = ['genres', 'keywords', 'tagline', 'cast', 'director']
print(selected_features)
```

```
['genres', 'keywords', 'tagline', 'cast', 'director']
```



# replacing the null values with null string

```
for feature in selected_features:
    movies_data[feature] = movies_data[feature].fillna('')
```



# combining all the 5 selected features

```
combined_features = movies_data['genres']+' '+movies_data['keywords']+' '+movies_data['tagline']+' '+movies_data['cast']+' '+movies_data['director']
```

```
▶ print(combined_features)
```

```
0      Action Adventure Fantasy Science Fiction cultu...
1      Adventure Fantasy Action ocean drug abuse exot...
2      Action Adventure Crime spy based on novel secr...
3      Action Crime Drama Thriller dc comics crime fi...
4      Action Adventure Science Fiction based on nove...
...
4798   Action Crime Thriller united states\u2013mexic...
4799   Comedy Romance A newlywed couple's honeymoon ...
4800   Comedy Drama Romance TV Movie date love at fir...
4801   A New Yorker in Shanghai Daniel Henney Eliza...
4802   Documentary obsession camcorder crush dream gi...
Length: 4803, dtype: object
```

```
▶ # converting the text data to feature vectors
```

```
vectorizer = TfidfVectorizer()
```

```
▶ feature_vectors = vectorizer.fit_transform(combined_features)
```

```
▶ print(feature_vectors)
```

```
(0, 2432)    0.17272411194153
(0, 7755)    0.1128035714854756
(0, 13024)   0.1942362060108871
(0, 10229)   0.16058685400095302
(0, 8756)    0.22709015857011816
(0, 14608)   0.15150672398763912
(0, 16668)   0.19843263965100372
(0, 14064)   0.20596090415084142
(0, 13319)   0.2177470539412484
(0, 17290)   0.20197912553916567
(0, 17007)   0.23643326319898797
(0, 13349)   0.15021264094167086
(0, 11503)   0.27211310056983656
(0, 11192)   0.09049319826481456
(0, 16998)   0.1282126322850579
(0, 15261)   0.07095833561276566
(0, 4945)    0.24025852494110758
(0, 14271)   0.21392179219912877
(0, 3225)    0.24960162956997736
(0, 16587)   0.12549432354918996
(0, 14378)   0.33962752210959823
(0, 5836)    0.1646750903586285
(0, 3065)    0.22208377802661425
(0, 3678)    0.21392179219912877
(0, 5437)    0.1036413987316636
:           :
(4801, 17266) 0.2886098184932947
(4801, 4835)  0.24713765026963996
(4801, 403)   0.17727585190343226
(4801, 6935)  0.2886098184932947
(4801, 11663) 0.21557500762727902
(4801, 1672)  0.1564793427630879
(4801, 10929) 0.13504166990041588
```



```

(0, 5437) 0.1036413987316636
:
(4801, 17266) 0.2886098184932947
(4801, 4835) 0.24713765026963996
(4801, 403) 0.17727585190343226
(4801, 6935) 0.2886098184932947
(4801, 11663) 0.21557500762727902
(4801, 1672) 0.1564793427630879
(4801, 10929) 0.13504166990041588
(4801, 7474) 0.11307961713172225
(4801, 3796) 0.3342808988877418
(4802, 6996) 0.5700048226105303
(4802, 5367) 0.22969114490410403
(4802, 3654) 0.262512960498006
(4802, 2425) 0.24002350969074696
(4802, 4608) 0.24002350969074696
(4802, 6417) 0.21753405888348784
(4802, 4371) 0.1538239182675544
(4802, 12989) 0.1696476532191718
(4802, 1316) 0.1960747079005741
(4802, 4528) 0.19504460807622875
(4802, 3436) 0.21753405888348784
(4802, 6155) 0.18056463596934083
(4802, 4980) 0.16078053641367315
(4802, 2129) 0.3099656128577656
(4802, 4518) 0.16784466610624255
(4802, 11161) 0.17867407682173203

```

## Cosine Similarity

```

# getting the similarity scores using cosine similarity

similarity = cosine_similarity(feature_vectors)

```

```

print(similarity)

[[1.          0.07219487 0.037733   ... 0.          0.          0.          ]
 [0.07219487 1.          0.03281499 ... 0.03575545 0.          0.          ]
 [0.037733   0.03281499 1.          ... 0.          0.05389661 0.          ]
 ...
 [0.          0.03575545 0.          ... 1.          0.          0.02651502]
 [0.          0.          0.05389661 ... 0.          1.          0.          ]
 [0.          0.          0.          ... 0.02651502 0.          1.          ]]

```

```
▶ print(similarity.shape)
```

```
(4803, 4803)
```

## Getting the movie name from the user

```
▶ # getting the movie name from the user
```

```
movie_name = input(' Enter your favourite movie name : ')
```

```
Enter your favourite movie name : iron man
```

```
▶ # creating a list with all the movie names given in the dataset
```

```
list_of_all_titles = movies_data['title'].tolist()  
print(list_of_all_titles)
```

```
['Avatar', 'Pirates of the Caribbean: At World's End', 'Spectre', 'The Dark Knight Rises', 'John Carter',
```

```
, 'Blackhat', 'Sky Captain and the World of Tomorrow', 'Basic Instinct 2', 'Escape Plan', 'The Legend of Hercules', 'The Sum of All Fears',
```

```
▶ # finding the close match for the movie name given by the user
```

```
find_close_match = difflib.get_close_matches(movie_name, list_of_all_titles)  
print(find_close_match)
```

```
['Iron Man', 'Iron Man 3', 'Iron Man 2']
```

```
▶ close_match = find_close_match[0]  
print(close_match)
```

```
Iron Man
```

```
▶ # finding the index of the movie with title
```

```
index_of_the_movie = movies_data[movies_data.title == close_match]['index'].values[0]  
print(index_of_the_movie)
```

```

▶ # getting a list of similar movies

similarity_score = list(enumerate(similarity[index_of_the_movie]))
print(similarity_score)

[(0, 0.033570748780675445), (1, 0.0546448279236134), (2, 0.013735500604224323), (3, 0.006468756104392058), (4, 0.03268943310073386),
40364272462), (3952, 0.012151496446083307), (3250, 0.012147477967543958), (3846, 0.012128241991713645), (1943, 0.012127214318469933),

```

```

▶ # print the name of similar movies based on the index

print('Movies suggested for you : \n')

i = 1

for movie in sorted_similar_movies:
    index = movie[0]
    title_from_index = movies_data[movies_data.index==index]['title'].values[0]
    if (i<30):
        print(i, '.',title_from_index)
        i+=1

```

## Movie Recommendation System

```

▶ movie_name = input(' Enter your favourite movie name : ')

list_of_all_titles = movies_data['title'].tolist()

find_close_match = difflib.get_close_matches(movie_name, list_of_all_titles)

close_match = find_close_match[0]

index_of_the_movie = movies_data[movies_data.title == close_match]['index'].values[0]

similarity_score = list(enumerate(similarity[index_of_the_movie]))

sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse = True)

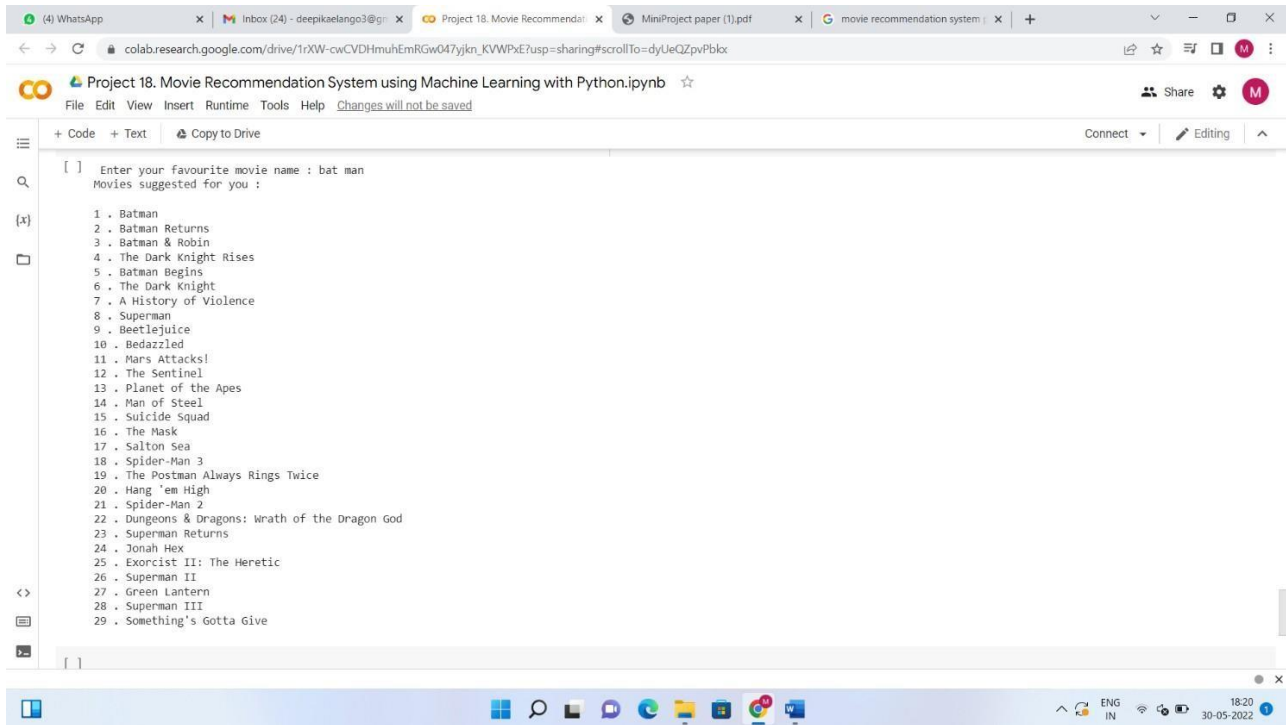
print('Movies suggested for you : \n')

i = 1

for movie in sorted_similar_movies:
    index = movie[0]
    title_from_index = movies_data[movies_data.index==index]['title'].values[0]
    if (i<30):
        print(i, '.',title_from_index)
        i+=1

```

## 5.4 SAMPLE OUTPUT



The screenshot shows a Google Colab notebook titled "Project 18. Movie Recommendation System using Machine Learning with Python.ipynb". The notebook is open to a code cell containing the following text:

```
[ ] Enter your favourite movie name : bat man
Movies suggested for you :
```

Below the code cell, a list of 29 movie recommendations is displayed, numbered 1 through 29:

- 1 . Batman
- 2 . Batman Returns
- 3 . Batman & Robin
- 4 . The Dark Knight Rises
- 5 . Batman Begins
- 6 . The Dark Knight
- 7 . A History of Violence
- 8 . Superman
- 9 . Beetlejuice
- 10 . Bedazzled
- 11 . Mars Attacks!
- 12 . The Sentinel
- 13 . Planet of the Apes
- 14 . Man of Steel
- 15 . Suicide Squad
- 16 . The Mask
- 17 . Salton Sea
- 18 . Spider-Man 3
- 19 . The Postman Always Rings Twice
- 20 . Hang 'em High
- 21 . Spider-Man 2
- 22 . Dungeons & Dragons: Wrath of the Dragon God
- 23 . Superman Returns
- 24 . Jonah Hex
- 25 . Exorcist II: The Heretic
- 26 . Superman II
- 27 . Green Lantern
- 28 . Superman III
- 29 . Something's Gotta Give

The notebook interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help), a toolbar with options like "+ Code", "+ Text", and "Copy to Drive", and a status bar at the bottom showing the system clock (18:20, 30-05-2022) and language settings (ENG, IN).

**Fig 5.2: Sample Output**

# **CHAPTER – 6**

## **TESTING AND MAINTENANCE**

## 6.1 UNIT TESTING

Unit Testing is a software testing technique by means of which individual units of software i.e., group of computer program modules, usage procedures and operating procedures are tested to determine whether they are suitable for use or not. It is a testing method using which every independent module is tested to determine if there are any issues by the developer himself. It is correlated with functional correctness of the independent modules. Unit Testing is defined as a type of software testing where individual components of a software are tested. Unit Testing of software products is carried out during the development of an application. An individual component may be either an individual function or a procedure. Unit Testing is typically performed by the developer. In SDLC or V Model, Unit testing is the first level of testing done before integration testing. Unit testing is such a type of testing technique that is usually performed by the developers. Although due to reluctance of developers to test, quality assurance engineers also do unit testing.

Objective of Unit Testing:

The objective of Unit Testing is:

- To isolate a section of code.
- To verify the correctness of code.
- To test every function and procedure.
- To fix bugs early in the development cycle and to save costs.
- To help the developers to understand the code base and enable them to make changes quickly.
- To help for code reuse.

### 6.1.1 ADVANTAGES OF UNIT TESTING

- Unit Testing allows developers to learn what functionality is provided by a unit and how to use it to gain a basic understanding of the unit API.
- Unit testing allows the programmer to refine code and make sure the module works properly.
- Unit testing enables testing parts of the project without waiting for others to be completed.

**System Testing** is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements.

In system testing, integration testing passed components are taken as input. The goal of integration testing is to detect any irregularity between the units that are integrated together. System testing detects defects within both the integrated units and the whole system. The result of system testing is the observed behavior of a component or a system when it is tested.

System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. System testing tests the design and behavior of the system and also the expectations of the customer. It is performed to test the system beyond the bounds mentioned in the software requirements specification (SRS).

## 6.2 BLACK BOX TESTING

Black box testing is a type of software testing in which the functionality of the software is not known. The testing is done without the internal knowledge of the products.

Black box testing can be done in following ways:

**1.Syntax Driven Testing** – This type of testing is applied to systems that can be syntactically represented by some language. For example- compilers, language that can be represented by context free grammar. In this, the test cases are generated so that each grammar rule is used at least once.

**2.Equivalence partitioning** – It is often seen that many types of inputs work similarly so instead of giving all of them separately we can group them together and test only one input of each group. The idea is to partition the input domain of the system into a number of equivalence classes such that each member of class works in a similar way, i.e., if a test case in one class results in some error, other members of class would also result into same error.

The technique involves two steps:

- **Identification of equivalence class** – Partition any input domain into minimum two sets: valid values and invalid values. For example, if the valid range is 0 to 100 then select one valid input like 49 and one invalid like 104.
- **Generating test cases**

(i) To each valid and invalid class of input assign unique identification number.

(ii) Write test case covering all valid and invalid test case considering that no two invalid inputs mask each other.

To calculate the square root of a number, the equivalence classes will be:

(a) Valid Inputs

(b) Invalid Inputs



**(a) Valid inputs:**

- Whole number which is a perfect square- output will be an integer.
- Whole number which is not a perfect square- output will be decimal number.
- Positive decimals

**(b) Invalid inputs:**

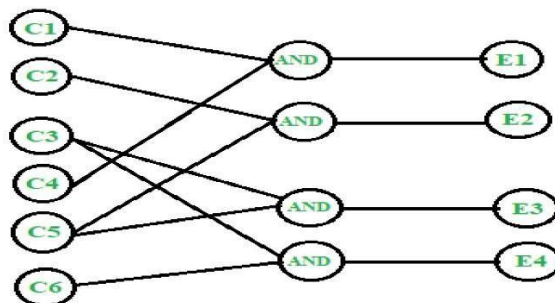
- Negative numbers (integer or decimal).
- Characters other than numbers like “a”, “!”, “;”, etc.

**3. Boundary value analysis** – Boundaries are very good places for errors to occur. Hence if test cases are designed for boundary values of input domain, then the efficiency of testing improves and probability of finding errors also increase. For example – If valid range is 10 to 100 then test for 10, 100 also apart from valid and invalid inputs.

**4. Cause effect Graphing** – This technique establishes relationship between logical input called causes with corresponding actions called effect. The causes and effects are represented using Boolean graphs. The following steps are followed:

- Identify inputs (causes) and outputs (effect).
- Develop cause effect graph.
- Transform the graph into decision table.
- Convert decision table rules to test cases.

For example, in the following cause effect graph:



**Fig 6.1: Black Box Testing**

**5. Requirement based testing** – It includes validating the requirements given in SRS of software system.

**6. Compatibility testing** – The test case result not only depend on product but also infrastructure for delivering functionality. When the infrastructure parameters are changed, it is still expected to work properly. Some parameters that generally affect compatibility of software are:

- Processor (Pentium 3, Pentium 4) and number of processors.
- Architecture and characteristic of machine (32 bit or 64 bit).
- Back-end components such as database servers.
- Operating System (Windows, Linux, etc).

### 6.3 WHITE BOX TESTING

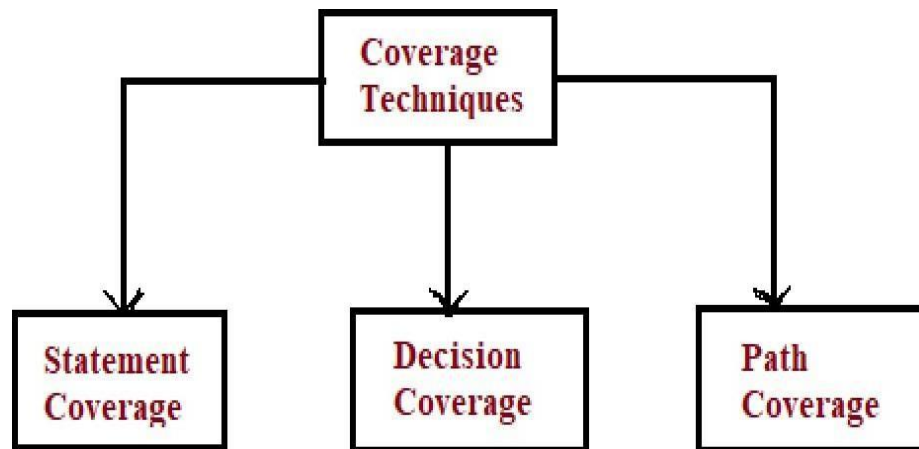
White box testing is also referred to as clear, glass box or structural testing. It is a testing method that tests the internal structure of an application. As opposed to black-box testing, it does not focus on the functionality but involves line to line assessment of the code.



**Fig 6.2: White Box Testing.**

In white box testing, the tester has to go through the code line by line to ensure that internal operations are executed as per the specification and all internal modules are properly implemented.

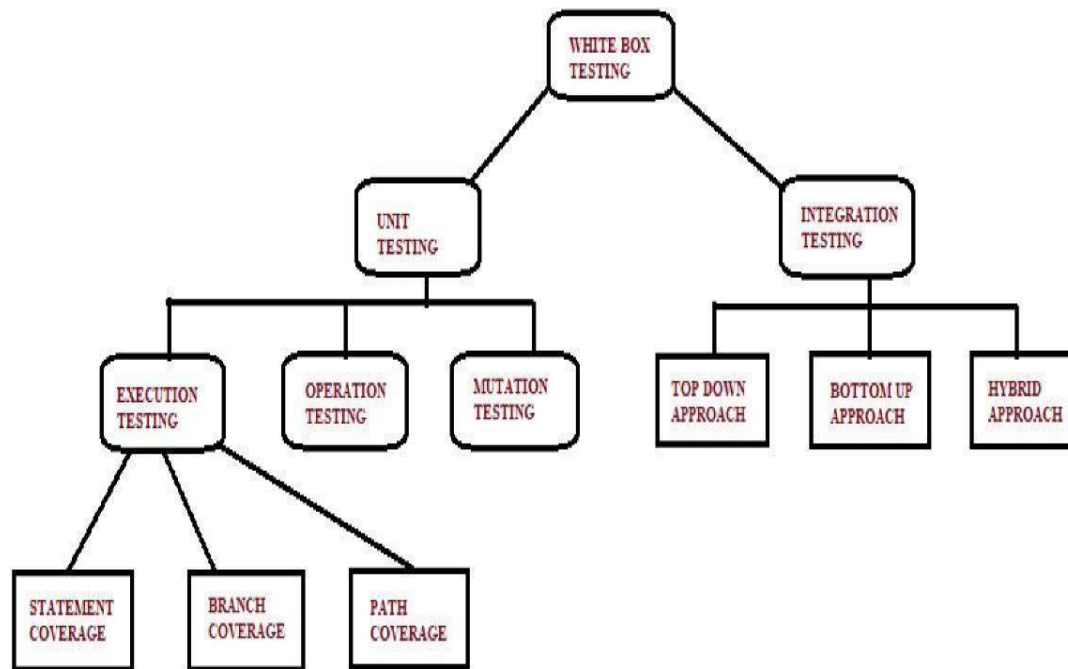
## White Box Testing Coverage



**Fig 6.3: White Box Testing Coverage.**

- **Code coverage:** It aims at testing the complete code.
- **Segment coverage:** Segment coverage confirms that every code statement is executed once while performing the testing process.
- **Branch Coverage or Node Testing:** Branch Coverage or Node Testing confirms that every code branch is executed once while testing.
- **Compound Condition Coverage:** compound condition coverage is a type of white box coverage with multiple test conditions each having multiple paths and combination to fulfil the condition.
- **Basis Path Testing:** every independent path is considered under basic path testing.
- **Data Flow Testing:** data flow testing deals with the data variable and tracks them to verify its use. They unveil the bugs relating variable initialize, declaration but not used, and so on.
- **Path Testing:** Path testing covers all the probable paths in the code.
- **Loop Testing:** Loop testing ensures the coverage of single loops, concatenated loops, and nested loops in the code.

## Types of White Box Testing



**Fig 6.4: Types of White Box Testing**

White box testing includes various types of testing, some of them are:

- **Unit Testing:** Unit testing is testing of a single unit to find defects. It is usually done by the developers to detect bugs in the code developed. It is usually the first round of testing.
- **Testing for Memory Leaks:** Memory leaks can take away all the credibility of the code, making it work slower. An experienced tester tests the code for the memory leaks detecting the actual leak point.
- **Penetration Testing:** with a detailed understanding of the code, and thorough network info, IP addresses, and all server info. The main aim of this testing is to expose the code to find its vulnerability to security threats.

## Techniques of White Box Testing

- Statement Coverage
- Branch Coverage
- Path Coverage

These techniques do not find bugs in the code but detect the path, statements, and branches that are not covered during testing.

**Statement Coverage:** It ensures whether each and every line of the code is executed at least once during testing.

**Branch Coverage:** Branch Coverage ensures that every branch from each decision point is executed.

**Path Coverage:** Path coverage ensures that every path is traversed at least once. This technique is used for testing complex programs.

### **White Box Testing Example**

Let us consider the following code snippet:

```
INPUT A &B  
  
C = A+B  
  
IF C>100  
  
PRINT "ITS DONE"
```

Now in the first, line, we assign the value of A and B. Let us suppose  $A = 60$  and  $B = 50$ . Moving on to the second line, now C is assigned a value of  $A+B$ , here  $A = 60$  and  $B = 50$ , hence  $C = 110$ . Moving on to the third line, we will check if  $C > 100$ , here the condition is true and hence we should get our result as ITS DONE

### **Advantages of White Box Testing**

- Code optimization
- Transparency of the internal coding structure
- Thorough testing by covering all possible paths of a code
- Introspection of the code by the programmers
- Easy test case automation

## **Disadvantages of White Box Testing**

- A complex and expensive procedure
- Frequent updating of the test script is required whenever change happens in the code
- Exhaustive testing for large-sized application
- Not always possible to test all the conditions.
- Need to create a full range of inputs making it a very time-consuming process

**CHAPTER – 7**  
**CONCLUSION AND FUTURE ENHANCEMENT**

## **7.1 CONCLUSION**

In this project we have designed a technique of content based filtering on a data base of movies. It collects movies from the user known as Data collection and then pre-processes it. The user enters in the search bar the movie name and gets similar movies depending on users interest. The models efficiency increases with a better data set and prediction. The results suggested by our movie recommendation system are leading and blockbuster movies and the system is helpful for many users. Our system only needs a movie which the user is interested in to come up with suitable recommendations. This project analysis similarity measure for recommendations forecasting in recommendations system. By using Recommendation System we predicted top 25 movies based on the requirements of the user and the output is shown.

## **7.2 FUTURE ENHANCEMENTS**

The project has a very vast scope in future. A possible future work includes the proposed method having better and enhanced data collection and preprocessing methods. More robust feature extraction methods shall be needed. The future plan is also to validate the model with real time datasets which are newly developed and finely tuned after applying parameters. Any other better classifier or model could also be developed to increase the performance of the movie recommendation system.



## **REFERENCES**

## REFERENCE

- [1] R. Sandeep, S. Sood, and V. Verma, “Twitter sentiment analysis of real-time customer experience feedback for predicting growth of Indian telecom companies,” in Proceedings of the 2018 4th International Conference on Computing Sciences (ICCS), pp. 166–174, IEEE, Phagwara, India, August 2018.
- [2] Nilashi, M., Ibrahim, O., Bagherifard, K.: ‘A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques’, *Expert Syst. Appl.*, 2018, 92, pp. 507–520
- [3] Tarun Bhatia: Recommendation System: Technology Review, ResearchGate 2018.
- [4] Urszula Kuzeleswska: Clustering Algorithms in Hybrid Recommender System on MovieLens Data, SILGR 2014.
- [5] Patrikainen, A., Manilla, H.: Subspace clustering of high-dimensional binary data - a probabilistic approach. In: Workshop on Clustering High Dimensional Data and its Applications, SIAM International Conference on Data Mining (2004).
- [6] Zan Wang, Xue Yu, Nan Feng, Zhenzua Wang, An improved collaborative movie recommendation system using computational intelligence, Elsevier 2014
- [7] T. Chen and Y. H. Chuang, “Fuzzy and nonlinear programming approach for optimizing the performance of ubiquitous hotel recommendation,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 2, pp. 275–284, 2018.
- [8] Y. H. Hu, P. J. Lee, K. Chen, J. M. Tarn, and D.-V. Dang, “Hotel recommendation system based on review and context information: a collaborative filtering appro,” in Proceedings of the Pacific Asia Conference on Information Systems PACIS, p. 221, Chiayi City, Taiwan, June-July 2016.
- [9] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, *Knowledge-Based Systems*, 46 (2013) 109-132.
- [10] P. Resnick, H.R. Varian, Recommender systems, *Communications of the ACM*, 40 (1997) 56-58.

- [11] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state--the-art and possible extensions, *IEEE Transactions on Knowledge and Data Engineering*, 17 (2005) 734-749.
- [12] D. Goldberg, D. Nichols, B.M. Oki, D. Terry, using collaborative filtering to weave an information tapestry, *Communications of the ACM*, 35 (1992) 61-70.
- [13] J.B. Schafer, D. Frankowski, J. Herlocker, S. Sen, Collaborative filtering recommender systems, in: P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.) *The Adaptive Web*, Springer Berlin Heidelberg 2007, pp. 291-324.
- [14] M. Pazzani, A framework for collaborative, content-based and demographic filtering, *Artificial Intelligence Review*, 13 (1999) 393-408.
- [15] A. Bellogin, I. Cantador, F. Diez, P. Castells, E. Chavarriaga, An empirical comparison of social, collaborative filtering, and hybrid recommenders, *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4 (2013) 1-29.
- [16] J. O'Donovan, B. Smyth, Trust in recommender systems, *Proceedings of the 10th International Conference on Intelligent User Interfaces*, ACM, San Diego, California, USA, 2005, pp. 167-174.
- [17] A.K. Dey, G.D. Abowd, D. Salber, A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications, *Human-Computer Interaction*, 16 (2001) 97-166.
- [18] W. Woerndl, M. Brocco, R. Eigner, Context-aware recommender systems in mobile scenarios, *International Journal of Information Technology and Web Engineering (IJITWE)*, 4 (2009) 67-85.
- [19] S. Stabb, H. Werther, F. Ricci, A. Zipf, U. Gretzel, D.R. Fesenmaier, C. Paris, C. Knoblock, Intelligent systems for tourism, *IEEE Intelligent Systems*, 17 (2002) 53-66.
- [20] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, E. Duval, Context-aware recommender systems for learning: a survey and future challenges, *IEEE Transactions on Learning Technologies*, 5 (2012) 318-335.

## **APPENDICES**

## Movie Recommendation System

<sup>1</sup>Uma S, <sup>2</sup>Deepika E, <sup>3</sup>Mohana Priya B, <sup>4</sup>Monisha D

*\*Department Of Information Technology, Panimalar Engineering College*

<sup>1</sup>[umaokj@gmail.com](mailto:umaokj@gmail.com), <sup>2</sup>[deepikaelango03@gmail.com](mailto:deepikaelango03@gmail.com),  
<sup>3</sup>[rekhamohana2001@gmail.com](mailto:rekhamohana2001@gmail.com), <sup>4</sup>[monisri31@gmail.com](mailto:monisri31@gmail.com)

**ABSTRACT** Now-a-days people are consuming content in form of movies, series, etc. for entertainment. In this modern era, people always look up to entertainment and in that process, they waste their time in searching for movies. Everyone wants to watch good films that have great content. It takes lot of time to search for a movie they like. Recommendation system comes into play in such situations. It helps to people by recommending movies. This paper develops a Movie Recommendation System to recommend movies based on different parameters. The principal objective of the project is to construct a movie recommendation framework to prescribe pictures to users. There are many algorithms that help to build a recommendation system. Here, the Content-based algorithm has been employed to recommend movies based on the similarity with other films by analysing the content of the movie. To find the similarity, the cosine similarity method has been used. Here, the cosine similarity has been computed by using linear kernel, where the parameters are taken by the result of TF-IDF vectorization. Then the most similar movies are recommended

**KEYWORD** Recommendation, Movies, Cosine similarity, Films

### 1.INTRODUCTION

Recommender systems are more popular and increase the production costs for many service providers. Today the world is an over-crowded so that the recommendations are required for recommending products or services. However, recommender systems minimize the transaction costs and improves the quality and decision-making process to users It is applied in various neighbouring areas like information retrieval or human computer interaction (HCI). Movie recommendation system design a big problem since other recommendation systems require fast computation and processing service from service providers and product distributors. To recommend movies, first collects the ratings for users and then recommend the top list of items to the target user. In addition to this, users can check reviews of other users before watching movie. A different recommendation schemes have been presented includes collaborative filtering, content-based recommender system, and hybrid recommender system. However, several issues are raised with users posted reviews. There are 3 types of recommendation systems

1. Popularity based recommendation engine
2. Content based recommendation engine
3. Collaborative filtering based recommendation engine

Content-Based methods (or cognitive filtering) on the other hand, use information and metadata about the content to find similarities among them, without incorporating user behaviour in any way. Items similar to those 'accessed 'or 'searched 'by the user are recommended here. Some approaches

analyze the audio and visual features (video frames, audio clips, movie posters etc.), as in using image and signal processing techniques while some analyze textual features (metadata like plots, subtitles, genre, cast etc.) via Natural Language Processing methods like tf-idf, as in [1], and word2vec, as in [2]. Content based recommendation engine takes in a movie that a user currently likes as input. Then it analyzes the contents of the movie to find out other movies which have similar content. Then it ranks similar movies according to their similarity scores and recommends the most relevant movies to the user.

## 2.LITERATURE SURVEY

1. Sang-Min Choi, et. al.--mentioned about the shortcomings of collaborative filtering approach like sparsity problem or the cold-start problem. In order to avoid this issue, the authors have proposed a solution to use category information. The authors have proposed a movie recommendation system which is based on genre correlations. The authors stated that the category information is present for the newly created content. Thus, even if the new content does not have enough ratings or enough views, still it can pop up in the recommendations list with the help of category or genre information. The proposed solution is unbiased over the highly rated most watched content and new content which is not watched a lot. Hence, even a new movie can be recommended by the recommendation system.
2. Muyeed Ahmed, et. al. --proposed a solution using K-means clustering algorithm. Authors have separated similar users by using clusters. Later, the authors have created a neural network for each cluster for recommendation purpose. The proposed system consists of steps like Data Pre-processing, Principal Component Analysis, Clustering, Data Pre-processing for Neural Network, and Building Neural Network. User rating, user preference, and user consumption ratio have been taken into consideration. After clustering phase, for the purpose of predicting the ratings which the user might give to the unwatched movies, the authors have used neural network. Finally, recommendations are made with the help of predicted high ratings.
3. S. Rajarajeswari, et. al. --discussed about Simple Recommender System, Content-based Recommender System, Collaborative Filtering based Recommender System and finally proposed a solution consisting of Hybrid Recommendation System. The authors have taken into consideration cosine similarity and SVD. Their system gets 30 movie recommendations using cosine similarity. Later, they filter these movies based on SVD and user ratings. The system takes into consideration only the recent movie which the user has watched because the authors have proposed a solution which takes as input only one movie.
4. Jiang Zhang, et. al. --proposed a collaborative filtering approach for movie recommendation and they named their approach as 'Weighted KM-Slope-VU'. The authors divided the users into clusters of similar users with the help of K-means clustering. Later, they selected a virtual opinion leader from each cluster which represents the all the users in that particular cluster. Now, instead of processing complete user-item rating matrix, the authors processed virtual opinion leader-item matrix which is of small size. Later, this smaller matrix is processed by the unique algorithm proposed by the authors. This way, the time taken to get recommendations is reduced.
5. Debashis Das, et. al. --wrote about the different types of recommendation systems and their general information. This was a survey paper on recommendation systems. The authors mentioned about Personalized recommendation systems as well as non-personalized systems. User based collaborative

filtering and item based collaborative filtering was explained with a very good example. The authors have also mentioned about the merits and demerits of different recommendation systems.

6. Md. Akter Hossain, et. al. --proposed NERS which is an acronym for neural engine-based recommender system. The authors have done a successful interaction between 2 datasets carefully. Moreover, the authors stated that the results of their system are better than the existing systems because they have incorporated the usage of general dataset as well as the behaviour-based dataset in their system. The authors have used 3 different estimators in order to evaluate their system against the existing systems.

7. Harper, et. al. --mentioned the details about the movie Lens Dataset in their research paper. This dataset is widely used especially for movie recommendation purpose. There are different versions of dataset available like movie Lens 100K / 1M / 10M / 20M / 25M /1B Dataset. The dataset consists of features like user id, item id / movie id, rating, timestamp, movie title, IMDb URL, release date, etc. along with the movie genre information.

8. V. Subramaniaswamy, et. al. --have proposed a solution of personalized movie recommendation which uses collaborative filtering technique. Euclidean distance metric has been used in order to find out the most similar user. The user with least value of Euclidean distance is found. Finally, movie recommendation is based on what that particular user has best rated. The authors have even claimed that the recommendations are varied as per the time so that the system performs better with the changing taste of the user with time.

9. Pavithra, M. et al --designed and implement a movie recommendation system. There are different genres, cultures and languages to choose from in the world of movies. Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. On an average of one year movie survey 600 movies are released in Hollywood. For streaming movie services like Netflix, recommendation systems are essential for helping users find new movies to enjoy. So far, a decent number of works has been done in this field. But there is always room for renovation.

10. Xi, W. et al. --proposed a novel recommendation algorithm based on Back Propagation (BP) neural network with Attention Mechanism (BPAM). In particular, the BP neural network is utilized to learn the complex relationship of the target users and their neighbors. Compared with deep neural network, the shallow neural network, i.e., BP neural network, can not only reduce the computational and storage costs, but also prevent the model from over-fitting. In addition, an attention mechanism is designed to capture the global impact on all nearest target users for each user.

## **2.a. EXISTING SYSTEM**

- Recommendation systems are software applications that suggest or recommend movies or product to users.

### **TYPES**

1. **CONTENT BASED:** It is also known as cognitive filtering. It provides recommendation by comparing representation of content describing an item or a product to representation of the content describing the interest to the user. It is suitable in situation or domains where items are more than users.

2. **POPULARITY BASED:** It is used to check movies that are in current trend or most popular among the users and it directly recommended it.
3. **COLLABORATIVE BASED:** It is a family of algorithms where there are multiple ways to calculate rating based on ratings of similar users i.e., previous data collected from other individuals.
4. We are Creating a system in python where user can give name of this favourite movies and based on this input, we are going to recommend certain movies to them.
5. In this we are going to do content based and certain kind of popularity based.

## 2.b. PROPOSED SYSTEM

With our proposed system we aim at building a system that gives more accurate results. Using Python language, we aim at creating a source code that is also compatible with our GUI. With the help Machine learning library like NumPy and Panda, we are making the system that can do mathematical dimensional array and matrices calculation on its own. In our project we use Python language as the main source code. The database which we are going to use contains Movie information and Ratings and as per that given information the system is going to give a recommendation, our system is going to start giving recommendations to the users which will be the final phase of our system.

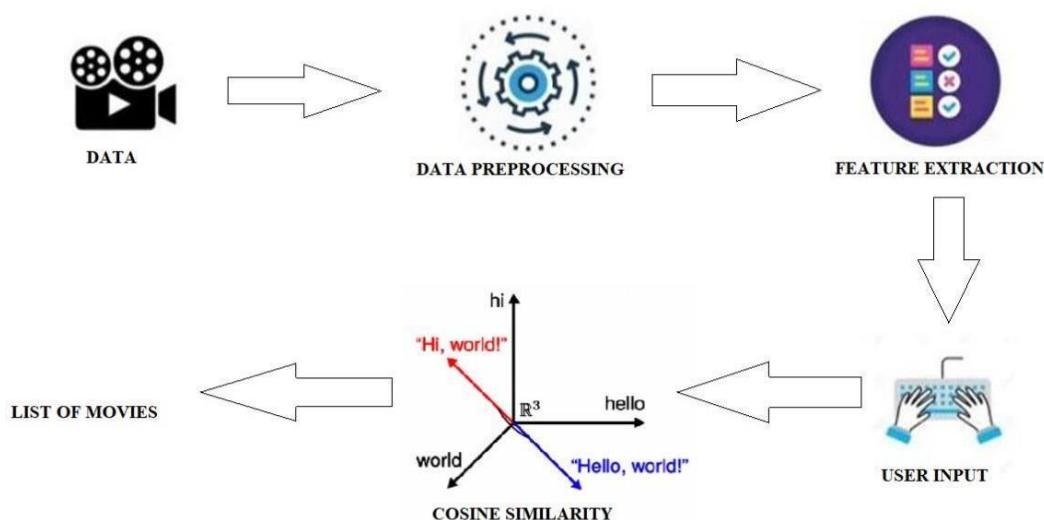


FIGURE 1: ARCHITECTURE DIAGRAM



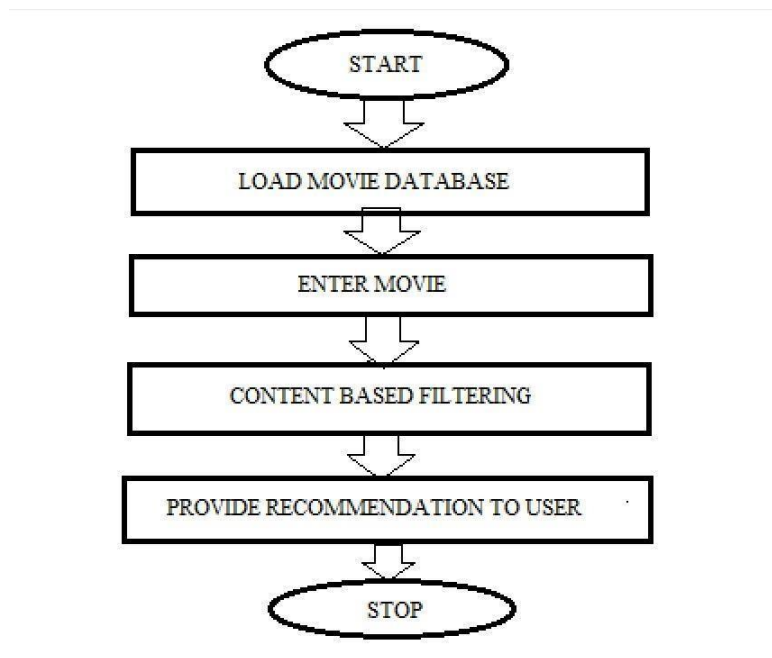


FIGURE 2: FLOW CHART

## 2.c. MODULES

In this, we plan the outline and execution of the project. There are four modules which we are going to explain:

1. Importing the dependencies.
2. Data collection and Pre-processing.
3. Cosine similarity.
4. Getting the movie name from the user.

### 1. Importing the dependencies:

The dependencies are nothing but the libraries and functions that we need. We are going to import libraries like numpy, pandas and difflib. Then we import TD-IDF and cosine-similarity.

### 2. Data collection and Pre-processing:

Data collection:

Collecting data for training the ML model is the basic step in the machine learning pipeline.

Pre-processing:

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data pre-processing task.

Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

### 3. Cosine similarity:

Cosine similarity is a metric used to measure how similar two items are. Mathematically it calculates the cosine of the angle between two vectors projected in a multidimensional space.

We need to find similarity of all movies we will do that by cosine similarity it will give some similarity score for all different movies compared to other movies we use this to recommended. Similarity score is numeric value which range from 0 to 1.

### 4. Getting the movie name from user:

We should get the movie name from user. The data filter the movies given by the user and they recommend the similar movies to the user.

## 3.RESULTS AND DISCUSSION

By using Recommended System, we predicted top 25 movies based on the requirements of the user and the output is shown in Fig. This is to conclude that recommended presented in this paper is very helpful to study the priorities of the customers and recommend other movies similar to their interest.

index	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	...	runtime	spoken_languages	status
0	0	Action Adventure Fantasy Science Fiction	<a href="http://www.avatarmovie.com/">http://www.avatarmovie.com/</a>	19995	culture clash future space war space colony so...	en	Avatar	In the 22nd century, a paraplegic Marine is dl...	150.437577	...	162.0	[{"iso_639_1": "en", "name": "English"}, {"iso...	Released
1	1	Adventure Fantasy Action	<a href="http://disney.go.com/disneypictures/pirates/">http://disney.go.com/disneypictures/pirates/</a>	285	ocean drug abuse exotic island east India trad...	en	Pirates of the Caribbean: At World's End	Captain Barbosa, long believed to be dead, ha...	139.082615	...	169.0	[{"iso_639_1": "en", "name": "English"}]	Released
2	2	Action Adventure Crime	<a href="http://www.sonypictures.com/movies/spectre/">http://www.sonypictures.com/movies/spectre/</a>	206847	spy based on novel secret agent sequel mi6	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	...	148.0	[{"iso_639_1": "fr", "name": "Fran\u00e7ais"}, {"iso...	Released
3	3	Action Crime Drama Thriller	<a href="http://www.thedarkknightrises.com/">http://www.thedarkknightrises.com/</a>	49026	dc comics crime fighter terrorist secret	en	The Dark Knight Rises	Following the death of District Attorney Harvey...	112.312950	...	165.0	[{"iso_639_1": "en", "name": "English"}]	Released

FIGURE 3 MOVIES LIST

```

Project 18. Movie Recommendation System using Machine Learning with Python.ipynb
File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text Copy to Drive

Enter your favourite movie name : bat man
Movies suggested for you :

1 . Batman
2 . Batman Returns
3 . Batman & Robin
4 . The Dark Knight Rises
5 . Batman Begins
6 . The Dark Knight
7 . A History of Violence
8 . Superman
9 . Beetlejuice
10 . Bedazzled
11 . Mars Attacks!
12 . The Sentinel
13 . Planet of the Apes
14 . Man of Steel
15 . Suicide Squad
16 . The Mask
17 . Salton Sea
18 . Spider-Man 3
19 . The Postman Always Rings Twice
20 . Hang 'em High
21 . Spider-Man 2
22 . Dungeons & Dragons: Wrath of the Dragon God
23 . Superman Returns
24 . Jonah Hex
25 . Exorcist II: The Heretic

```

FIGURE 4: SUGGESTED MOVIES

### 3. CONCLUSION

In this paper we have designed a technique of content based filtering on a data base of movies. It collects movies from the user known as Data Collection and then pre-processes it. The user enters in the search bar the movie name and gets recommended 25 movies depending on the user's interest. The model's efficiency increases with a better data set and prediction.

### REFERENCES

- [1] R. Sandeep, S. Sood, and V. Verma, "Twitter sentiment analysis of real-time customer experience feedback for predicting growth of Indian telecom companies," in Proceedings of the 2018 4th International Conference on Computing Sciences (ICCS), pp. 166–174, IEEE, Phagwara, India, August 2018.
- [2] Nilashi, M., Ibrahim, O., Bagherifard, K.: 'A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques', Expert Syst. Appl., 2018, 92, pp. 507–520
- [3] Tarun Bhatia: Recommendation System: Technology Review, ResearchGate 2018.
- [4] Urszula Kuzeleswska: Clustering Algorithms in Hybrid Recommender System on MovieLens Data, SILGR 2014.
- [5] Patrikainen, A., Manilla, H.: Subspace clustering of high-dimensional binary data - a probabilistic approach. In: Workshop on Clustering High Dimensional Data and its Applications, SIAM International Conference on Data Mining (2004).
- [6] Zan Wang, Xue Yu, Nan Feng, Zhenzua Wang, An improved collaborative movie recommendation system using computational intelligence, Elsevier 2014

- [7] T. Chen and Y. H. Chuang, “Fuzzy and nonlinear programming approach for optimizing the performance of ubiquitous hotel recommendation,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 2, pp. 275–284, 2018.
- [8] Y. H. Hu, P. J. Lee, K. Chen, J. M. Tarn, and D.-V. Dang, “Hotel recommendation system based on review and context information: a collaborative filtering appro,” in *Proceedings of the Pacific Asia Conference on Information Systems PACIS*, p. 221, Chiayi City, Taiwan, June-July 2016.
- [9] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, *Knowledge-Based Systems*, 46 (2013) 109-132.
- [10] P. Resnick, H.R. Varian, Recommender systems, *Communications of the ACM*, 40 (1997) 56-58.
- [11] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, *IEEE Transactions on Knowledge and Data Engineering*, 17 (2005) 734-749.
- [12] D. Goldberg, D. Nichols, B.M. Oki, D. Terry, using collaborative filtering to weave an information tapestry, *Communications of the ACM*, 35 (1992) 61-70.
- [13] J.B. Schafer, D. Frankowski, J. Herlocker, S. Sen, Collaborative filtering recommender systems, in: P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.) *The Adaptive Web*, Springer Berlin Heidelberg 2007, pp. 291-324.
- [14] M. Pazzani, A framework for collaborative, content-based and demographic filtering, *Artificial Intelligence Review*, 13 (1999) 393-408.
- [15] A. Bellogin, I. Cantador, F. Diez, P. Castells, E. Chavarriaga, An empirical comparison of social, collaborative filtering, and hybrid recommenders, *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4 (2013) 1-29.
- [16] J. O'Donovan, B. Smyth, Trust in recommender systems, *Proceedings of the 10th International Conference on Intelligent User Interfaces*, ACM, San Diego, California, USA, 2005, pp. 167-174.
- [17] A.K. Dey, G.D. Abowd, D. Salber, A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications, *Human-Computer Interaction*, 16 (2001) 97-166.
- [18] W. Woerndl, M. Brocco, R. Eigner, Context-aware recommender systems in mobile scenarios, *International Journal of Information Technology and Web Engineering (IJITWE)*, 4 (2009) 67-85.
- [19] S. Stabb, H. Werther, F. Ricci, A. Zipf, U. Gretzel, D.R. Fesenmaier, C. Paris, C. Knoblock, Intelligent systems for tourism, *IEEE Intelligent Systems*, 17 (2002) 53-66.
- [20] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, E. Duval, Context-aware recommender systems for learning: a survey and future challenges, *IEEE Transactions on Learning Technologies*, 5 (2012) 318-335.