**Machine A Movie Recommender System: MOVREC usingLearning Technique**

**I. INTRODUCTION**

Every one of us needs entertainment to lift our spirits in this fast-paced world energy. Our confidence for work is restored by entertainment, and we work more ardently as a result .We can watch our favourite movies or listen to our favourite music to recharge choice. We can use internet movie suggestion services to view good movies which are more trustworthy, as finding favourite movies will take more and more time something one simply cannot waste[1].

This paper discuss about recommendations of the movies, a movie suggestion is significant in our social lives .In offering better entertainment. A system like this can recommend a selection of films to consumers based on their preferences or the popularity of the cinema. A recommendation system is employed to make suggestions for things to see or buy.A recommender system, or a recommendation system (sometimes replacing 'system' with a synonym such as platform or engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item[1][2]. They are primarily used in commercial applications. MOVREC also help users to find the movies of their choices based on the movie experience of other users in efficient and effective manner without wasting much time in useless browsing.[1]

Movies are of various kinds, such as those meant for amusement, those meant for teaching, children's animation movies, horror movies, and action movies. Movies' genres, such as comedy, thriller, animation, action, etc., make it simple to distinguish between them. Another approach to differentiate between movies is to look at their release year, language, director, etc.

Recommendation systems basically use two approaches [1-3]. The first one is based on content based filtering and the second one is based on collaborative filtering.[3] Current recommender systems typically combine one or more approaches into a hybrid system.

**2.METHODS**

In the field of machine learning, classification methods which use different strategies to organize and classify data. Classifiers could possibly require training data.

**a. Collaborative filtering**

**b. Content-based filtering**

**c. Hybrid recommender systems**

1. **Collaborative Filtering**

Collaborative filtering is based on the ratings given by different users. It works based on the fact that if two users give same rating to some movies, then they are having similar preferences. So it recommends movie to the users based on the rating of the movie by a similar user.[3] Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past[1]

They produce suggestions by identifying peer users or things with rating histories similar to the current user or item.

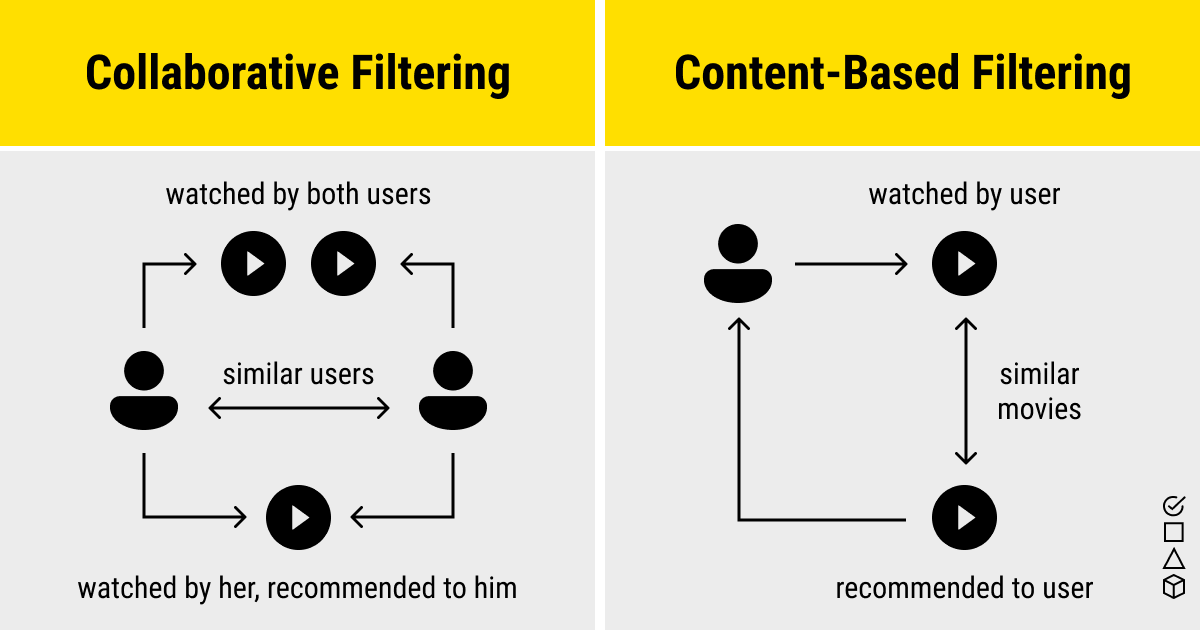
Collaborative filtering methods are classified as memory-based and model-based. A well-known example of memory-based approaches is the user-based algorithm,[2,4] while that of model-based approaches is the Kernel-Mapping Recommender.

The fact that the collaborative filtering approach does not rely on content that can be automatically analysed means that it may accurately recommend complicated objects like movies without necessitating a "understanding" of the item itself.

**b. Content-based filtering**

Content based filtering systems generally analyse the content of the information and finds the similarity among them. In a movie recommendation system based on content based filtering, to recommend a new movie to a particular user, the system will first analyse all the movies watched by that user over a period of time and analyse the content of those movies. It then recommends the movies which are having similar content to the user[3]  By the data we create a user profile, which is then used to suggest to the user, as the user provides more input or take more actions on the recommendation, the engine becomes more accurate.[5]

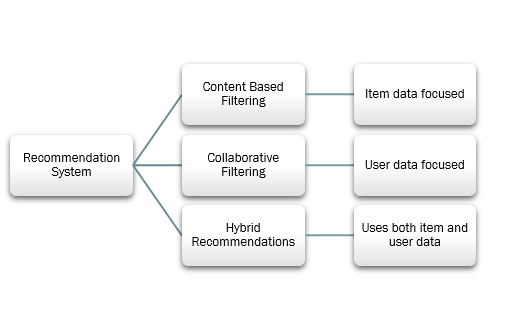
During recommendation, the similarity metrics (We will talk about it in a bit) are calculated from the item’s feature vectors and the user’s preferred feature vectors from his/her previous records. Then, the top few are recommended.[6] Content-based filtering does not require other users' data during recommendations to one user.



**c. Hybrid recommender systems**

A hybrid recommendation system is a special type of recommendation system which can be considered as the combination of the content and collaborative filtering method.

This approach overcomes the limitations of both content-based and collaborative filtering methods.

Hybrid recommender system approaches can be implemented in various ways like by using content and collaborative-based methods to generate predictions separately and then combining the prediction or we can just add the capabilities of collaborative-based methods to a content-based approach (and vice versa). [7] 

**3. RELATED WORK**

1. Kuzelewska, U. [1,3] a two-phase recommendation system with an offline phase and an online phase was proposed. Offline phase groups objects using the clustering method, people with comparable interests. For each cluster, representatives are chosen. In order to recommend a movie to a user during the algorithm's online phase, only the representatives of the clusters chosen during the offline phase are taken into account.
2. Geetha et al. [2,3] developed a hybrid strategy for the movie recommendation system employing collaborative filtering and content-based filtering. K-means clustering method was utilised to increase accuracy of the apparatus. The correlation coefficient is used to determine how similar the movies are to one another
3. De Campos et al. [3] proposed a hybrid recommendation system. Bayesian network was used to find the distribution of probability in the ratings awarded by the users. They have used the static topology information which represent the information about user profile and dynamic topology information which represent the relationship between different movies.
4. Manoj Kumar et al. [4,3] introduced a recommendation system called MOVREC that uses the K-means algorithm and a collaborative filtering approach to suggest movies to users. Using the data that other users had contributed, they suggested movies to users. They suggested a cumulative weighted approach, in which weights are awarded to films depending on five criteria: actor, director, rating, genre, and year.
5. Movie Recommender System proposed by Nupur Kalra et al. [5,3] employed collaborative filtering to generate recommendations that may aid consumers in choosing an item of interest out of thousands of options. Their paper's goal is to investigate the operation of the collaborative filtering approach utilising the Filmtrust dataset. The users were given a list of recommendations based on the results.
6. Recommendation system with collaborative filtering using big data was a survey done by Sonali et al. [6,3] They developed a strategy that blends big data techniques like association rule mining with recommender system methods like collaborative filtering.

Their effort was primarily focused on developing a scalable and reliable recommendation system.

1. MovieMender[7,3] is a movie recommendation system for users presented by Rupali Hande et al. They suggested a hybrid approach built on collaborative filtering and content-based filtering. The Pearson Correlation Coefficient was used to gauge user similarity. The hybrid algorithm's implementation specifics weren't provided.
2. S. Sridevi1 , Celeste Murnal 2 1[3] implemented a recommendation system based on content-based filtering and collaborative filtering. Cold start problem in the dataset is addressed by adding 0 ratings. The proposed system used 13 features consisting of user information, movie information and predicted top-10 movies that are similar to user interests using content based and collaborative based filtering. The performance of the system is improved by applying XGBoost algorithm.
3. A Movie Recommender System: MOVREC using Machine Learning Techniques Ashrita Kashyap1 , Sunita. B2 , Sneh Srivastava , Aishwarya. PH4 , Anup Jung Shah. In this paper , it allows a user to select his choices from a given set of attributes and then recommend him a movie list based on the cumulative weight of different attributes and using K-means algorithm.[1]
4. “MOVIE RECOMMENDATION SYSTEM” PAVAN KUMAR P B ,NITESH S, MURALIDHARA REDDY SY ,PRAVEEN V .In this research, a hybrid technique by combining content-based filtering and sentiment analysis was used to increase the precision, quality, and scalability of the movie recommendation system.The suggested methodology uses Singular Value Decomposition (SVD) as a classifier and Cosine Similarity as the filtering metric. For three separate movie datasets, existing pure methodologies and the hybrid approach are put into practise, and the outcomes are compared.[10]

11.PRODUCT RECOMMENDATION SYSTEM USING MACHINE LEARNING TECHNIQUES By MUKHUL KANAGALA .The primary goal of this project is to provide recommendations to the user in a e-commerce website by making use of machine learning algorithms. Designed and implemented the system using collaborative filtering and Pearson correlation coefficient. The dataset considered has the ratings given by the other users to a specific product and depending on the similarity between the rated product we try to recommend the products to our current user.

12.BOOK RECOMMENDATION SYSTEM USING MACHINE LEARNING Anjali Sanjivanrao More1 , Kalyani Manoj Swamy2 , Apurwa Baliram Bhoir3 , Kazi Nujhat Parveen Mohd Afjal4 The overall process of recommending books to the user of all age group category make use of collaborative filtering methodology where different users give ratings on the same book and the average number of rating is been calculated and the top-rated book is been recommended to the user. The system mainly focuses on the easy finding of best books which does not need much time or work.

## 13.Book Recommendation System Using Machine Learning, Prof. S. R. Hiray, Mr. Atish Bhosale, Ms. Komal Patil, Ms. Amruta Gaikwad, Mr. Riddhesh Deshmukh,In this research, we describe a collaborative filtering-based recommendation system. The main objective was to speed up suggestions, which is to design a system that can give customers high-quality recommendations without requiring them to register for an extended period of time and have an excellent profile experience, browsing history, etc. According to test results, the suggested strategy offers sound advice. The suggested action can be used to various fields to advertise items like movies, music, and other goods.

## 4. Algorithims(used in ML)

## There are various algorithms used in Recommender System. Some of the most common ones are:

**4.1)Support Vector Machines (SVMs):**

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. 

**4.2)Decision Trees:**

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.



**4.3)Random Forests:**

**Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.** It is based on the concept of **ensemble learning,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.



# 4.4)Logistic Regression:-

Logistic regression is one of the Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. It predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.



# 4.5)Linear Regression:-

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables. It shows a linear relationship between a dependent (y) and one or more independent (y) variables.



# 4.6)K-Nearest Neighbor(KNN):-

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. It stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as for Classification but mostly it is used for the Classification problems. Classification but mostly it is used for the Classification problems.



# 4.7)Naïve Bayes Classifier:-

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset.

# 4.8)[Vectorization](https://www.tutorialandexample.com/vectorization-in-machine-learning):-

Vectorization in machine learning refers to the process of converting data into arrays of numerical values, known as vectors. **This** is typically done to speed up the processing time of machine learning algorithms, as many of these algorithms are designed to operate on numerical data in vector form.

Vectorization is particularly important when working with large datasets, as non-vectorized algorithms can be slow and computationally expensive. By converting data into vectors, machine learning algorithms can operate on the data much more quickly and efficiently.

**5.Steps for building the project**

**1.Gathering Data:-**

Collecting all the required data set from Kaggle web site.In this project we have 5000 movies data,it has two files one is movies.css and another one is credits.css.Here in movies.css we have 20 columes of different datas, like budget,popularity,geners,id,original language,production house etc and in credits.css we have cast and crew information.

[**https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata?resource=download**](https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata?resource=download)

**2.Data preparation:-**

For building the movie recommdation system we use Jupiter notebook.Here we import our downloaded data-set and apply different techniques on it.

import numpy as np # linear algebra

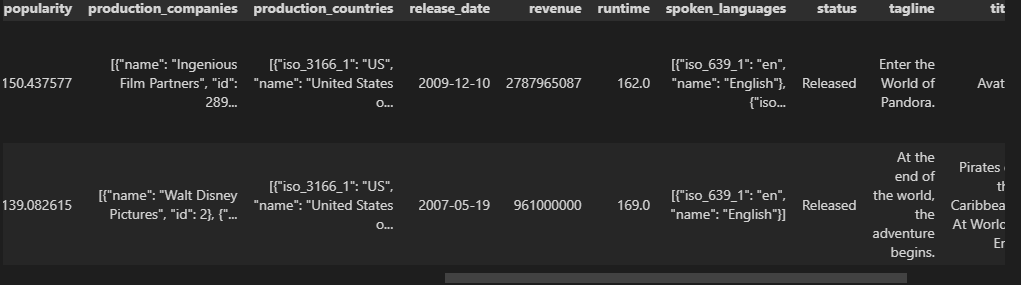
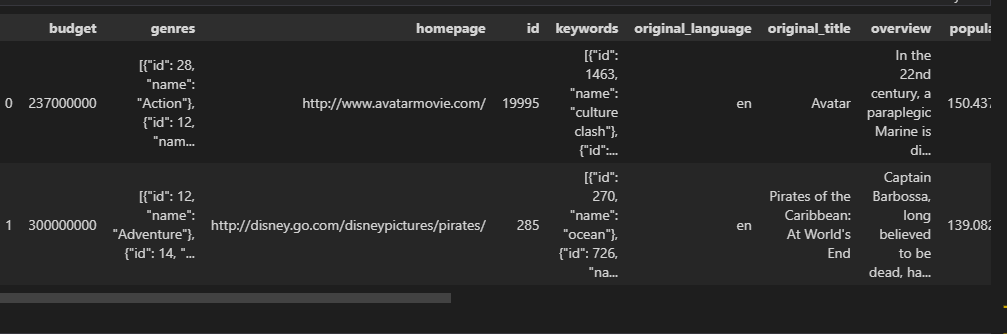
import pandas as pd # data processing, CSV file I/O (e.g. pd.read\_csv)

movies = pd.read\_csv('/kaggle/input/tmdb-movie-metadata/tmdb\_5000\_movies.csv')

credits = pd.read\_csv('/kaggle/input/tmdb-movie-metadata/tmdb\_5000\_credits.csv')

And our data is look like this

movies.head(2)

****

movies.shape

.shapes will tell about the size of the data and in this case we have

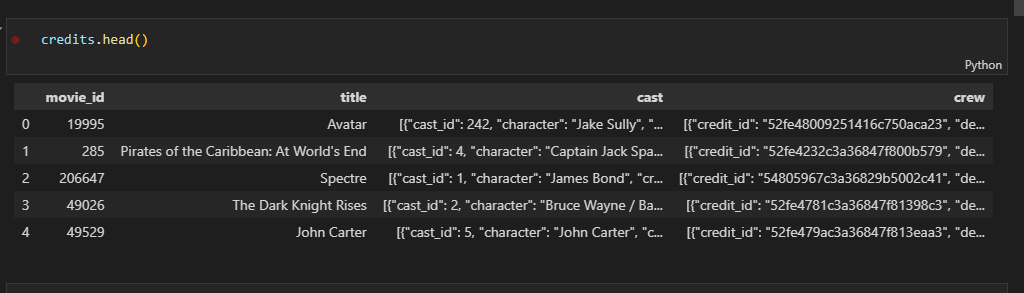
(4803,20)

It has different heads of our dataset like:-

* Budget
* Genres
* Homepage (website)
* I'd(refer ( to tmbd)
* Keywords
* Original language
* Original title
* Overview
* Popularity
* Product company
* Production country(where this movie is shoot)
* Release date
* Revenue
* Runtime
* Spoken language
* Status
* Tagline
* Tittle
* Vote average
* Vote count

And in credits we have:-

* Movie I'd
* Tittle
* Cast
* Crew

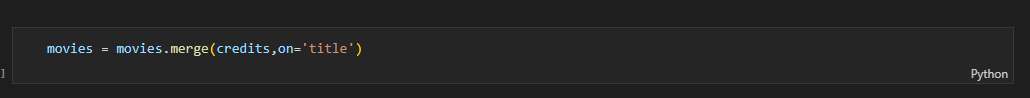


**3.Data Wrangling:-**

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step.

So we now going to clear our data and select some of the important heads for our recommendation system.

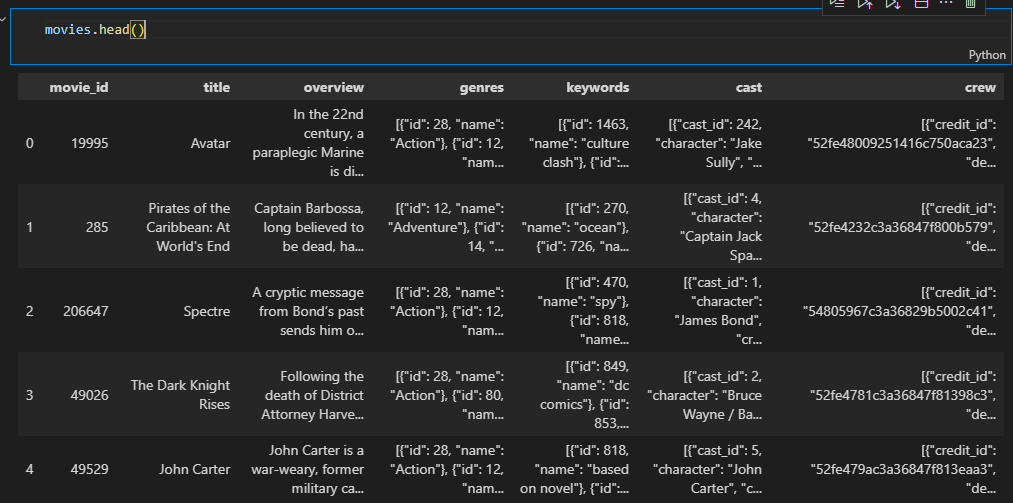
Here we have two different data set is movies.css and credits.css, now we going to join the two dataset for simplify our task . For that we take common head from both the dataset i.e “tittle”

****

And now our dataset has a size of (4809,23)

Now we going to create some tags for our recommendation system and this tags are:-

* Genres
* I'd
* Keywords
* Tittle
* Overview (content similarity)
* Cast (we select top three most artist )
* Crew(we select director here)
* movies = movies[['movie\_id','title','overview','genres','keywords','cast','crew']]



And for creating tags we going to merge overview with geners,keywords,cast and crew.

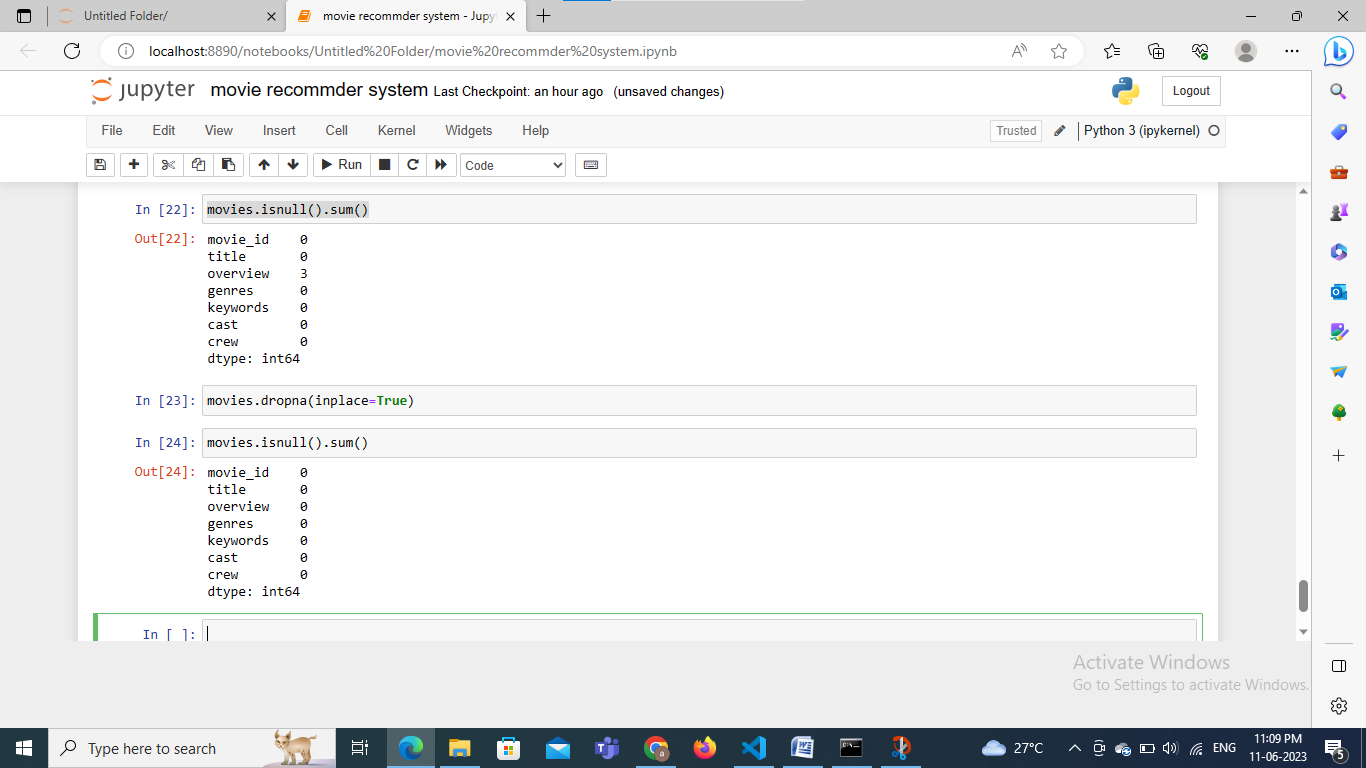
And at finally we get our dataset for recommendation system i.e

* movie\_id,
* tags,
* tittle

Also we not use numeric type heads because it will affect our approach for recommendation system .

movies.isnull().sum()

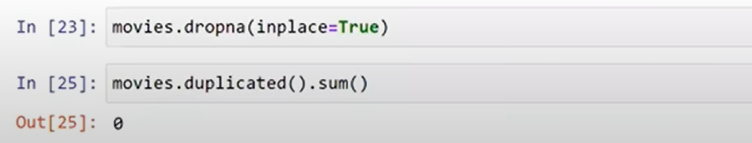
Now we removing missing data from our dataset



So now we going to drop them



And for checking duplicate data we write

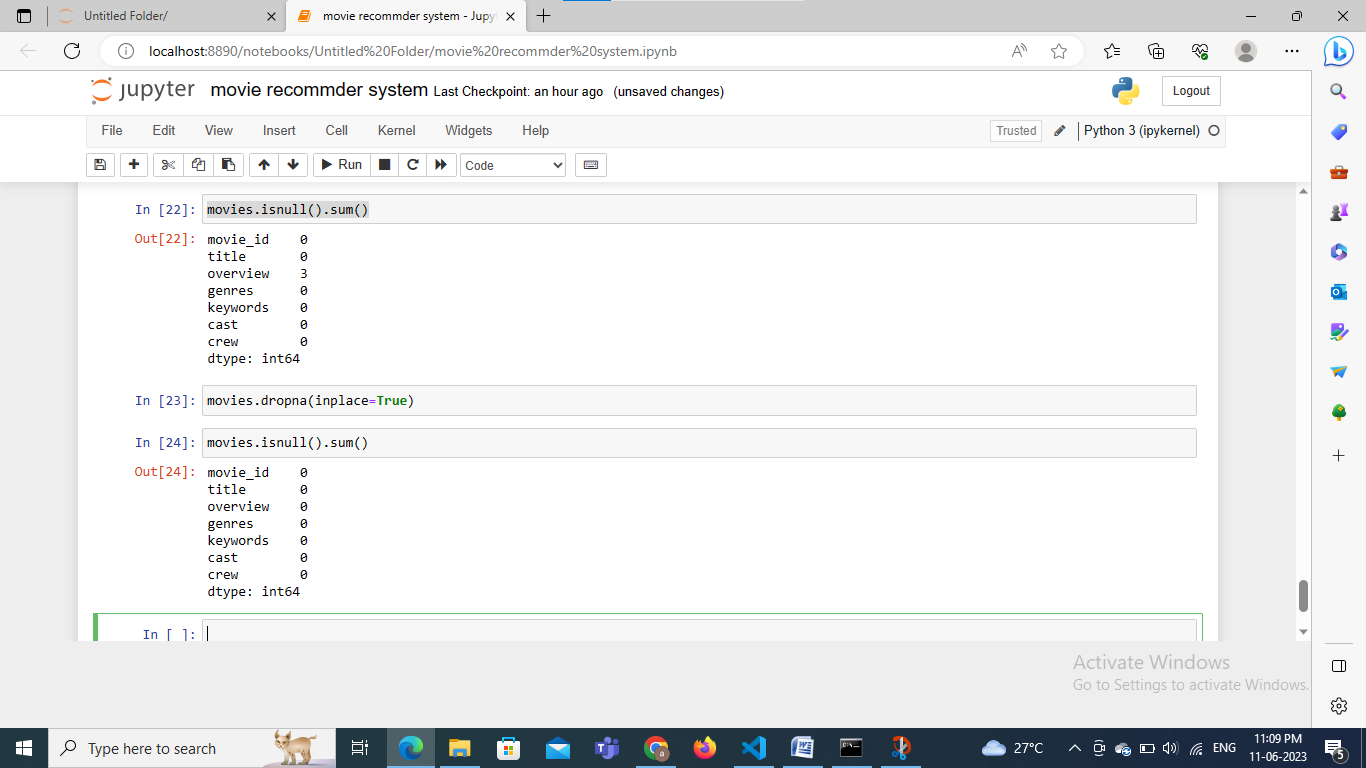


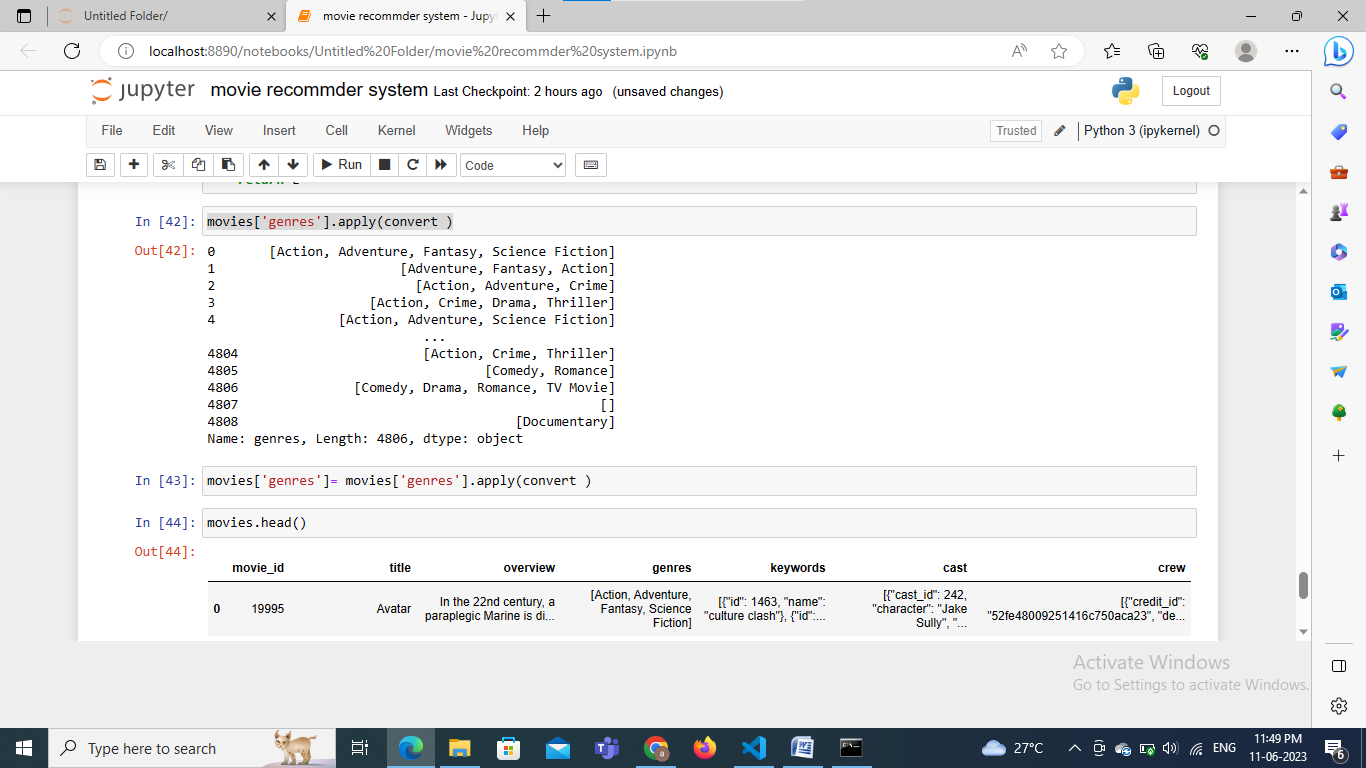
And we find that in our dataset we have no duplicate data.

**4. Analyse Data:-**

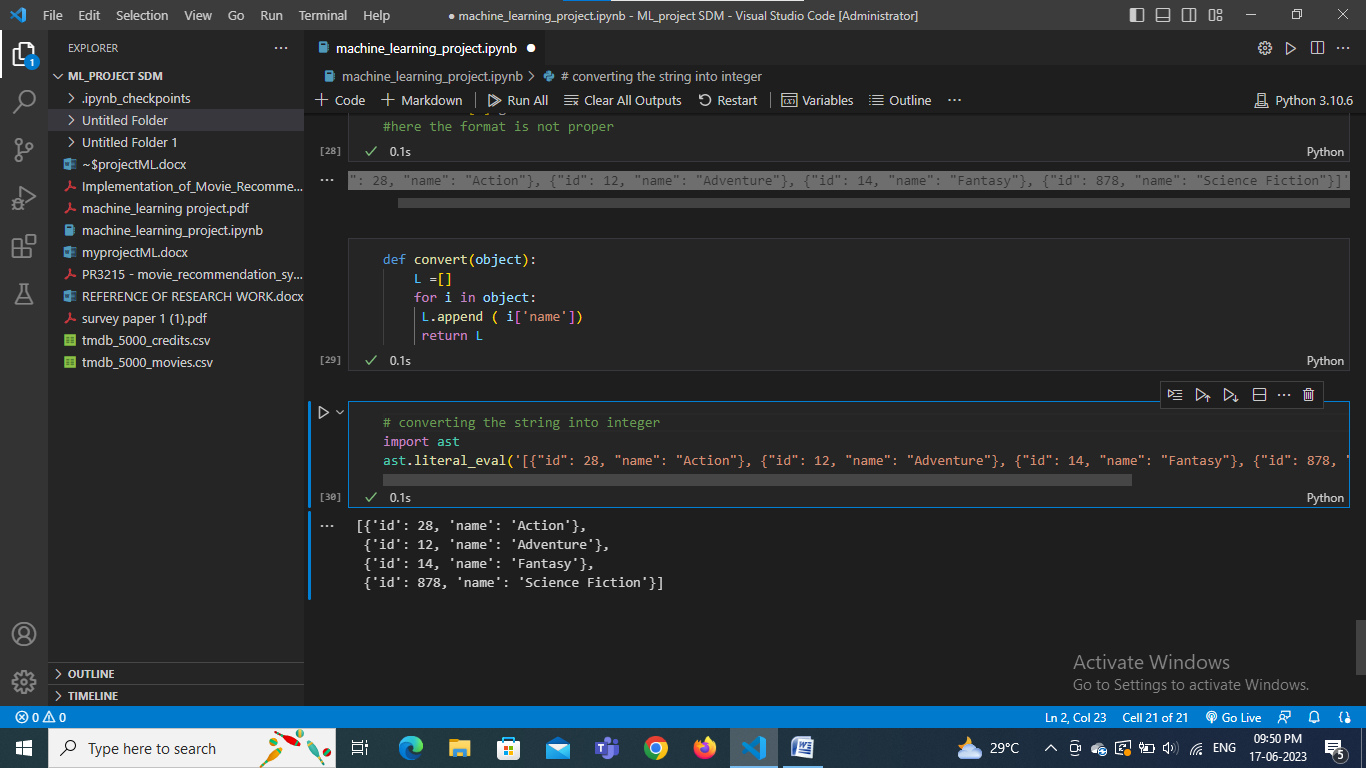
Now for  analyze the dataset we going to arrange the data in a proper manner, for that we create a helper function

And convert the data of genres and keyword in a proper manner.

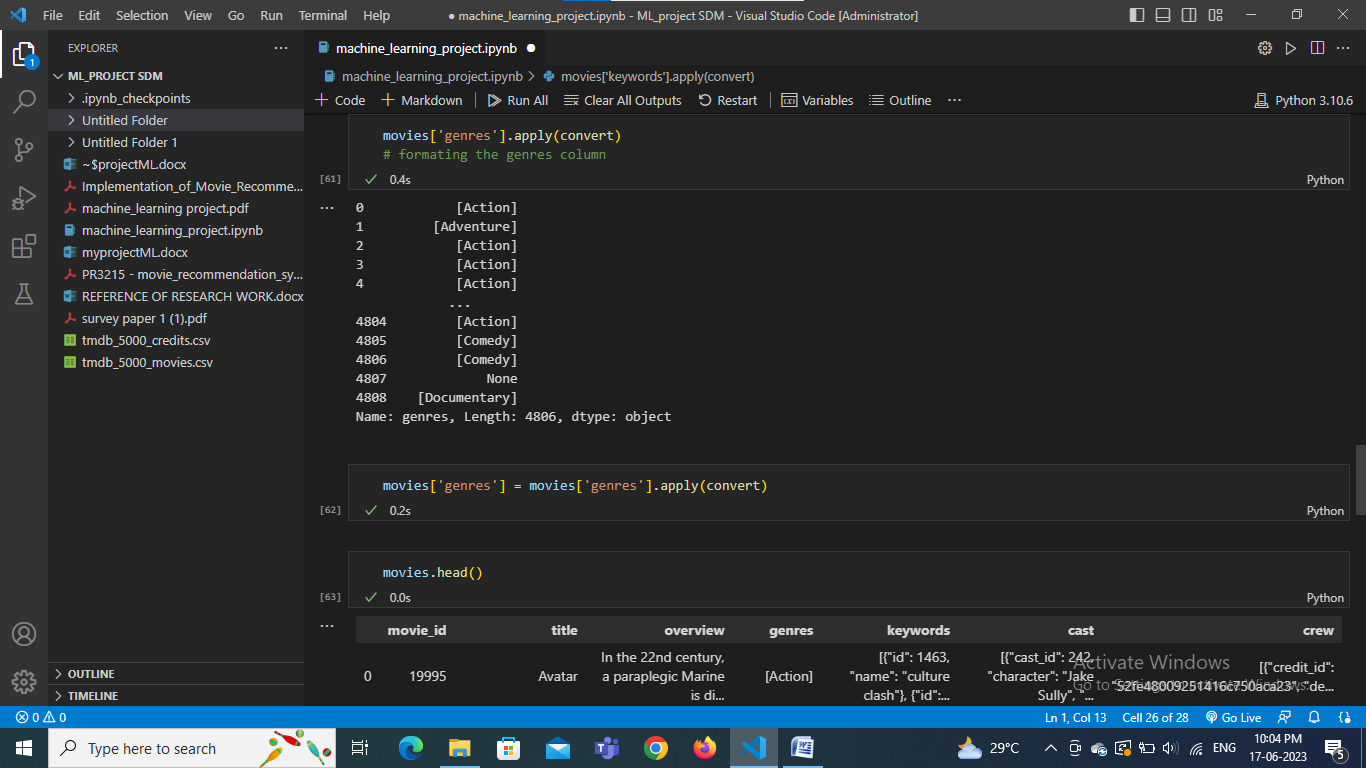




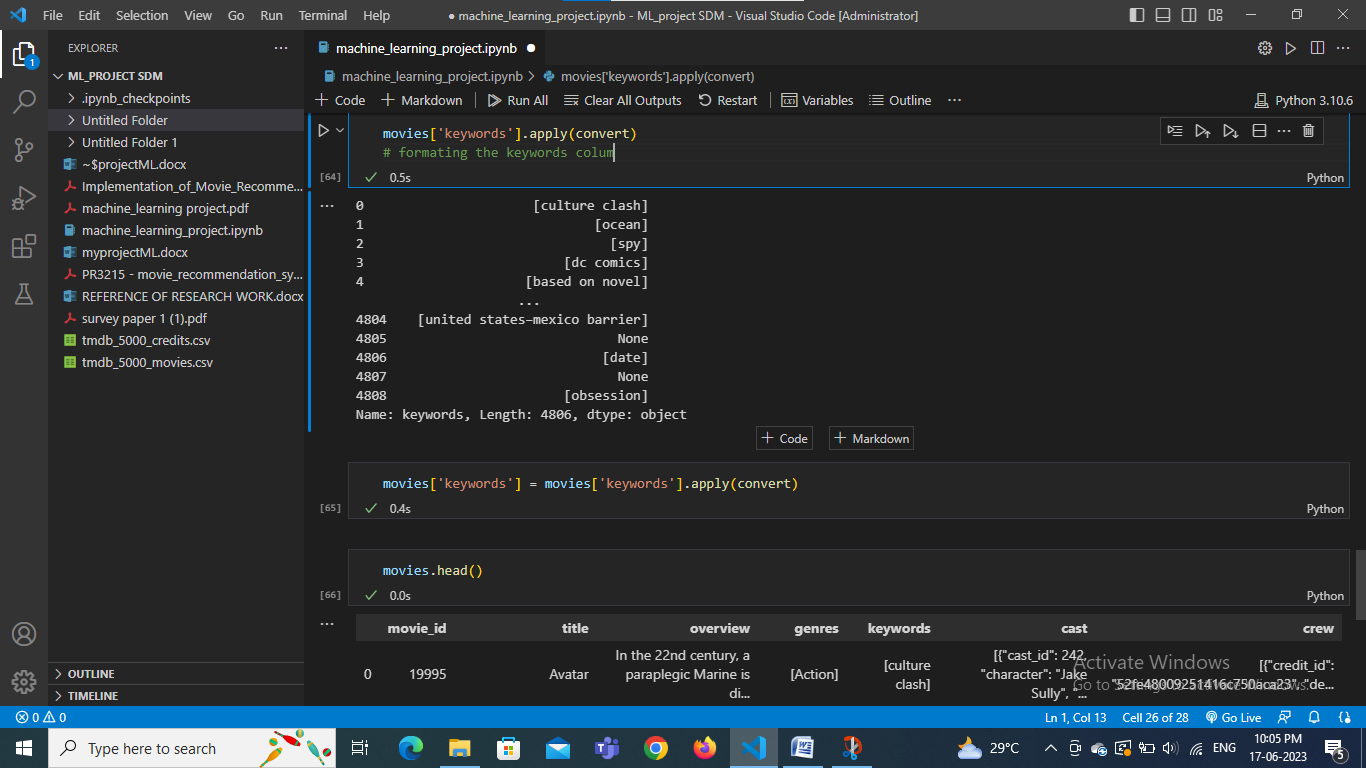
We converting the strings tags to list so that we can perform our task on it



Formatting the geners part



Formatting the keywords column



**Feature selection:**

**Model selection:**

**Model training**

**Test the model**

**Deployment**