iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** V **Month of publication:** May 2022

DOI: https://doi.org/10.22214/ijraset.2022.42255

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

Intelligent Movie Recommendation System Using AI and ML

D.S. Hirolikar[1], Ajinkya Satuse[2], Omkar Bhalerao[3], Pavan Pawar[4], Hrithik Thorat[5]
[1]*Professor,* [2, 3, 4, 5] *Student, Department of Information Technology, Pune District Education Association's College of Engineering, Manjari Bk., Pune, India.*

Abstract: ***Recommender system are systems which provide you with a similar type of products or solutions and results, you are looking for. For example, if you go to a Clothing shop, you ask for a T-shirt with different designs or different colors, Then the shopkeeper recommends you with different colors. This recommending task for websites is done by recommending systems. A recommendation engine uses several algorithms to filter data and then recommends the most relevant items to consumers. A Movie Recommender system will recommend the most relevant and connected movie for the given category of search, if a user visits a movie site for the first time, the site will have no previous history of that user. In such cases, the user can search for their movie recommendations based on genre, year of release, director or actor and their favorite movie itself to get a new movie recommendation.***

Keywords: ***Movie Recommendation Systems, Content-Based Filtering, Movie recommendation, machine learning project.***

I. INTRODUCTION

In the era of 21st Century, and the increasing e-commerce over the internet. Online shopping and entertainment industry are on peak levels. Online Everything will be a new normal in upcoming years. Imagine you are shopping online on website like amazon.com. They have over 60 million products for sale and the same goes for flipkart and other ecommerce websites.

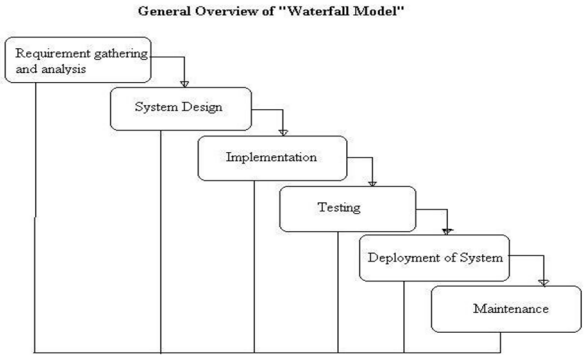
Entertainment websites like Netflix and Amazon Prime and Hotstar have over 10 million movies and series to watched.

If you want anything specific from these Websites, you can simply search for it. But, What about the rest of the products? If you want something similar or better product than you are search results. If you searching overall, it will like searching a golden tree in a Forest. You will be lost and never find your way out.

That's where, recommendation systems becomes your ally. Recommendation System plays an important role of being a guide in the systems of Amazon, Netflix and etc. Without Recommendation Systems, Many E-commerce and Entertainment websites will be like a database and You will need to be sure of what you are looking for. It will be a great loss for these companies, if people don't buy their products or don't watch any movie. Similarly, I will be great disadvantage for users, if they can't get the necessary product.

Therefore, it is an industrial and user necessity to have a Recommendation System embedded into various websites. We have decided to learn and implement such Recommendation System and take it on next level.

II. METHODOLOGY



©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

A. Requirement/Data Gathering

Data is the most important and foundation for machine learning projects. Gathering data from various datasets is key for a recommendation system. The more the data available the better the recommending results.

B. Pre-processing

In the pre-processing stage, Filtering and making ready the data for the project, we will make some changes such as, we will build tags that will describe the data and help us to calculate its similarity with other data.

1) *System Designing:* In this system design phase, we design the system which is easily understood by the end-user i.e., user friendly. We design some UML diagrams and data flow diagrams to understand the system flow and system module and sequence of execution.

C. Model Implementation and Testing

We will create a framework of data that will coordinate with the code this phase involves the core part of our project that is coding and model designing. The model will make sure the project works well at the local level.

The different test cases are performed to test whether the project module is giving the expected outcome in the assumed time.

D. Website Designing

After we have created a working model, we will create the same into a website. This stage will involve designing an immersive UI.

E. Deployment of System

Once the functional and non-functional testing is done, the product is deployed in the virtual environment or released into the local hosting like Heroku, over the internet.

III. OVERVIEW OF RECOMMENDATION SYSTEM

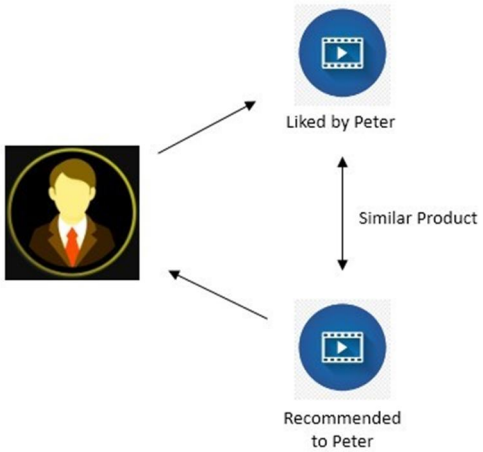
A. Recommender System

Recommender system are systems which provide you with a similar type of products or solutions and results, you are looking for. For example, if you go to a Clothing shop, you ask for a T-shirt with different designs or different colors, Then the shopkeeper recommends you with different colors. This recommending task for websites is done by recommending systems.

B. Types of Recommender System

1) *Content Based Recommender System:* Content based recommender system, recommends by the similarity of the content or the product. In our case, it will be the Movies. The system compares the two products and calculates its similarity. This operation is performed on every single product over the system. Whenever, The Similarity of one product is calculated with every other product present in the database. For example, while listening to songs, YouTube suggests similar language songs.

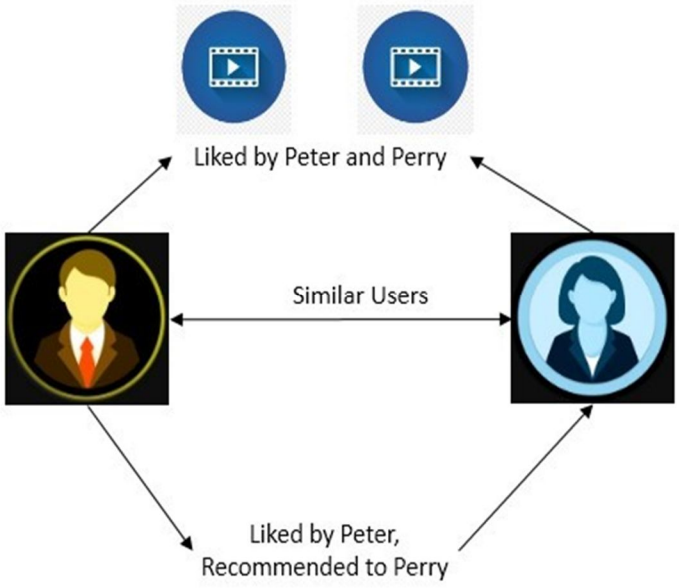
This is called Content based Recommendation.



©IJRASET: All Rights are Reserved

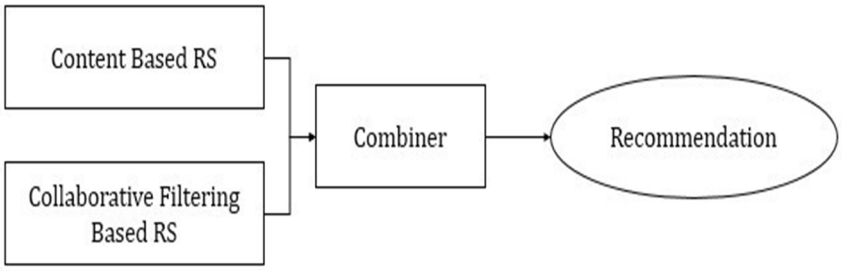
2) *Collaborative Recommender System:* In Collaborative Recommender System, The Data from users is collected to recommend different products. The Users using the system collaborate with each other to recommend products. The Similarity between Users is calculated and matched to recommend products to another similar user. For example, User A and User B are similar. The Products liked by User A are recommended to User B. Similarly, the products used by User B can be recommended to User A. This type of recommendation system can be seen in Facebook and Social Media platforms, where you are suggested the friends or videos which are seen by your friends.

Therefore, it is called Collaborative Recommendation System.



3) *Hybrid Recommendation System:* Hybrid Recommendation System is a combination of both Content based and Collaborative Recommendation system. In this, The Content or Product is recommended based on the data provided by the User. The System will provide you with results based on your previous actions and history. This type of recommendation system uses various data from different platforms and merge it together for better user experience.

Google Search results and suggestions are a great example of Hybrid Recommendation System.



C. Comparison

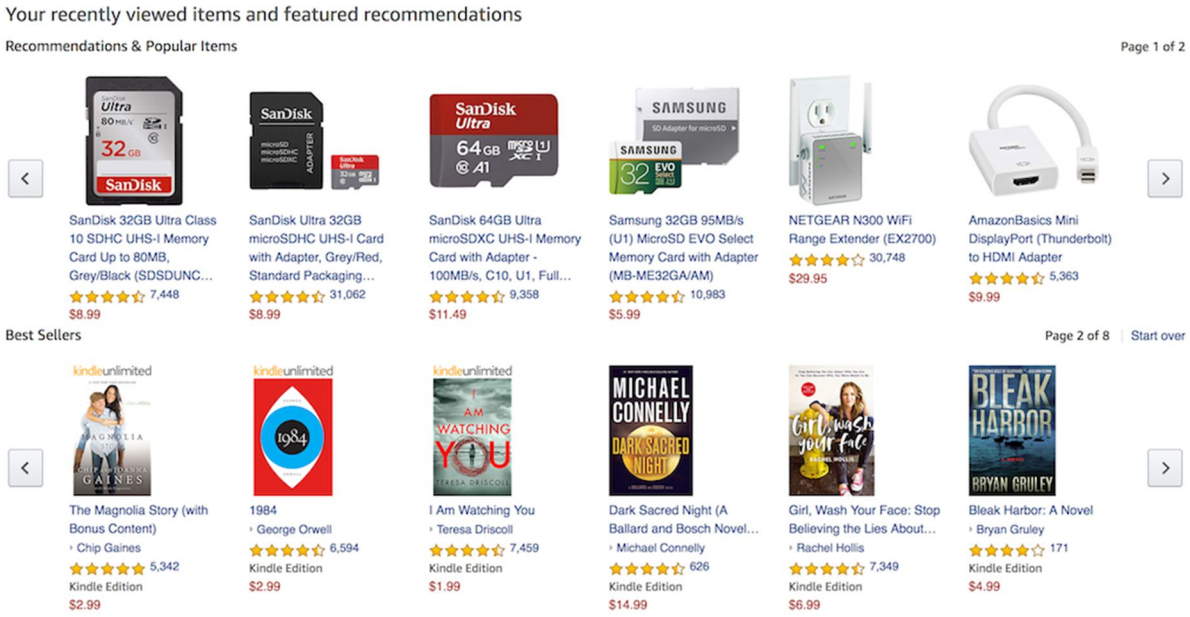
Each Recommendation System has its own advantage and disadvantage. The deciding factor remains that where the system is going to used and how effective the selected recommendation system will be in those conditions.

Famous Recommendation Systems:

Personalized recommender system analyzes a huge amount of user behavior data and provides personalized content to different users. E-commerce, movie, video, music, social network, reading, local based service, email and advertisement are some of the fields that widely use this system. It improves click rate and conversions of the website.

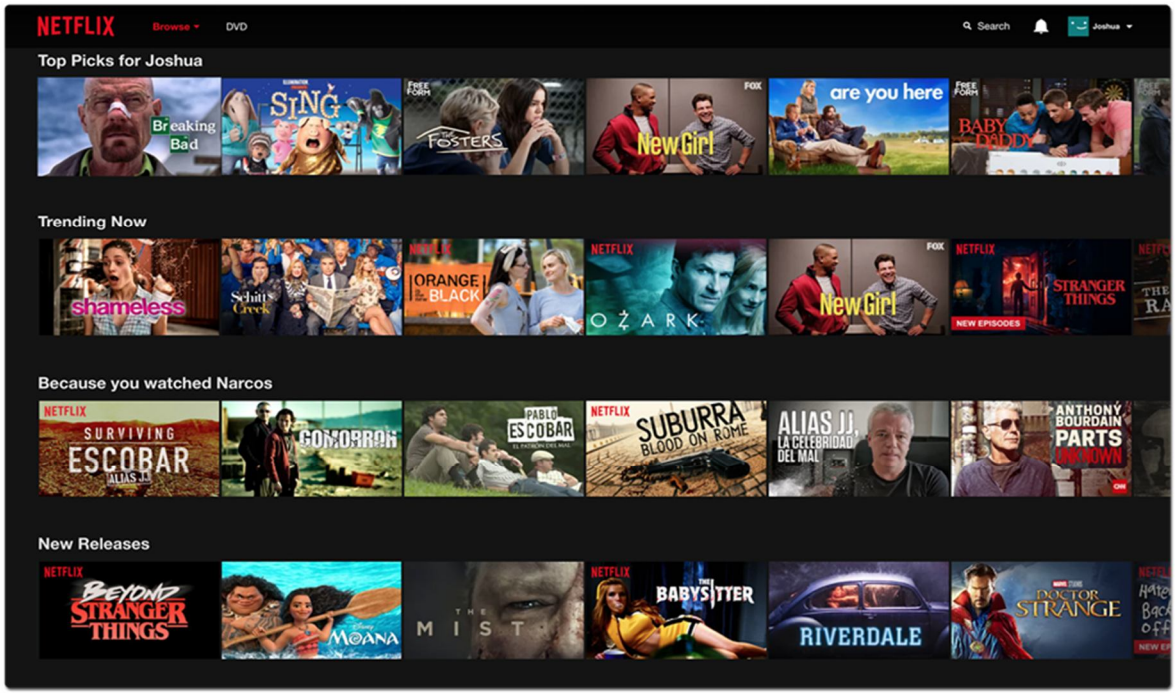
D. E-Commerce

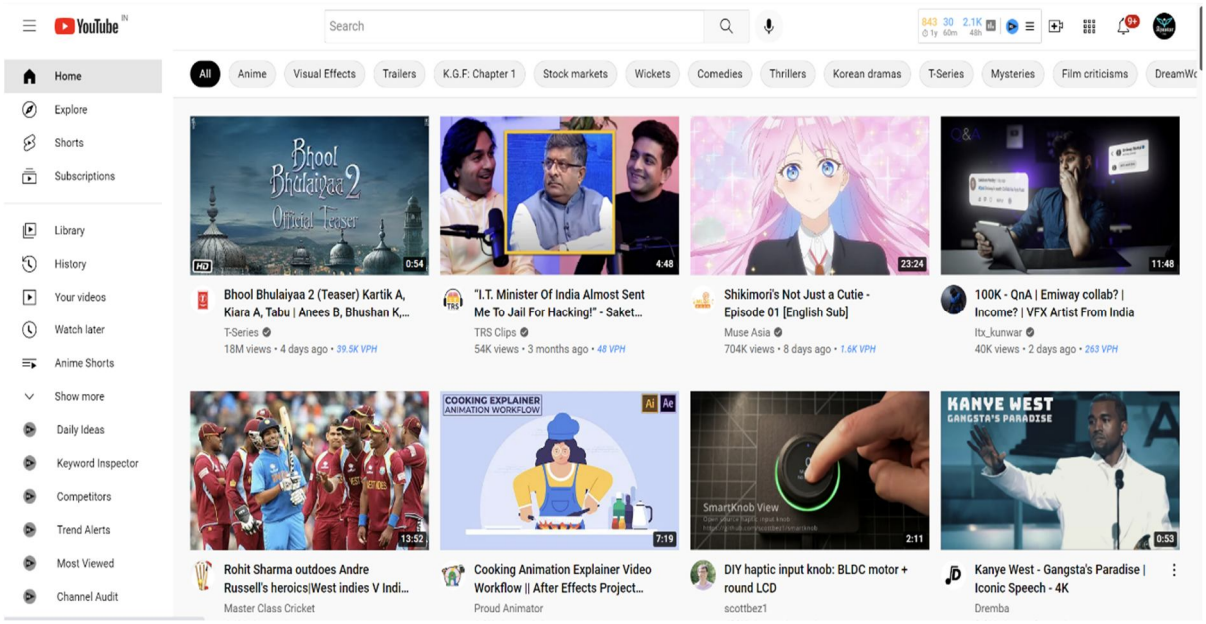
Websites like Amazon.com and Flipkart use Recommender Systems based on User previous purchase history. Also, these recommender systems use Filters to select the right product to be suggested. The Recommender also suggests the products which can be a used to with other products or bought at the same time by other users/customers. Following are some of the recommending pages over Amazon:



E. Movie and Video Recommenders

Netflix and Hotstar are the perfect example of Movie and video recommender systems. However, most upgraded and user-friendly recommendation can be seen in YouTube websites. The Similar content is recommended which is already seen or liked by the user. The system uses a content-based system which also has parameters called Tags which include Genre, language, etc. which defines the movie or the video.





IV. INTELLIGENT MOVIE RECOMMENDER SYSTEM

A. Datasets and Pre-processing

The Collection of Movies and their information was a huge task in normal circumstances. However, For the sake of this project, we have used publicly available datasets of tomb dataset of 5000 movies. It has mostly popularized movies from Hollywood and Bollywood that are known by most of the users. The datasets contain useful information regarding the movie such as movie name, cast (actors, actress and their roles), crew (directors and other members), Genre, release date, description, etc.
The data provided by these datasets has to processed and should be prepared for the use of recommendation.

B. Extraction of Data

The extraction of data takes in account the required tables and information regarding a movie. Any unwanted data should be neglected. For example, the movie length and movie budget doesn't affect the similarity of movies. The null values are also to be neglected for avoiding further missing errors and difficulties.

C. Creation of Tags

A data frame is a set of result of only necessary values. In our case, it would be Movie Title, Genre, Release Date, Cast and Crew.
To make an intelligent movie to movie recommendation, we need to create a single table named 'Tags' which can be combination of all the available data and keywords. Each Tag will be representing a single movie from the dataset.

D. Normalization

To reduce the occurrence of similar worlds and stop words (on, the, are, is, that). All the available data should be first normalized with each single variable. The Count Vectorizer is used to manipulate the data and eliminate these stop words. We can use the Natural language processing model (not) to perform such operation. The Function called Porter Stemmer would replace similar words with a single word. For Example, 'Loving, Loved, Lover' would be replaced by a single word 'love'. This will make calculating similarity between the tags more easy and more accurate.

E. Vectorization

The created Tags would now be recognized as single word and can be converted into vectors.
Vectors are points on a graph which have co-ordinates. For e.g. A (2,3)
The Conversion of Text to vectors can be performed by using text. Vectorization method.
The methods available are TF-IDF, Word to Vic, etc.
'Bag of Words' is such a vectorization technique which is easy to understand and easiest to work on.

F. Bag of Words

In Bag of Words, We Combine all the words in Tags into one Single long word.
For ex. Tag1 + Tag2 + Tag3 + …. = Tag
As we have 5000 movies dataset, we get 5000 tags representing 5000 movies.
Now, we have to calculate the 5000 common words which describe each movie.
By calculating the frequency of words in each tag, we can get these 5000 words.

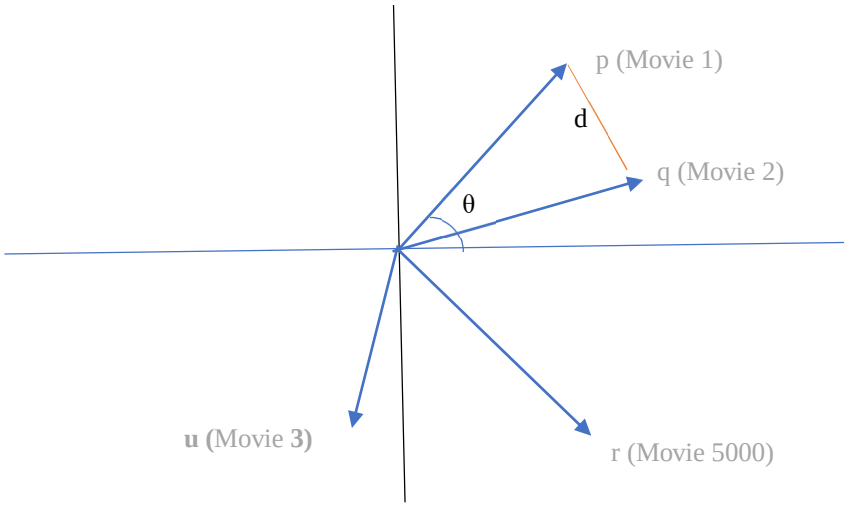
Let's say, Word 1 = Action
Word 2 = Adventure
Word 3 = War
Word 5000 = 2021

G. Matrix of Vectors

In Vectorization, we got 5000 words representing 5000 movies.
Now, we create a matrix of 5000 movies x 5000 words (5000,5000)

	Word 1	Word 2	Word 3	Word 5000
Movie 1	5	7	1	4
Movie 2	5	5	0	4
Movie 3	6	1	4	0
Movie 4	3	3	0	3
Movie 5000	0	2	3	7

Movie Vector Representation
Every movie in the dataset is converted into A vector.
Graphically it can be represented as,



H. Similarity

To Calculate the Similarity between two movies, we need to calculate the distance between their vectors.
There are two methods to calculate this distance:

1) Euclidian Distance
2) Cosine Distance

I. Euclidian Distance

Euclidian Distance is the distance between tip of one segment to the tip of another segment. We can call it Tip-to-tip distance between two-line segments. It is useful distance calculation technique but at the same time not very effective.
The Euclidian Distance between two movies cannot be accurate measure to calculate their similarity.
Formula for Euclidian distance:

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

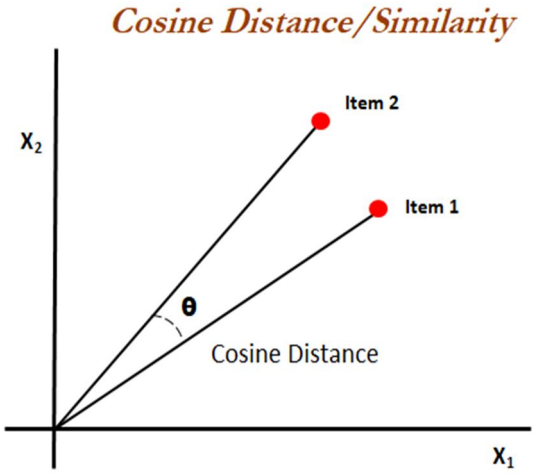
J. Cosine Distance

Cosine Distance is measured by the taking the cosine of the angle between the two vectors. It is very effective measure of distance calculating as the value lies between 0 to 1.
Formula for Cosine distance:
Cos (θ) = Distance between p (movie 1) and q (movie 2)

K. 3.7 Cosine Similarity

Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis.

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

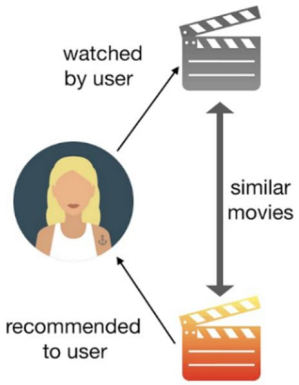


L. Retaining the calculated Similarity

All the vectors will be passed through the Cosine-Similarity function. Every Movie will have its own similarity score all other movies. We will sort them while using the greatest similarity score at first.
The Calculated similarity will be stored into a separate file, so that the system can work as fast as possible. Whenever the recommend function is called, The Similarity file would be fetched to inquire the required similar movies.

V. IMPLEMENTATION

We are using content-based recommendation system as it is easier compared to collaborative and hybrid system.
Many small businesses used content-based filtering in their E-commerce website and online market.



We will be using the cosine similarity to calculate a numeric quantity that denotes the similarity between two movies. We use the cosine similarity score since it is independent of magnitude and is relatively easy and fast to calculate. Mathematically, it is defined as follows:

We are now in a good position to define our recommendation function. These are the following steps, we'll follow:

1) Get the index of the movie given its title.
2) Get the list of cosine similarity scores for that particular movie with all movies. Convert it into a list of tuples where the first element is its position and the second is the similarity score.
3) Sort the aforementioned list of tuples based on the similarity scores; that is, the second element.
4) Get the top 10 elements of this list. Ignore the first element as it refers to self (the movie most similar to a particular movie is the movie itself).Return the titles corresponding to the indices of the top elements. While our system has done a decent job of finding movies with similar plot descriptions, the quality of recommendations is not that great. "The Dark Knight Rises" returns all Batman movies while it is more likely that the people who liked that movie are more inclined to enjoy other Christopher Nolan movies. This is something that cannot be captured by the present system. We are going to use a method called text vectorization

The credits genres actors and keywords are all combined and converted into tags.

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000,stop_words='english')
```

```
vectors = cv.fit_transform(new_df['tags']).toarray()
```

The “tags” are something that describes the movie.

```
In [41]: movies.head()
```

Out[41]:

	movie_id	title	overview	genres	keywords	cast	crew	tags
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[culturecIash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCameron]	[In, the, 22nd, century,, a, paraplegic, Marin...
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbinski]	[Captain, Barbossa,, long, believed, to, be, d...
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMendes]	[A, cryptic, message, from, Bond's, past, send...
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman]	[ChristopherNolan]	[Following, the, death, of, District, Attorney...
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[AndrewStanton]	[John, Carter, is, a, war-weary,, former, mili...

nltt is natural language processing library.
We used porter stemmer which is used to combine and split text (loving,love,lover) substitute it with it one word 'lov' to avoid error in calculating similarity

```
: import nltk

: from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()

: def stem(text):
    y = []

    for i in text.split():
        ps.stem(i)
```

The tags are transformed into vectors by text.vectorization.
We use the technique "Bag of words" to calculate the similarity between the tags.
We use the library scikit.learn.

```
]: from sklearn.metrics.pairwise import cosine_similarity

]: cosine_similarity(vectors)
```

We are going to calculate theta.
This theta is the distance between two segmentas i.e. vectors.
Cosine_similarity has a value from 0 to 1.
It calculates the angles between the vectors and converts them into 0 to 1 value.

```
In [34]: similarity

Out[34]: array([[1.        , 0.08964215, 0.06071767, ..., 0.02519763, 0.0277885 ,
                 0.        ],
                [0.08964215, 1.        , 0.06350006, ..., 0.02635231, 0.        ,
                 0.        ],
                [0.06071767, 0.06350006, 1.        , ..., 0.02677398, 0.        ,
                 0.        ],
                ...,
                [0.02519763, 0.02635231, 0.02677398, ..., 1.        , 0.07352146,
                 0.04774099],
                [0.0277885 , 0.        , 0.        , ..., 0.07352146, 1.        ,
                 0.05264981],
                [0.        , 0.        , 0.        , ..., 0.04774099, 0.05264981,
                 1.        ]])
```

We will create a function 'recommend' which can take 'Movie name' as an input and search its index position into the similarity file. First 5 greatest score will be the most similar movies based on the tags we provided and the score calculated by the similarity function.
Their names are also retained along with their index positions.

```
]: def recommend(movie):
    movie_index = new_df[new_df['title'] == movie].index[0]
    distances = similarity[movie_index]
    movies_list = sorted(list(enumerate(distances)),reverse=True,key=lambda x:x[1])[1:6]

    for i in movies_list:
        print(new_df.iloc[i[0]].title)
```

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

We will pass all the vectors (movies) through the function 'recommend'. It will calculate the similarity of one vector with another vector. We will sort this array according to maximum calculated similarity. The first five movies will be our output. As shown below,

```
In [37]: recommend('Batman')

Batman
Batman & Robin
The Dark Knight Rises
Batman Begins
Batman Returns
```

A. Creating other Recommending Filters

Similarly, we can perform the same operations on the remaining tables of our Dataset.

We will calculate similarity based on Genre – Category, Year of Release, Actors, Directors and retain their similarity in separate files.

```
In [37]: genrelist

Out[37]: ['action',
          'adventure',
          'animation',
          'comedy',
          'crime',
          'documentary',
          'drama',
          'family',
          'fantasy',
          'foreign',
          'history',
          'horror',
          'music',
          'mystery',
          'romance',
          'sciencefiction',
          'thriller',
          'tvmovie',
          'war',
          'western']
```

```
In [38]: recommend ('crime')

Wall Street: Money Never Sleeps
Black Mass
Catch Me If You Can
Casino
American Hustle
Mean Streets
21
Black Water Transit
Blow
Once Upon a Time in America
```

```
In [55]: recommend2 ('2015.0')

Avengers: Age of Ultron
Jurassic World
Furious 7
The Good Dinosaur
Jupiter Ascending
Inside Out
The Lovers
Tomorrowland
The Hunger Games: Mockingjay - Part 2
Terminator Genisys
```

©IJRASET: All Rights are Reserved

```
In [69]: recommend4 ('JohnnyDepp OrlandoBloom KeiraKnightley')

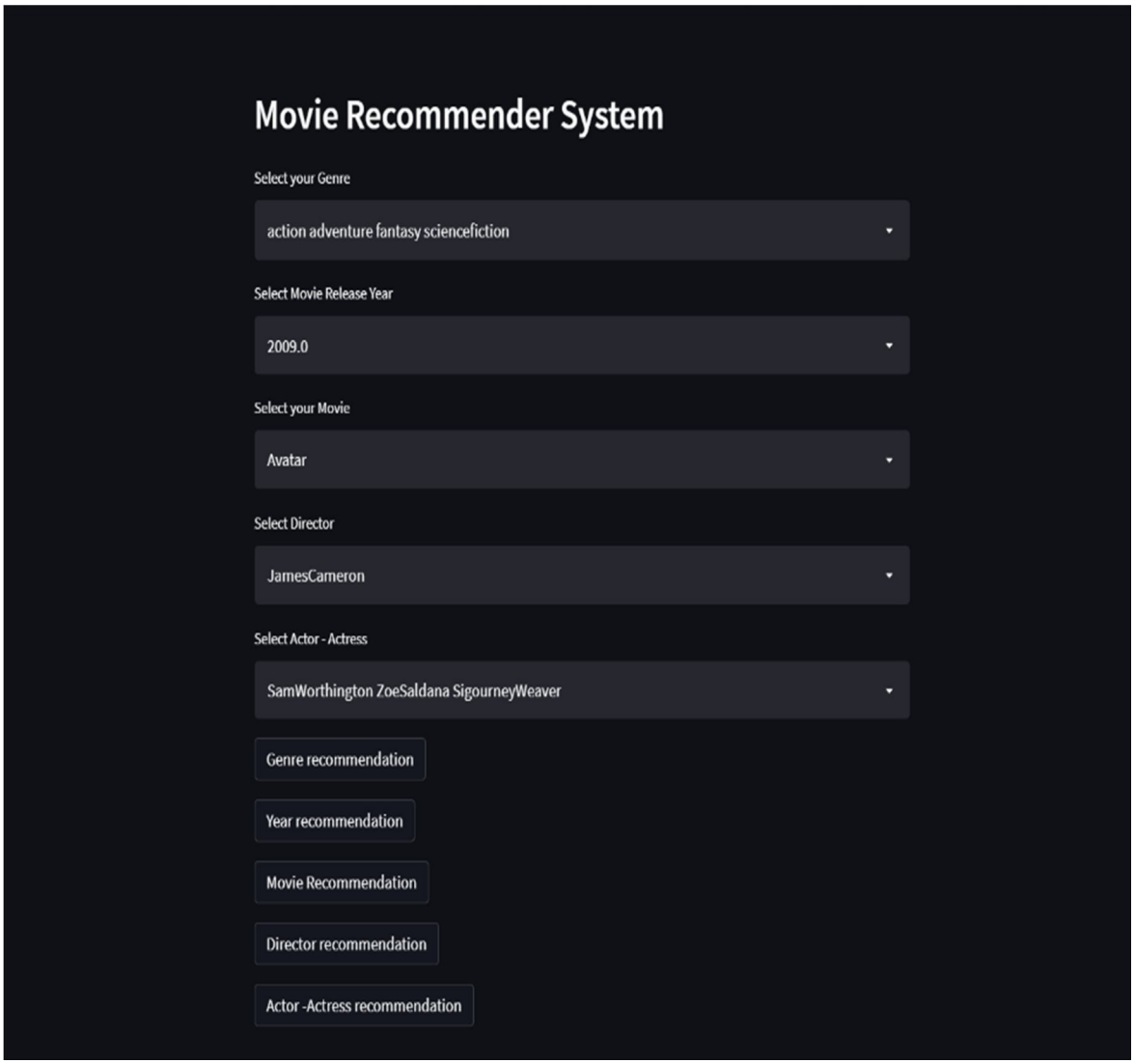
         Pirates of the Caribbean: Dead Man's Chest
         Pirates of the Caribbean: The Curse of the Black Pearl
         The Lone Ranger
         Pirates of the Caribbean: On Stranger Tides
         Alice in Wonderland
         Alice Through the Looking Glass
         Charlie and the Chocolate Factory
         Dark Shadows
         Troy
         Rango
```

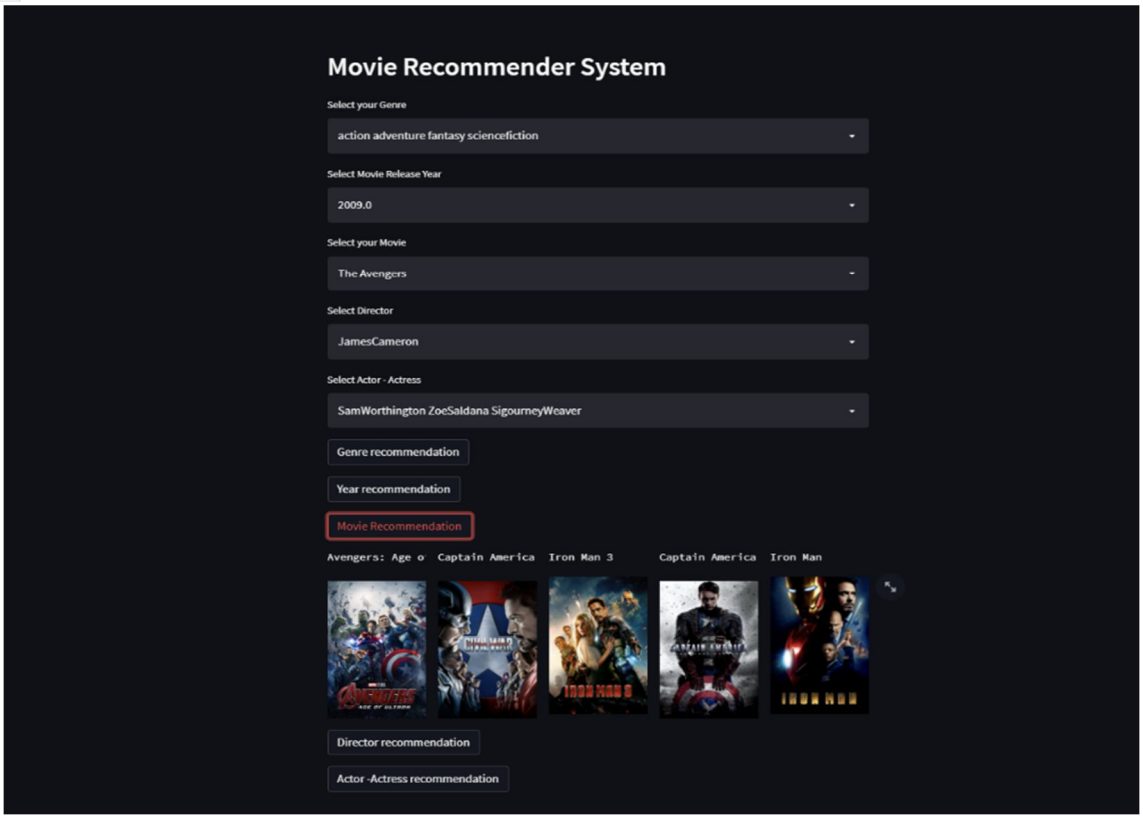
Each Function will have a different result based on the selected names from the respective entity.

B. System Design and Model:

Using PyCharm as an IDE for python application development. Creating a virtual environment, We can create a private website interface using Streamlit python library. Streamlit is an open-source python library for creating and sharing web apps for data science and machine learning projects. The library can help create and deploy data science solutions in a few minutes with a few lines of code.

VI. RESULTS





VII. CONCLUSION

The recommendation systems can be enhanced for present and future requirements for increasing the quality and for better recommendation results.

Recommendation system can become your virtual guide on E-Commerce platforms when powered with AI

It will be a great loss for companies like Amazon and Netflix if people don't buy or don't watch their product

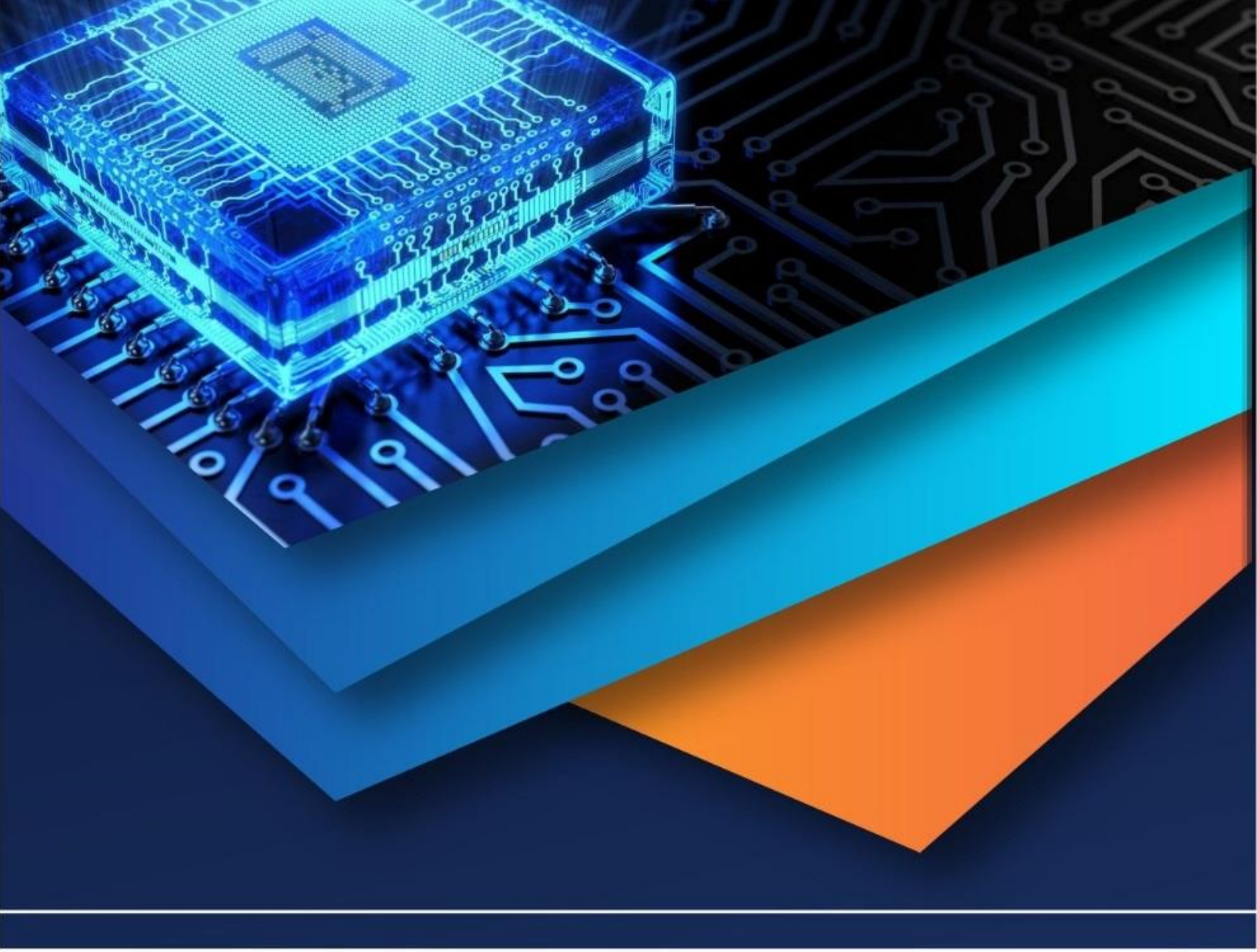
With the ever-increasing demand for machine automated solutions 'ML' has become one of the rapidly evolving technologies along with AI and Data Science.

Recommender systems will be used in the future to predict demand for products, connect buyers and sellers and eventually become the backbone for the supply chain.

Mega companies like Amazon, Netflix and Facebook need recommendation systems now more than anything with respect to the increasing products and users.

REFERENCES

[1] Mahesh Giyani and Neha Chourasia "A Review of Movie Recommendation System: Limitations, Survey and Challenges"
[2] Nirav Raval, Vijayshri Khedkar Moviellaborative Filtering Based Moive Recommendation System"
[3] Bhusan K. and Sripant "Recommendation System: Literature Survey and Challenges.
[4] 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.
[5] Bilge, A., Kaleli, C., Yakut, I., Gunes, I., Polat, H.: A survey of privacy-preserving collaborative filtering schemes. Int. J. Softw. Eng. Knowl. Eng. 23(08), 1085–1108 (2013).
[6] Calandrino, J.A., Kilzer, A., Narayanan, A., Felten, E.W., Shmatikov, V.: You might also like: privacy risks of collaborative filtering.
[7] Research.ijcaonline.org
[8] Dataset: tmdb-5000-movies dataset.
[9] Documentation: sklearn and streamlit



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)