PART 1

RECOMMENDER SYSTEM

1.1.0 Introduction

Consumer behavior has shifted in the aftermath of the epidemic, as they seek more tailored services from brands. To help internet users with the issue of information overload and a fast shift to online shopping, then came Recommender System.

That moment you walk into a physical store or business and ask to speak to a salesman/ customer representative, whenever you want to buy a new product or service. You will tell them exactly what you want from the product, give a detail of what specification you are looking out for in the said product then you are in return asked some certain questions to help them better understand your needs. Following that, they'll give some suggestions based on your talk. You can sometimes rely on this discussion to help you decide which product is best for you. In the digital world, recommender systems do the function of a salesperson.

This paper presents an overview of recommender systems and goes over collaborative filtering in depth. It also identifies the drawbacks of standard recommendation approaches.

An easy explanation of what a recommender system would be is that a recommender system is a collection of several components rather than a single entity. That is why the phrase **"SYSTEM"** is included.

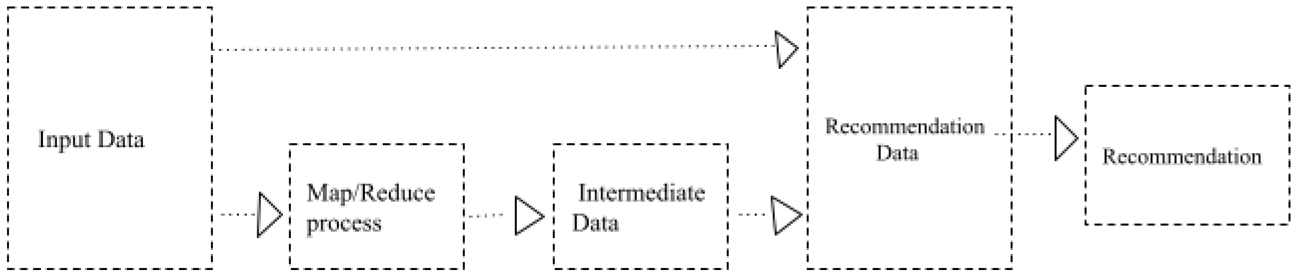
Recommender systems, also known as recommendation systems, are a type of data classification algorithm that attempts to predict a user's 'reviews' or 'choice' for an item (such as Music (songs), Articles/ journal(books), or Films (movies)) or public feature (e.g. people(celebrities), or groups(Association) and location) that they have not yet recognized, by using a model based on the item's characteristics (content-based approaches) or the user's social setting (collaborative filtering approaches). A recommender system is a program that makes suggestions to a user. (2022)

Recommender systems are software that allows you to deal with vast, complex data sets. They give the user a personalized view of these locations, highlighting items that are likely to be of interest to them. (International Research Journal of Engineering and Technology)

"In a typical recommender system, users offer recommendations as inputs, which the system then collects and routes to the right receivers." In some circumstances, the primary transformation is aggregation; in others, the system's value is in its capacity to match recommenders with those seeking recommendations. (P.Rensnick and H.R. Varian)

1.1.1 How is Recommender System (RS) applied?

Recommender systems have proven to be one of the most effective and widely used applications of machine learning technologies in business. This is a technique of information filtering that is used to forecast a user's preference and this application is used by collecting Customer’s data and automatically analyzing them by recommendation systems, which then provide personalized recommendations for your consumers. The main idea is to compare user profiles to known reference features in order to estimate how satisfied a user would be with a particular item offered.



The flow of data into a recommendation Engine.

Data collecting may be done in a variety of methods, both **Implicit**(The number of times it has been viewed, and the number of times it has been Purchased items, items utilized in social situations, and network analysis)and **Explicit** (Rates/Rank, Select from two options, Choose from a list of items) such as surfing history and purchases, as well as user ratings.

1.2 Challenges of recommender systems

One of the most important criteria of a recommender system is to provide consumers with exciting and relevant items. The quality of recommendations is directly connected to the users' trust in the system. If consumers do not receive satisfactory products and services, the recommendation engine may be deemed insufficient in terms of customer satisfaction, prompting users to seek out alternative solutions. As a result, to enhance preferability and efficacy, a recommender system must meet a certain level of prediction accuracy.

To this effect, the following can be tagged as the challenges faced by the recommender system

Accuracy: The most debated difficulty of recommender systems is accuracy, which is frequently explored in three ways: accuracy of rating forecasts, usage predictions, and item ranking (Shani and Gunawardana 2011)

Sparsity: this appears to be a significant difficulty in recommender systems; it occurs when a user has a huge matrix containing things to buy, movies to watch, or music to listen to. When the user didn't rate these things, sparsity emerged. While recommender systems rely on users' ratings in a matrix to determine who they should recommend to others,

Cold-start: occurs most often when a new user joins the site or when a new item is added to the system. To begin with, we don't know the new user's interests, and he hasn't yet rated any things. Second, to whom can we promote this new thing to others, even though no one has rated it as good or poor enough to be likely by users.

Scalability: Is a metric that measures a system's capacity to run efficiently with high performance while expanding its data. While the number of users or the number of products expanded, the recommender system had to continue to recommend items to them without changing. More computations and expensive hardware are required to do this.

Overspecialization: Users are only shown items that are previously known or defined by their user profiles, rather than discovering new items and other options.

Novelty: the recommended items must include some that have never been seen before(NEW ITEMS)

Serendipity: in addition to novelty, it may also be an objective that some of the recommended things are not only unfamiliar but also surprising to the user.

Privacy: One of the major issues in recommender systems is privacy. For recommender systems to recommend products that are relevant to the user's interests, we need to know certain information about the user's data. Users must understand what information is required to recommend more desirable goods to them, as well as how that information is utilized.

Shilling Attacks: This occurs when a hostile person or rival gains access to a system and begins giving fake ratings on certain things, either to enhance or decrease their popularity.

Gray Sheep: arises in collaborative filtering systems when a user's thoughts do not equal with any group and, as a result, the person is unable to profit from suggestions.

1.3 Dataset description

A Publicly available dataset taken from:
(https://www.kaggle.com/datasets/gargmanas/movierecommenderdataset?select=movies.csv)

The algorithms applied to the Movieid 193,609 data sets. It contains 100,837 ratings from 610 users on 9743 movie titles. All users in the data sets rated at least 4 movies.

There are two types of Data files. The rating data set has userid, movieid, rating, and timestamp sections (4 Columns). The movie dataset contains information about movies such as movie id, title, and genres(3Columns). Since movie ids are the same in both data sets, these data sets were connected thereby making them columns.
The data analytics tool is python.

1.4 Comparison of Approaches/ Techniques used to solve Recommender systems

To solve recommender systems, many techniques are used and studies demonstrate that recommender systems are grouped into three primary groups nevertheless, the Collaborative Filtering Recommender approach and the Content-based recommender approaches will be compared and contrasted.

1. The collaborative Filtering approach gives suggestions to its users based on the likes and dislikes of other users.It examines the preferences of comparable users and makes recommendations.

2. Content-based Recommendations generate suggestions to users based on similarities between new goods and previous favorites.

Content-based approach (CB)	**Criterion**	**Collaborative-filtering-based approach (CF)**
The features are derived from the information item.	Features	The features are influenced by the user environment (social, user preferences, patterns, etc.)
New users must first engage with the platform for a pattern to be created and for the platform to become optimal for their interests.	Challenge (Cold-Start)	New users must first engage with the platform for a pattern to be created and for the platform to become optimal for their interests.
Each user is given special attention. There is no presumption of belonging to a group or a community.	User treatment	It posits that a set of users have similar interests in items. It then attempts to predict an engaged user's undetected preferences using a linear scaled mixture of all other customers' choices.
based on product details and a database of the user's interests.	Computation	Similarity is calculated on two levels: user and item.
When rating data are scarce, content-based models are the best option for recommending things. This is because the user may have rated other things with identical features.	Advantage	The main advantage of collaborative suggestion is its ease of use.
Because specific words or content are used, content-based approaches give obvious recommendations. For instance, a user who has never utilized an item that has a specific set of words, that item is unlikely to be recommended. They are ineffective when it comes to making suggestions to new users.	Disadvantage	The lack of item descriptions makes it difficult for the recommend.

1.5 CHOSEN APPROACH: COLLABORATIVE FILTERING (CF) RECOMMENDER SYSTEM

Collaborative Filtering delivers great forecasting (Predictive) accuracy for recommender systems while using the lowest amount of data. Collaborative filtering employs similarities between users and items at the same time to generate suggestions, which addresses some of the limitations of content-based filtering.

This recommendation technique necessitates certain data or information about the user's preferences or past viewing history. It is founded on prior behavior or explicit feedback hence the reason for this recommender system since most other recommender systems do not employ this method because they demand data or are not trustworthy enough.
It is also a well-known and widely used algorithm in the industry. There are two major filtering algorithms in memory-based approaches. There is also a model-based approach, which is less reliable than the memory-based strategy.
For the purpose of this project, our focus will be on the Model-based approach.

PART 2 APPLICATION OF AL, ML, AND DS APPROACH

2A BENEFITS OF PRE-PROCESSING TECHNIQUES APPLIED

There was no preprocessing step taken for the chosen dataset because it was in its clean state with no null values as evidenced by the code snippet below;

```
In [8]:  1 dataset.isnull().sum()

Out[8]: userId     0
        movieId    0
        rating     0
        title      0
        genres     0
        dtype: int64
```

Fig 1

Instead, the step done before EDA was a merger of the rating dataset and the movie dataset as both had a common column title (movieid) to make one complete dataset. See *fig 2.*

```
In [3]:  1 dataset = pd.merge(df1,df2,on='movieId')

In [4]:  1 dataset.head()

Out[4]:
```

	userId	movieId	rating	timestamp	title	genres
0	1	1	4.0	964982703	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	5	1	4.0	847434962	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
2	7	1	4.5	1106635946	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
3	15	1	2.5	1510577970	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
4	17	1	4.5	1305696483	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy

fig 2.

Then removal of the unnecessary column timestamp (Data reduction)

```
In [5]:
# Dropping the timestamp column
dataset = dataset.drop(columns=['timestamp'])
```

Fig 3

It presents the benefit of simplicity and the ability to monitor precisely which columns emerged discarded when (with a simple change to the code).

2B EXTRACTING FEATURES/ATTRIBUTES

For the purpose of this project the extracting process was done using a sample of Exploratory Data Analysis (EDA) Univariate Analysis and Joint Analysis

I. Unique Count and Data Shape

The count of unique values in a dataframe, whether continuous, categorical, or something completely else like separate texts, necessitates a decent exploration step this explains the fig 4 accomplished by using a dataset nunique(axis = 0) and then plotting it in a bar graph.

```
In [7]:
# Looking at Unique Counts and Data shape
print(dataset.nunique(axis=0))
sns.barplot(x = dataset.columns, y = dataset.nunique(axis=0));
```

```
userId      610
movieId    9724
rating       10
title      9719
genres      951
dtype: int64
```

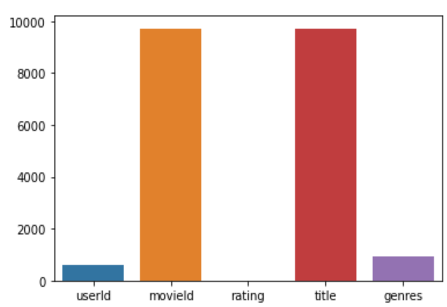


Fig 4

Univariate Analysis

This was used to evaluate the attributes of each variable to help better understand three questions stated below:

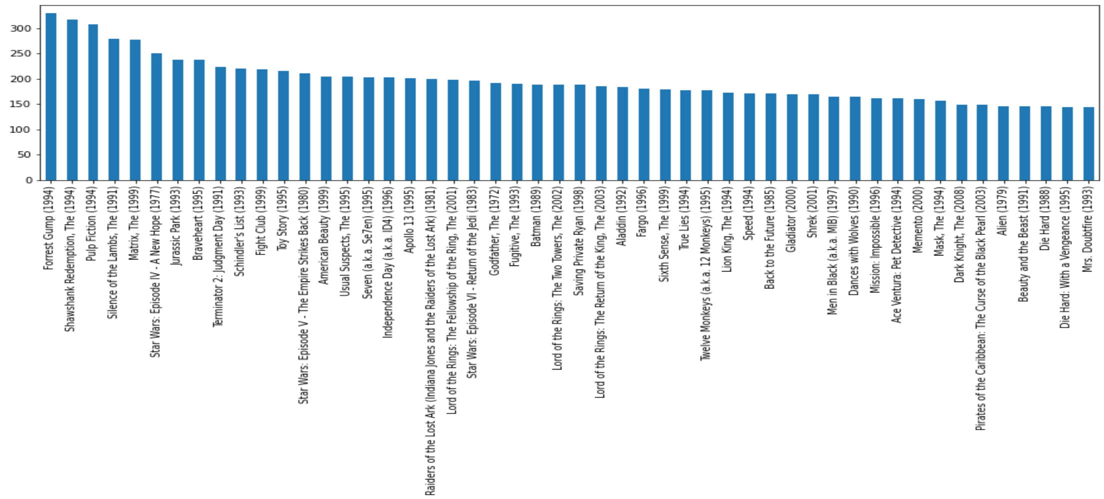
A). ***WHAT ARE THE MOVIES WITH THE MOST REVIEWS?***

Fig 5 below shows the movie with the most reviews in descending order with X-axis as Movie title and Y-axis as Rating with the movie title Forrest Group (1994) being the most reviewed which has no more than 350 ratings and the least rated Mrs. Doubtfire (1993)

Fig 5

In [8]:

```
# Univariate Analysis
plt.figure(1, figsize = (16,4))
dataset['title'].value_counts()[:50].plot(kind = 'bar') #take top 50 movies
plt.figure(2, figsize = (16,4))
dataset['userId'].value_counts()[:50].plot(kind = 'bar') #take top 50 users
plt.figure(3, figsize = (8,4))
dataset['rating'].plot(kind = 'hist');
```



B.) ***WHO ARE THE USERS THAT PROVIDE THE MOST REVIEWS?***

Fig 6 indicates that reviews are uniformly distributed and that the userid with the most reviews is shown in descending order, with userid '414' (X-axis)being the most reviewed with no more than 2700 reviews, and the least user id as 360

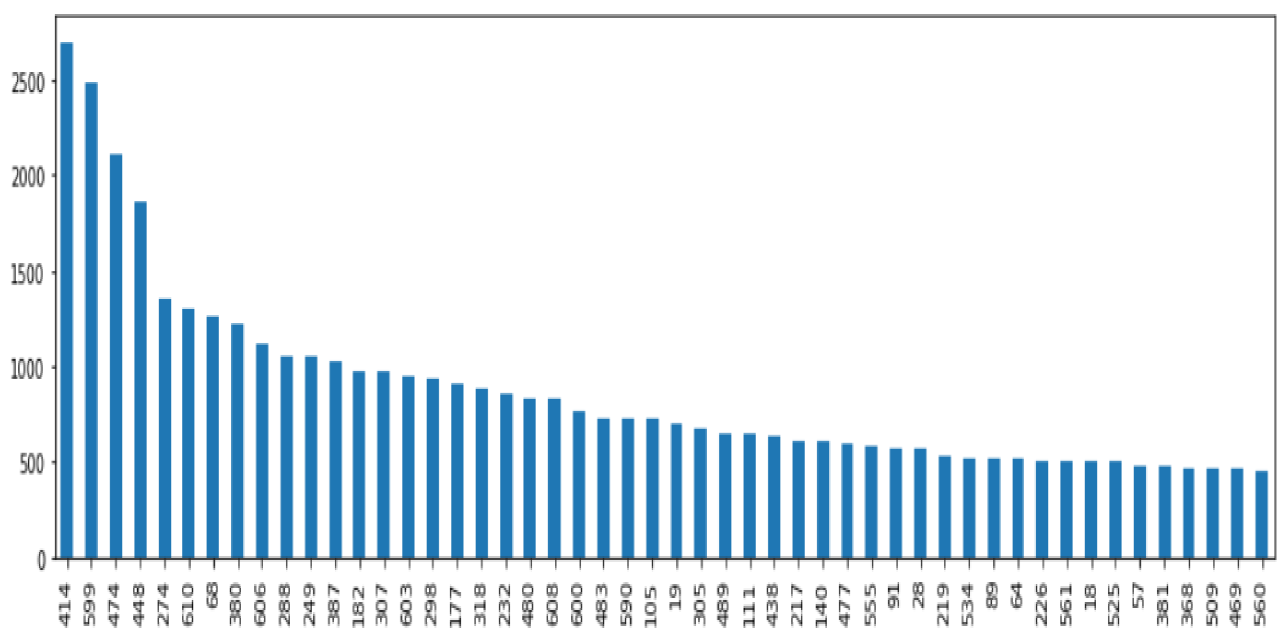


Fig 6

C.) ***WHAT DOES THE DISTRIBUTION LOOK LIKE FOR RATINGS?***

According to the graphs below, the majority of people will give a 4 out of 5-star rating.

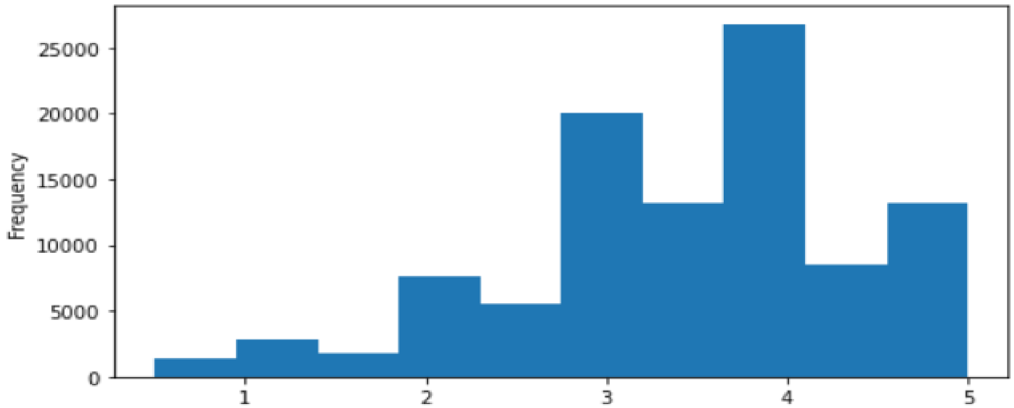


Fig 7

Labels as X= Rating
Y=Frequency

II. Joint Analysis

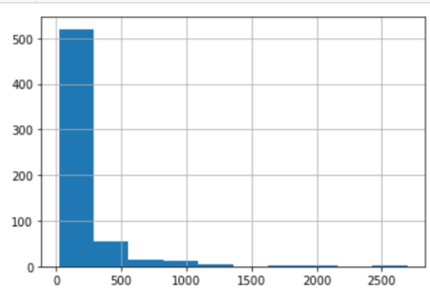
In contrast to univariate analysis, which offers a more thorough perspective on certain movies or users, aggregation typically refers to acting on a larger scale, such as groups rather than individuals, or broader areas rather than smaller ones.

1. ***WHAT IS THE DISTRIBUTION OF RATINGS GIVEN TO EACH MOVIE?***

The histogram demonstrates that 510 users (84%) rate between 0 and 250, while the remaining 16% (100 users) score higher.X=rating, Y=users

Fig 8

In [9]:
```
ratings_per_user_ID = dataset.groupby('userId')['movieId'].count()
ratings_per_user_ID.hist();
```

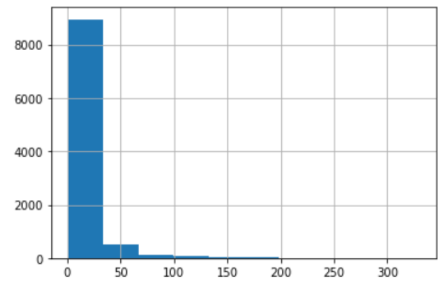


2. ***WHAT IS THE DISTRIBUTION OF USERS WHO PROVIDE RATINGS?***

The histogram illustrates of the 9,724 movies 9,000 which is 93% have fewer than 25 ratings.

Fig 9

In [10]:
```
ratings_per_movieid = dataset.groupby('movieId')['userId'].count()
ratings_per_movieid.hist();
```



With the respective charts shown above, thanks to the explanatory descriptive analysis (EDA) used, a clearer insight was derived

2C IMPLEMENTATION OF COLLABORATIVE FILTERING

The collaborative filtering technique (item-based approach) was used in the proposed recommendation system, which is much more easy and timely to use seeing as the item-based method can be undertaken offline and is non-dynamic, whilst the user-based method changes. The proposed technique euclidean distance first between specific movies and any other movie in the dataset using the KNN algorithm, next and evaluates the top k nearest relatively similar movies using mean squared similarity. Various similarity measures, such as cosine angle similarity, can also be used. Preceding are some of the tactics used in this proposed algorithm:

a). The k-nearest neighbors (KNN) technique is a basic, supervised machine learning algorithm that may be used to address classification and regression issues.

The technique is based on the premise that comparable data points (i.e. data points in close proximity in space) are likely to have similar labels (i.e. they tend to belong to the same class). As a result, the KNN method may be used to predict the label of a new data point based on the labels of surrounding data points in space. (Muhammad)

b). Matrix Factorization - Singular Value Decomposition (SVM)

In recommender systems, matrix factorization is a type of collaborative filtering technique. The user-item interaction matrix is decomposed into the product of two smaller dimensionality rectangular matrices via matrix factorization methods. (Wikipedia)

The Singular Value Decomposition (SVD), is a linear algebra approach that has been widely employed in machine learning as a dimensionality reduction tool. The SVD approach decreases the number of features in a dataset by lowering the space dimension from N to K (where KN). The SVD is utilized as a collaborative filtering mechanism in the recommender system. It employs a matrix format, with each row representing a user and each column representing a product. The ratings that people assign to things are the matrix's elements. (Kumar)

In [70]:

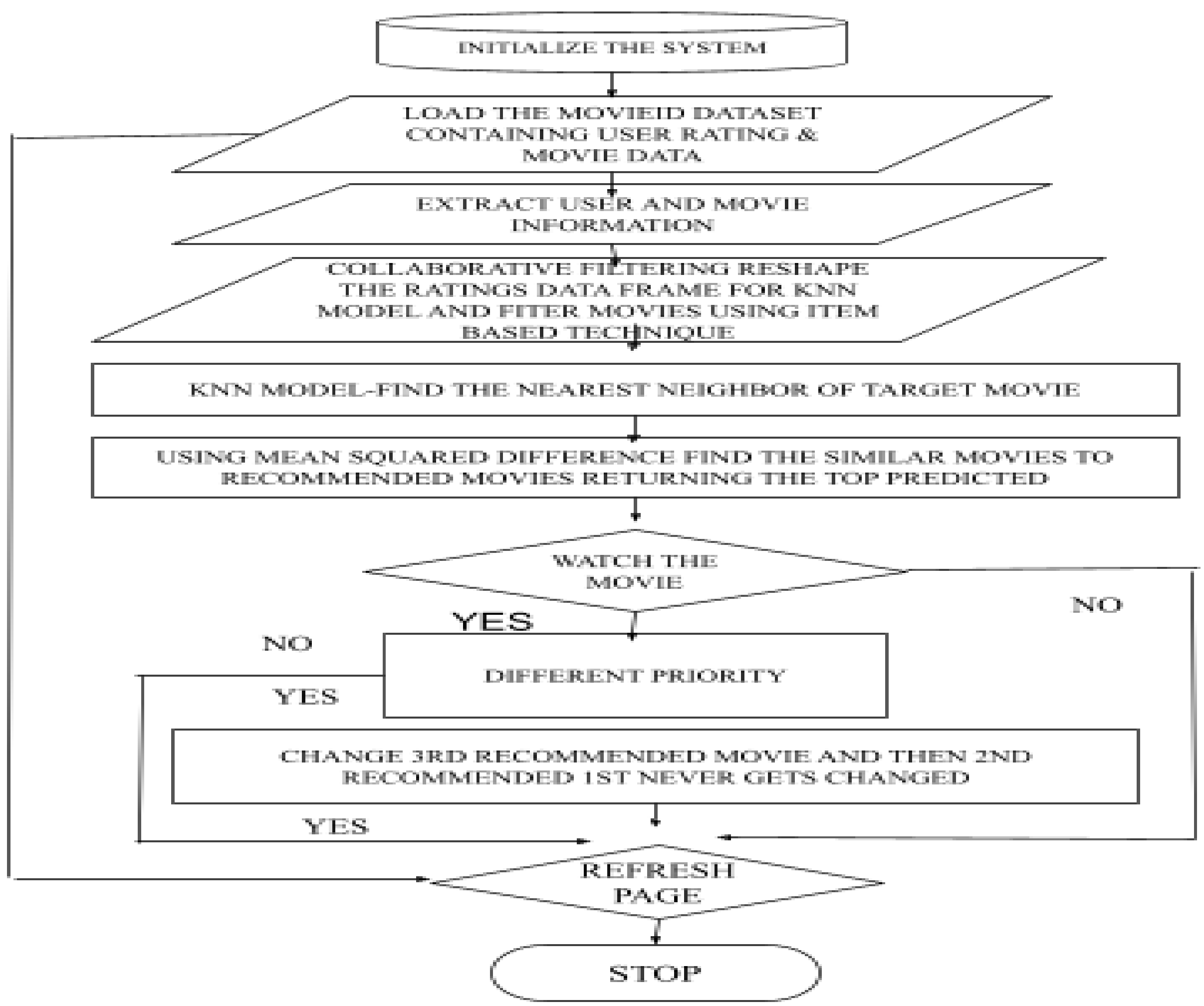
```python
# Fitting in different Algorithms model to our trainset
svd = SVD()
svd.fit(trainset)

knn_WithMeans = KNNWithMeans()
knn_WithMeans.fit(trainset)

knn_WithZScore = KNNWithZScore()
knn_WithZScore.fit(trainset)

co_Clustering = CoClustering()
co_Clustering.fit(trainset)
```

```
Computing the msd similarity matrix...
Done computing similarity matrix.
Computing the msd similarity matrix...
Done computing similarity matrix.
```

Out[70]: `<surprise.prediction_algorithms.co_clustering.CoClustering at 0x1e6608c2a90>`

Fig 10

The graphic above demonstrates how we may use Python's surprise package to construct multiple algorithms for movie recommendation.



FLOWCHART FIG 11

A movie dataset was used, which was described in detail in section 1.3, to build these methods. By limiting the data frame to only include popular movies, the datasets are filtered exclusively on the basis of their popularity.

Movie.csv

movieId	title	genres
1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
2	Jumanji (1995)	Adventure|Children|Fantasy
3	Grumpier Old Men (1995)	Comedy|Romance
4	Waiting to Exhale (1995)	Comedy|Drama|Romance
5	Father of the Bride Part II (1995)	Comedy

Fig 12

Ratings.csv

userId	movieId	rating	timestamp
1	1	4	964982703
1	3	4	964981247
1	6	4	964982224
1	47	5	964983815
1	50	5	964982931

Fig 13

Analysing the movie dataset (as shown in figure 12) the following python libraries were used Pandas, Matplotlib, NumPy, SnS, and, Surprise, on Jupyter Notebook to obtain efficient results.

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from surprise import SVD
from surprise import Reader, Dataset, SVD, SVDpp
from surprise import accuracy
from surprise import KNNWithMeans, KNNWithZScore, CoClustering
```

Fig 14

2D APPLICATION OF PREPROCESSING AND OUTCOMES

The Mean Square Difference Similarity on Knn WithMeans, Knn WithZscore, svd, and Co-clustering was used in this research. The Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were used to evaluate the clustering and performance of these models (MAE).

The prediction quality was measured using Mean Square Error (MSE) and Root Mean Square Error, which are primarily the average square difference between the actual and estimated values. The following formulas are provided:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

(Goswami)

Result of the RMSE and MAE of the different algorithms implemented

]:

	Model Name	Rmse	Mae
2	svd_100	0.639	0.497
4	knn_WithZScore	0.697	0.518
3	knn_WithMeans	0.702	0.525
1	svd_50	0.720	0.557
0	svd_20	0.780	0.603
5	co_Clustering	0.823	0.637

Fig 15

With svd_100 showing an RMSE of 0.639, this implies that it is the best algorithm that can be used for our movie recommendation which also shows in terms of a graphical visualization where we have svd_100 having the lowest point on the graph.

In [59]:
```
sns.lineplot(x = y['Model Name'], y = y['Rmse']);
```

Fig 16

svd_100 movie recommendation

]:

	movieId	actual	predicted
0	Rear Window (1954)	3.501631	4.926674
1	Usual Suspects, The (1995)	3.501631	4.883640
2	Hoop Dreams (1994)	3.501631	4.875801
3	Princess Bride, The (1987)	3.501631	4.846531
4	American Beauty (1999)	3.501631	4.838948
5	To Catch a Thief (1955)	3.501631	4.834058
6	Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1964)	3.501631	4.825415
7	North by Northwest (1959)	3.501631	4.821797
8	Shawshank Redemption, The (1994)	3.501631	4.810064
9	Lawrence of Arabia (1962)	3.501631	4.796926

Fig 17

The svd_100 recommends Rear Window (1994) as the top movie to watch closely followed by Usual suspects, The (1995) and Hoop Dream(1994).

APPLICATION OF MODEL FOR RECOMMENDATIONS

a.) Knn_with Zscore

	movieId	actual	predicted
0	Heat (1995)	3.501631	5
1	Seven (a.k.a. Se7en) (1995)	3.501631	5
2	Usual Suspects, The (1995)	3.501631	5
3	Bottle Rocket (1996)	3.501631	5
4	Braveheart (1995)	3.501631	5
5	Rob Roy (1995)	3.501631	5
6	Desperado (1995)	3.501631	5
7	Clerks (1994)	3.501631	5
8	Ed Wood (1994)	3.501631	5
9	Star Wars: Episode IV - A New Hope (1977)	3.501631	5

fig 18

b.) Co_Clustering

	movieId	actual	predicted
0	Heat (1995)	3.501631	5
1	Seven (a.k.a. Se7en) (1995)	3.501631	5
2	Usual Suspects, The (1995)	3.501631	5
3	From Dusk Till Dawn (1996)	3.501631	5
4	Bottle Rocket (1996)	3.501631	5
5	Braveheart (1995)	3.501631	5
6	Rob Roy (1995)	3.501631	5
7	Desperado (1995)	3.501631	5
8	Billy Madison (1995)	3.501631	5
9	Clerks (1994)	3.501631	5

Fig 19

On this dataset, using these models (Knn-Zscore & Co-clustering) to predict/recommend shows that the movies are essentially the same movie prediction, with both projecting a value of 5.

PART 3

CONCLUSION

Within this project, I considered calculating a user's relevance in the recommender based on the quality of suggestions received and recommendations produced by users. The database has 610 users, 193,609 objects, and 100,837 ratings, with each user rating a minimum of 20 items. The objects are movies of 9,743 titles, and the star ratings range from 1 to 5. Various graphs of the findings were produced by using the MovieId database.

The collaborative filtering process of the recommender system has the most effect on the outcomes of the data retrieved. Collaborative Filtering is based on predicting a user's interest in the preferences of a set of users who are regarded as similar to this user. Regardless of the technique used in the CF process, the analytical goal is to reduce overfitting by increasing the recommender accuracy; however, there are certain goals to consider, such as avoiding overspecialization circumstances, finding useful items, and recommendation credibility, precision, and recall measures.

The Mean Absolute Error (MAE) and its associated metrics: MSE & RMSE, and normalized mean absolute error are commonly used in similarity techniques to find the similarity given distinct users x and y: sim(x,y) relying on their ratings of things that both users have evaluated. As can be shown, there are no significant changes in the quality of suggestions received whether values of k-neighborhoods are employed.

The increase in performance and results produced using the suggested technique enables the creation and implementation that enable the monitoring and optimization of the collaborative Filtering process of the recommender findings.

REFERENCES

Goswami, Raunak. “Root-Mean-Square Error (RMSE) | Machine Learning.” *Includehelp.com*, 16 August 2018, https://www.includehelp.com/ml-ai/root-mean-square%20error-rmse.aspx. Accessed 21 April 2022.

International Research Journal of Engineering and Technology (IRJET). “Movie RecommendationSystem using Machine Learning Algorithms.” *Movie Recommendation System using Machine Learning Algorithms*, vol. 07, no. 04, 2020, p. 3696. *Introduction to recommender system*, https://www.irjet.net/archives/V7/i4/IRJET-V7I4718.pdf#:~:text=Abstract%20-Recommender%20systems%20are%20one%20of%20the%20most,are%20books%2C%20news%2C%20articles%2C%20music%2C%20videos%2C%20movies%20etc. Accessed 14 March 2022.

J. Bobadilla, F. Serradilla and A. Gutiérrez, "Recommender systems: Improving collaborative filtering results," 2009 7th International Conference on ICT and Knowledge Engineering, 2009, pp. 100-106, doi: 10.1109/ICTKE.2009.5397339.

Kumar, Vaibhav. “Singular Value Decomposition (SVD) In Recommender System.” *Analytics India Magazine*, 25 March 2020, https://analyticsindiamag.com/singular-value-decomposition-svd-application-recommender-system/. Accessed 21 April 2022.

Marwa H Mohamed et al. “Recommender Systems Challenges and Solutions Survey.” *Recommender Systems Challenges and Solutions Survey*, no. International Conference on Innovative Trends in Computer Engineering, 2019, pp. 1-155. *Recommender Systems Challenges and Solutions Survey*, https://www.researchgate.net/publication/331063850_Recommender_Systems_Challenges_and_Solutions_Survey. Accessed 14 March 2022.

Michael D. Ekstrand et al. “Foundations and Trends R 有 inHuman–Computer InteractionVol. 4, No. 2 (2010) 81–173c 有 2011 M. D. Ekstrand, J. T. Riedl and J. A. Collaborative Filtering Recommender Systems.” vol. 04, 2010, pp. 81-173, https://www.academia.edu/37912443/Ekstrand?email_work_card=view-paper. Accessed 14 03 2022.

Muhammad, Bilal. “What is knn algorithm?” *NF AI*, 20 March 2022, https://www.nfaicompany.com/what-is-knn-algorithm/. Accessed 19 April 2022.

P. Khurana & S.Parveen. “Approaches of Recommender System: A survey.” *International Journal of Computer Trends and Technology (IJCTT)*, vol. 34, 2016, http://ijcttjournal.org/2016/Volume34/number-3/IJCTT-V34P124.pdf. Accessed 14 03 2022.

P.Rensnick and H.R. Varian. “Recommender Systems: An Overview.” *Recommender Systems: An Overview*, vol. 40, no. 03, 1997, pp. 56-58. *Recommender Systems: An Overview*, https://www.researchgate.net/publication/220604600_Recommender_Systems_An_Overview. Accessed 14 march 2022.

Raham M.M et al. “"Machine Learning approach for Item-basedMovie Recommendation using the most relevant similarity techniques,."” *3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA*, vol. doi: 10.1109/HORA52670.2021.9461381., 2021, pp. 1-4, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9461381. Accessed 19 04 2022.

Shani, Gunawardana G. *Evaluating recommendation systems*. Boston,, Springer, 2011. *A review on deep learning for recommender systems: challenges and remedie*, https://daiwk.github.io/assets/Batmaz2018_Article_AReviewOnDeepLearningForRecomm.pdf#:~:text=Although%20existing%20recommender%20systems%20are%20successful%20in%20producing,from%20challenges%20such%20as%20accuracy%2C%20scalability%2C%20and%20cold-start. Accessed 14 March 2022.

Terveen, Loren. "Evaluating collaborative filtering recommender systems." ACM Transactions on Information Systems, 01 2004, https://www.academia.edu/4906714/Evaluating_collaborative_filtering_recommender_systems?email_work_card=view-paper. Accessed 14 03 2022.

Wikipedia. "Matrix factorization (recommender systems)." *Wikipedia*, https://en.wikipedia.org/wiki/Matrix_factorization_(recommender_systems). Accessed 21 April 2022.

W.Liang et al. "Difference Factor KNN collaborative filtering recommendation algorithm,." *Difference Factor KNN collaborative filtering recommendation algorithm,*, 2014, pp. 175-184. *In International Conference on Advanced Data Mining and Applications*, https://ieeexplore.ieee.org/document/9155879/figures#figures. Accessed 19 April 2022.