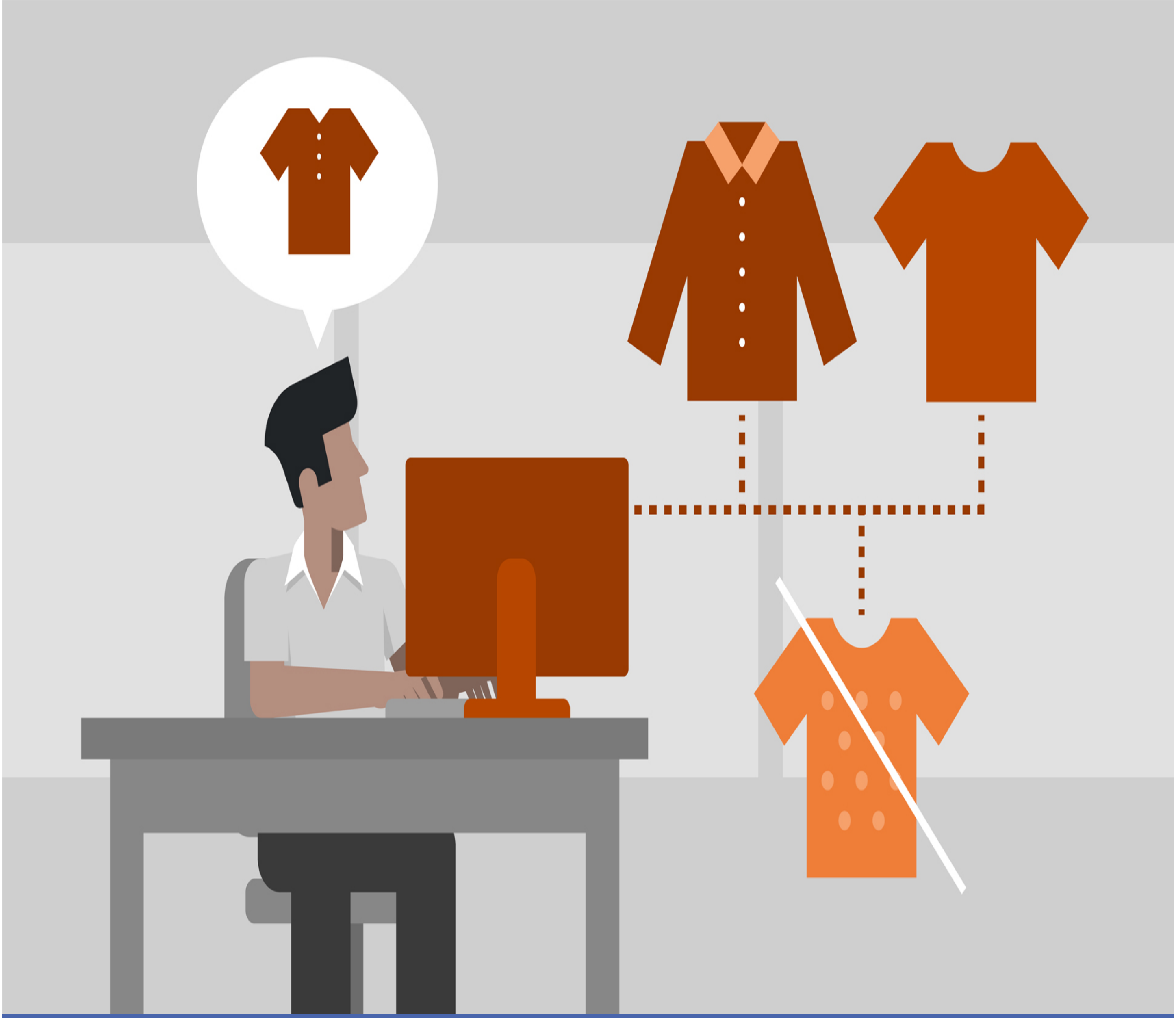


MOVIE RECOMMENDATION SYSTEMS

A guide to algorithmically predicting what your customers want

CONTENT

RECOMMENDATION SYSTEM

1.Introduction:

A recommendation system is a type of information filtering system which attempts to predict the preferences of a user, and make suggests based on these preferences. There are a wide variety of applications for recommendation systems. These have become increasingly popular over the last few years and are now utilized in most online platforms that we use. The content of such platforms varies from movies, music, books and videos, to friends and stories on social media platforms, to products on e-commerce websites, to people on professional and dating websites, to search results returned on Google. Often, these systems are able to collect information about a user's choices and can use this information to improve their suggestions in the future.

People use a variety of strategies to make choices about what to buy, how to spend their leisure time or what to read. Search engines help us a little bit. But Recommender systems automate some of these strategies with the goal of providing affordable, personal, and high-quality recommendations.

Recommender systems are a way of suggesting like or similar items and ideas to a user. It is an information filtering technique, which provides users with information, which he/she may be interested in.

Search engines focused more on Information retrieval but recommender system focused on Information Filtering. In search engines you see a query box where you type in what you're looking for and they bring back a list of results. But in recommender system you don't build a query, Recommendations engines observe your actions and construct queries for you so some content has just appeared on your screen that is relevant to you but you didn't request it.

2. Need of Recommender systems

{i}Value for the customer

Find things that are interesting: Recommender systems provide affordable, personal, and high-quality recommendations, which he/she may be interested in.

Narrow down the set of choices: A company with an inventory of thousands and thousands of items would be hard to product suggestions for all of its products, Recommendation system will narrow down the choices.

Help me explore the space of options: People generally like to be recommended things which they would like, and when they use a site which can relate to his/her choices extremely perfectly then he/she is bound to visit that site again.

Discover new things: Recommender systems_recommend products which are similar to the ones that a user has liked in the past. If you like an item you will also like a 'similar' item.

{ii}Value for the provider

Unique personalized service for the customer: Companies using recommender systems focus on increasing sales as a result of very personalized offers and an enhanced customer experience.

Increase trust and customer loyalty: The user starts to feel known and understood and is more likely to buy additional products or consume more content which increase customer loyalty.

Increase sales, click trough rates, conversion: Recommendations typically speed up searches and make it easier for users to access content they're interested in, and surprise them with offers they would have never searched for.

Opportunities for promotion, persuasion: Companies are able to gain and retain customers by sending out emails with links to new offers that meet the recipients' interests, or suggestions of films and TV shows that suit their profiles.

Obtain more knowledge about customer: By knowing what a user wants, the company gains competitive advantage and the threat of losing a customer to a competitor decrease.

3. Area of use

An increasing number of online companies are utilizing recommendation systems to increase user interaction and enrich shopping potential. Use cases of recommendation systems have been expanding rapidly across many aspects of eCommerce and online media over the last 4-5 years.

Current recommendation engine use-cases at Amazon, Netflix, YouTube, Facebook and others

#Amazon

At Amazon.com, we use recommendation algorithms to personalize the online store for each customer. The store radically changes based on customer interests. When a customer clicks on the “your recommendations” the link leads to another page where recommendations may be filtered even further by subject area, product types, and ratings of previous products and purchases. According to McKinsey & Company, **35 percent** of Amazon.com 's revenue is generated by its recommendation engine.

For example, if Amazon observes that a large number of customers who buy the latest Apple Macbook also buy a USB-C-to USB Adapter, they can recommend the Adapter to a new user who has just added a Macbook to his cart. Due to the advances in recommender systems, users constantly expect good recommendations. They have a low threshold for services that are not able to make appropriate suggestions.

#Netflix

Netflix uses Recommender systems personalized diversity to generate Top Ten recommendations for user households, so that it can offer videos that each member of the household may be interested in. The company also focuses on awareness and promoting trust to help develop its personalized approach. According to McKinsey, **80 percent** of what users watch on Netflix come from product recommendations.

#YouTube

The YouTube online video community uses Recommender systems to create personalized recommendation so users can quickly and easily find videos that are relevant to their interests. Because of the value of keeping users engages, each user's activity on the site and to simultaneously highlight the wide range of available content. Recommendations now drive **70 percent** of overall "watch time" on YouTube, compared with **40 percent** in early 2014

#Facebook

For example, Facebook can monitor your interaction with various stories on your feed in order to learn what types of stories appeal to you. Sometimes, the recommender systems can make improvements based on the activities of a large number of people. Recommendations now drive **40 percent** of overall when compared with **20 percent** in early 2015

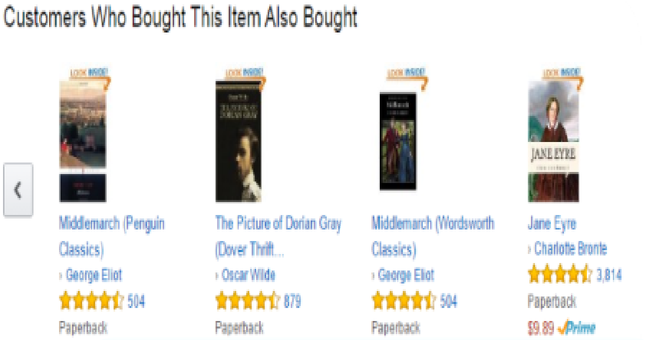
#Music Streaming app like: Wynk , ganna etc…

If a music streaming app is not able to predict and play music that the user likes, then the user will simply stop using it. This has led to a high emphasis by tech companies on improving their recommendation systems. However, the problem is more complex than it seems. Every user has different preferences and likes. In addition, even the taste of a single user can vary depending on a large number of factors, such as mood, season, or type of activity the user is doing. For example, the type of music one would like to hear while exercising differs greatly from the type of music he'd listen to when cooking dinner. Another issue that recommendation systems have to solve is the exploration vs exploitation problem. They must explore new domains to discover more about the user, while still making the most of what is already known about of the user.

4.Type of recommendations

{i}Personalized:

Personalized recommendation takes into consideration users' previous history for rating and predicting items.



Customers Who Bought This Item Also Bought

Middlemarch (Penguin Classics)
› George Eliot
504
Paperback
$8.7

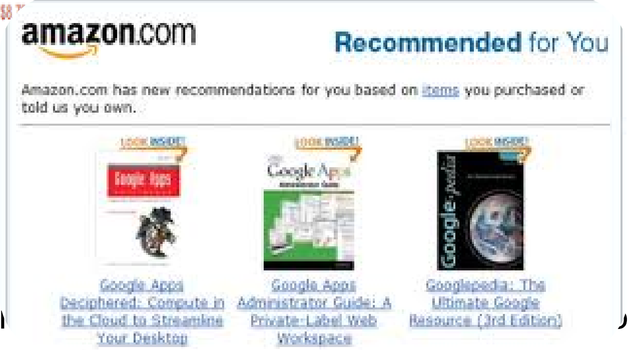
The Picture of Dorian Gray (Dover Thrift...
› Oscar Wilde
879
Paperback

Middlemarch (Wordsworth Classics)
› George Eliot
504
Paperback

Jane Eyre
› Charlotte Bronte
3,814
Paperback
$9.89 Prime

Example

- Amazon's recommendation mainly be achieved through “persona three modules: “recomme “recommended reason”.
- Recommended result contains overview of basic information p
- Rating result” is based on the r evaluate the overall quality of the book.
- Recommended reasons is concentrating on the inextricable link between the positive historical behaviour of the user and the book (most likely, past, tendencies of similar or similar books) to get the user's exact preference data., so readers can see the exact origin of their recommendations.



amazon.com Recommended for You

Amazon.com has new recommendations for you based on items you purchased or told us you own.

Google Apps Deciphered: Compute in the Cloud to Streamline Your Desktop

Google Apps Administrator Guide: A Private-Label Web Workspace

Googlepedia: The Ultimate Google Resource (3rd Edition)

{ii}Non-Personalized:

Non-personalized recommendation systems recommend what is popular and relevant to all the users which can be a list of top-10 items for every new user.

Example

- Amazon's “related recommended list” feature is also very powerful. When users buy goods by way of several other commodities tell the user to other users in the purchase of this product will be purchased in order to achieve the purpose of marketing package.
- Other customers purchased this product also often buy” and “containing viewed Other products of this user frequently purchased” constitutes a relevant recommendation list.

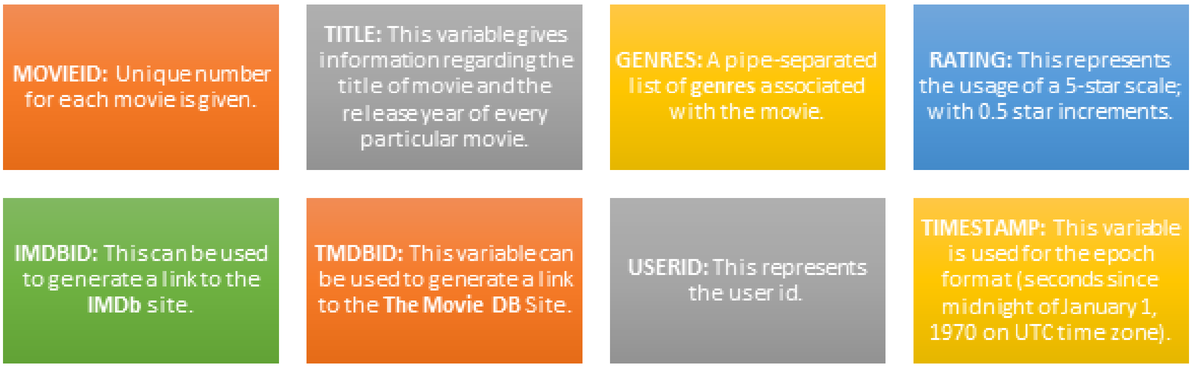
5.Exploratory Data Analysis

1.Load the data which are in csv file format.
We use **pandas** to read the files

2. Modify the rating timestamp: from universal seconds to datetime year.
3. Check for NaN values. Clean (delete rows) if % of NaN values is small.
4. Categorize genres properly: split strings into boolean columns per genre.

Data:

Movie data consists of 610 users, 9742 movies(movieId,Title,Genres)
Rating data consists of 100004 ratings (UserId, MovieId, Rating, Timestamp)



MOVIEID: Unique number for each movie is given.

TITLE: This variable gives information regarding the title of movie and the release year of every particular movie.

GENRES: A pipe-separated list of genres associated with the movie.

RATING: This represents the usage of a 5-star scale; with 0.5 star increments.

IMDBID: This can be used to generate a link to the **IMDb** site.

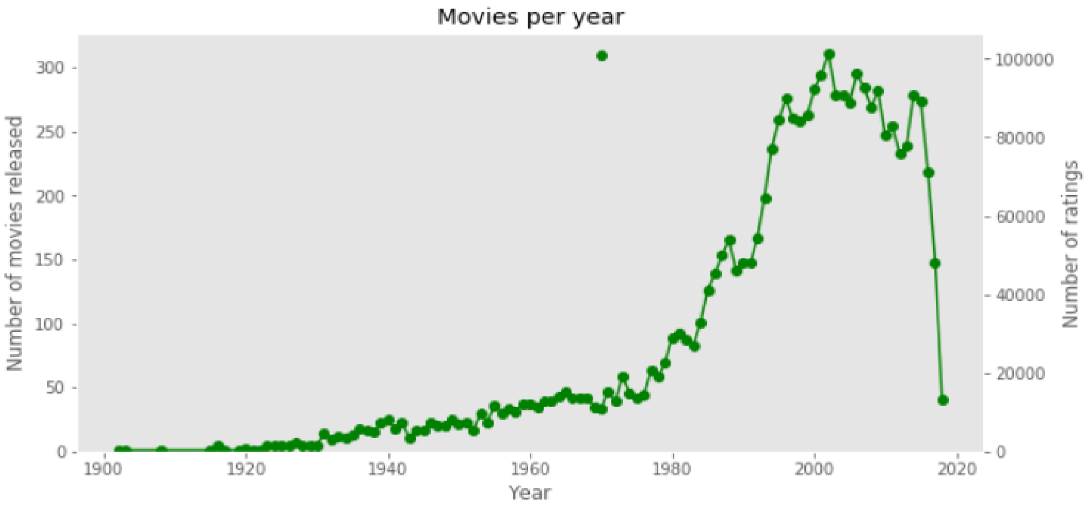
TMDBID: This variable can be used to generate a link to the **The Movie DB** Site.

USERID: This represents the user id.

TIMESTAMP: This variable is used for the epoch format (seconds since midnight of January 1, 1970 on UTC time zone).

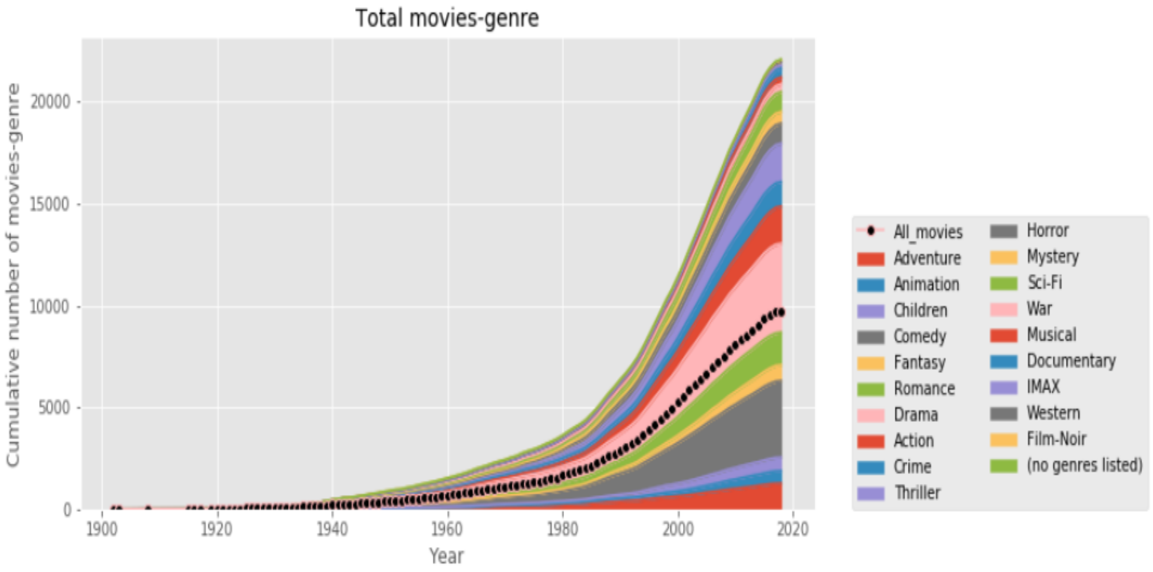
1.Number of movies and ratings per year.

Number of movies released per year increasing almost exponentially until 2000, then flattening and dropping in 2020

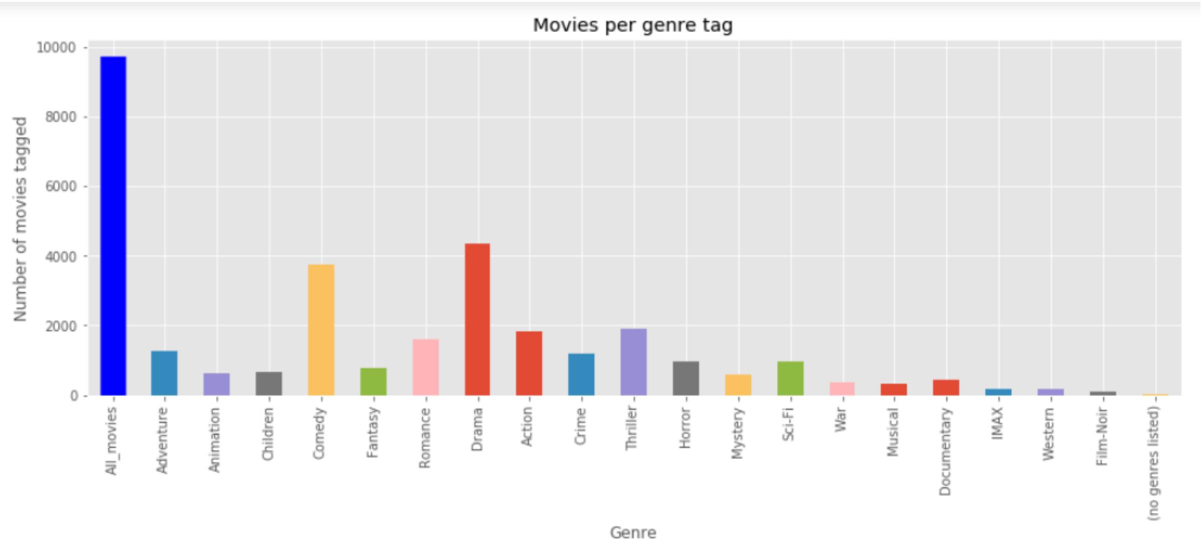


2.Cumulative number of movies, in total and per genre.

On average, movies are categorized into 2 genres .Comedy and Drama are the top genres used.

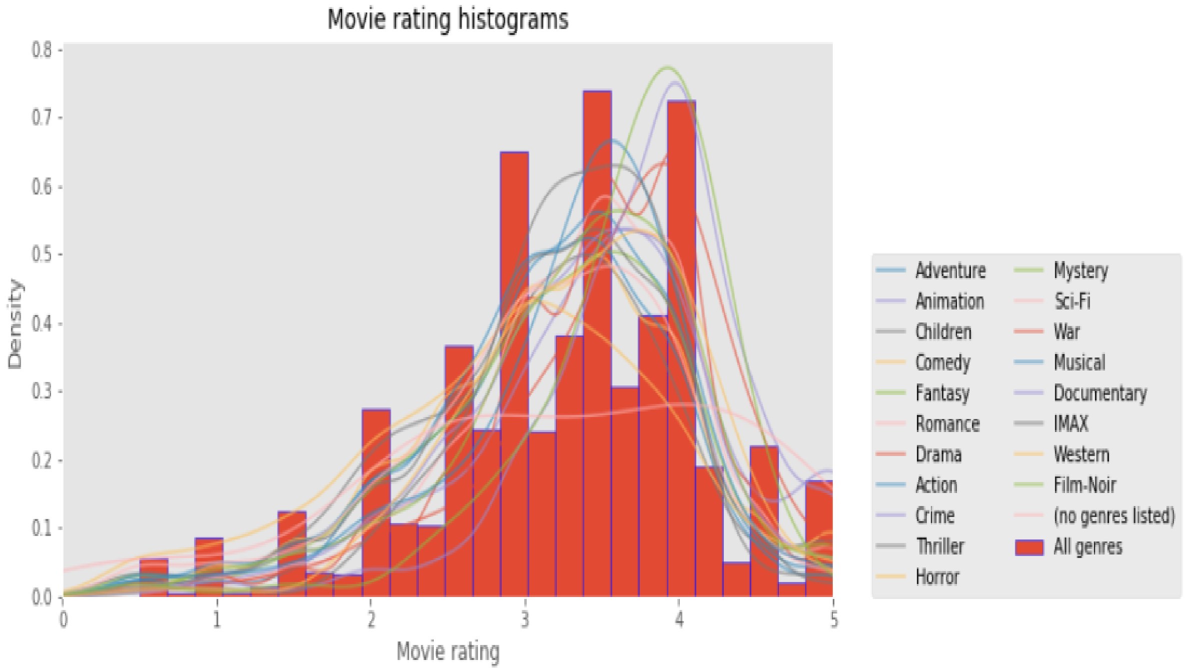


3.Movie per genre tag: Here we have more drama and comedy movies

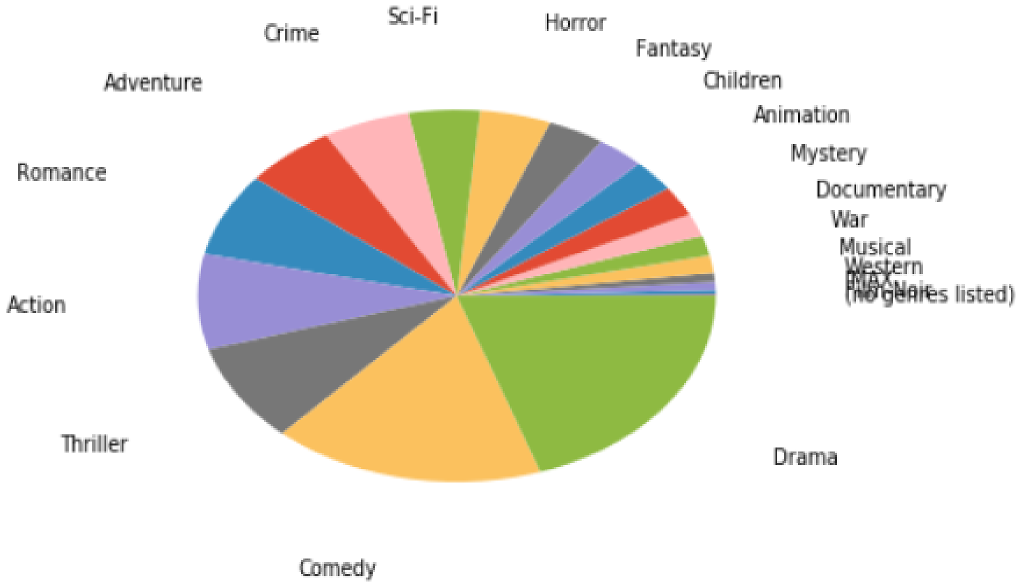


4.Distributions by genre, on top of total rating distribution.

This will help identifying consitent ratings or outliers (e.g. Comedies being rated higher in general). All genres show a similar pattern except Horror movies which are a bit skewed to the left. Movies without a tagged genre (no-genres listed) are also outliers, but likely due to the low number of ocurrences.



5.Quick view of pie-chart and word cloud of dominating geners

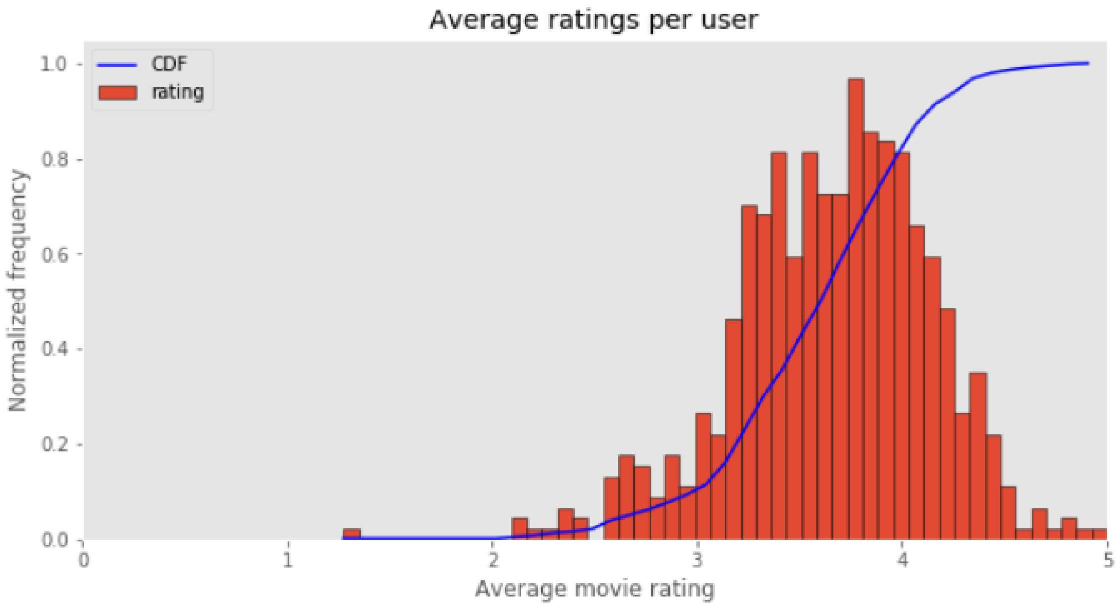


And also a word cloud representing this

Drama
Comedy
Thriller
Action
Romance
Adventure
Crime
Horror
Sci-Fi
Animation
Children
Fantasy
Mystery
Documentary
War
IMAX
Musical
Western
Film-Noir

6. Average ratings per user.

Users have a positive bias in general, with roughly 95% of their average ratings above the mid-point of 2.5.This is to be expected, and could have many explanations: users actually watch the better movies due to available ratings (and this should get better over time, as the rating system expands); users don't bother that much to rate bad movies as they do with the good ones.



6.Machine learning Algorithms

Recommendations can be generated by a wide range of algorithms. these techniques are used to predict ratings and opinions in which a user might have a propensity to express.

The three approaches through which recommendation system are designed:

1.Collaborative Filtering.

2.Content-based Filtering.

Techniques which selectively make use of both approaches (collaborative and content-based filtering) are called Hybrid recommendation systems.

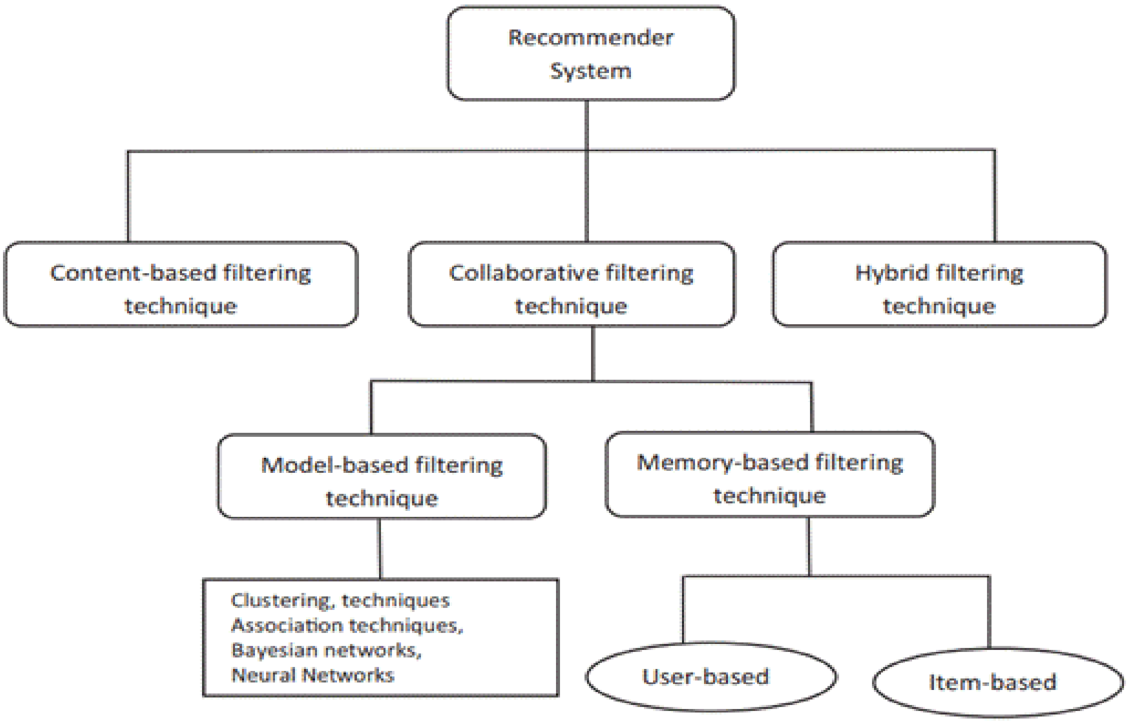

Recommender System
Content-based filtering technique
Collaborative filtering technique
Hybrid filtering technique
Model-based filtering technique
Memory-based filtering technique
Clustering, techniques
Association techniques,
Bayesian networks,
Neural Networks
User-based
Item-based

Figure: Types of recommendation systems

7.Collaborative Filter methods

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviours, activities or preferences and predicting what users will like based on their similarity to other users. 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.

Assume there are m users and n items, we use a matrix with size m*n to denote the past behaviour of users. Each cell in the matrix represents the associated opinion that a user holds. For instance, M {i, j} denotes how user { i } likes item { j }. Such matrix is called utility matrix. CF is like filling the blank (cell) in the utility matrix that a user has not seen/rated before based on the similarity between users or items.

There are two types of opinions, explicit opinion and implicit opinion. The former one directly shows how a user rates that item (think of it as rating an app or a movie), while the latter one only serves as a proxy which provides how a user likes an item (e.g. number of likes, clicks, visits). Explicit opinion is straight forward than the implicit one as we do not need to guess what do that number implies.

User-based Collaborative Filtering

Here we find look alike customers and offer products which first customer's look alike has chosen in past. For this we need to compute the similarity between users in user-based CF. To calculate the cosine similarity.

Let u_{i, k} denotes the similarity between user i and user k and v_{i, j} denotes the rating that user i gives to item j with v_{i, j} = ? if the user has not rated that item. These two methods can be expressed as the followings:

$$u_{ik} = \frac{\sum_j (v_{ij} - v_i)(v_{kj} - v_k)}{\sqrt{\sum_j (v_{ij} - v_i)^2 \sum_j (v_{kj} - v_k)^2}}$$

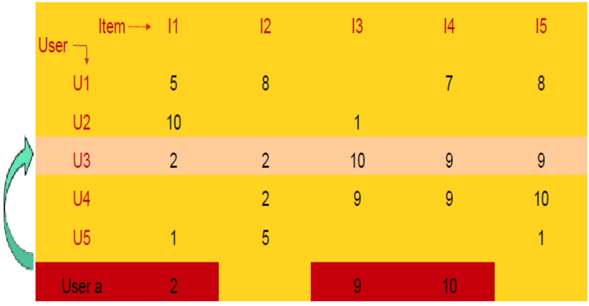
Cosine similarity:

$$\cos(u_i, u_j) = \frac{\sum_{k=1}^{m} v_{ik} v_{jk}}{\sqrt{\sum_{k=1}^{m} v_{ik}^2 \sum_{k=1}^{m} v_{jk}^2}}$$

Now, we can predict the users' opinion on the unrated items with the below equation:

$$v_{ij}^* = K \sum_{v_{kj} \neq ?} u_{jk} v_{kj}$$

In the given matrices (in ppt), each row represents a user, while the columns correspond to different items, each cell represents the rating. Here we can see user3 and user5 are highly correlated user. Hence we can recommend user3 on basis of user 5.



Item → User ↓	I1	I2	I3	I4	I5
U1	5	8		7	8
U2	10		1		
U3	2	2	10	9	9
U4		2	9	9	10
U5	1	5			1
User a	2		9	10	

Recommend items preferred by highly correlated user U3 → I5

Advantage:

- No knowledge about item features needed

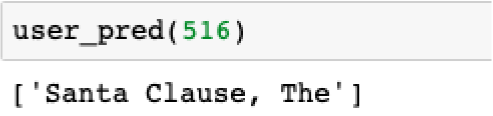
Problems:

- New user cold start problem

- New item cold start problem: items with few rating cannot easily be recommended

- Sparsity problem: If there are many items to be recommended, user/rating matrix is sparse and it hard to find the users who have rated the same item.

- Popularity Bias: Tend to recommend only popular items.



```
user_pred(516)
```

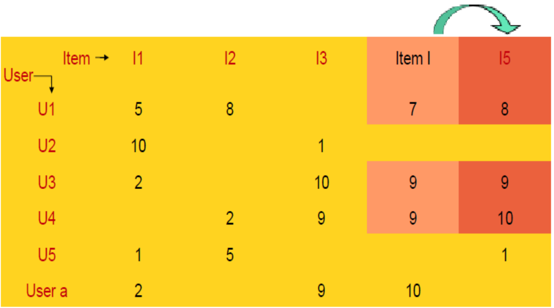
```
['Santa Clause, The']
```

Item-Item Collaborative filtering

Here we find look alike item. we can recommend alike items to customer who have purchased any item from the store. Instead of measuring the similarity between users, the item-based CF recommends items based on their similarity with the items that the target user rated.

similarities between pair of items are computed using cosine similarity metric.

In the given matrices (in figure), each row represents a user, while the columns correspond to different items, each cell represents the rating. Here 4th column is the targeted item. Here we can see the column item5 highly correlated to preferred items



User ↓ / Item →	I1	I2	I3	Item I	I5
U1	5	8		7	8
U2	10		1		
U3	2		10	9	9
U4		2	9	9	10
U5	1	5			1
User a	2		9	10	

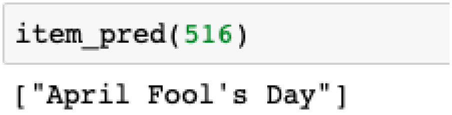
Recommend items highly correlated to preferred items → I5

Advantages:

- No knowledge about item features needed
- Better scalability, because correlations between limited number of items instead of very large number of users
- Reduced sparsity problem

Problems:

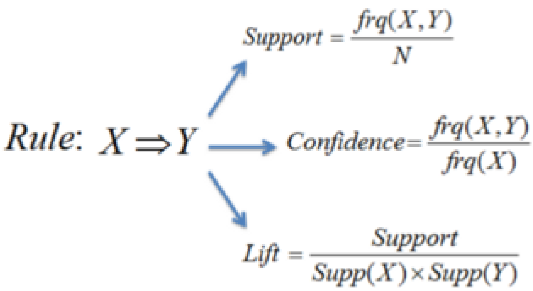
- New user cold start problem
- New item cold start problem



```
item_pred(516)
```

```
["April Fool's Day"]
```

8.Market Basket Analysis



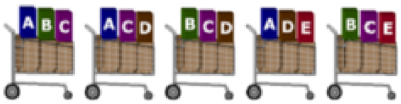
$$Rule:\ X \Rightarrow Y$$

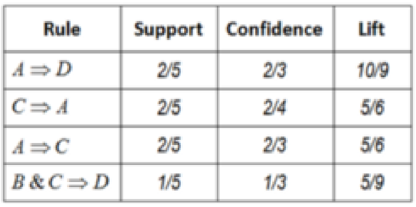
$$Support = \frac{frq(X,Y)}{N}$$

$$Confidence = \frac{frq(X,Y)}{frq(X)}$$

$$Lift = \frac{Support}{Supp(X) \times Supp(Y)}$$

Example:





Rule	Support	Confidence	Lift
$A \Rightarrow D$	2/5	2/3	10/9
$C \Rightarrow A$	2/5	2/4	5/6
$A \Rightarrow C$	2/5	2/3	5/6
$B \& C \Rightarrow D$	1/5	1/3	5/9

Introduction:

Market Basket Analysis is a type of frequent itemset mining which analyses customer buying habits by finding associations between the different items that customers place in their “shopping baskets”. The discovery of these associations can help retailers develop marketing strategies by gaining insight into which items are frequently purchased together by customers.

With Market Basket Analysis, the buying patterns of the customers are represented using **“Association Rules”.** The interestingness of a rule is measured using two metrics viz. **support** and **confidence.**

Example:

milk => bread [support = 2%, confidence = 60%]

A **support** of 2% for the above rule states that 2% of all the transaction under analysis show that milk and bread are purchased together.

support(A => B) = P(A U B)

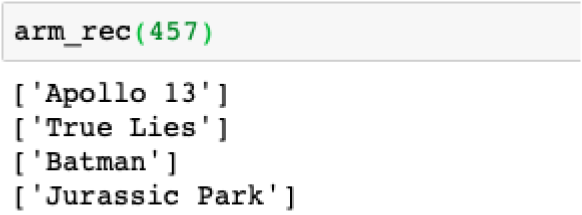
A **confidence** of 60% means that 60% of the customers who purchased milk also bought the bread.

Confidence(A => B) = P(B|A) = support(A U B) / support(A)

Typically, association rules are considered interesting if they satisfy both a **minimum support threshold** and a **minimum confidence threshold**.

We implemented the association rule mining for over data and the following are the results.

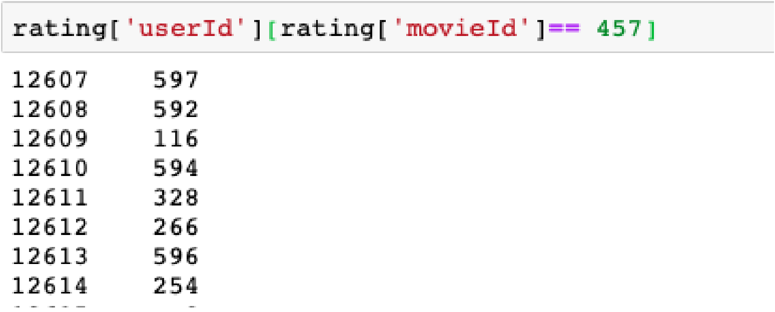
If the user sees a movie we can recommend based on the movie id from the association rules



```
arm_rec(457)
```

```
['Apollo 13']
['True Lies']
['Batman']
['Jurassic Park']
```

For movie id '457' the above are the recommended movies



```
rating['userId'][rating['movieId']== 457]
```

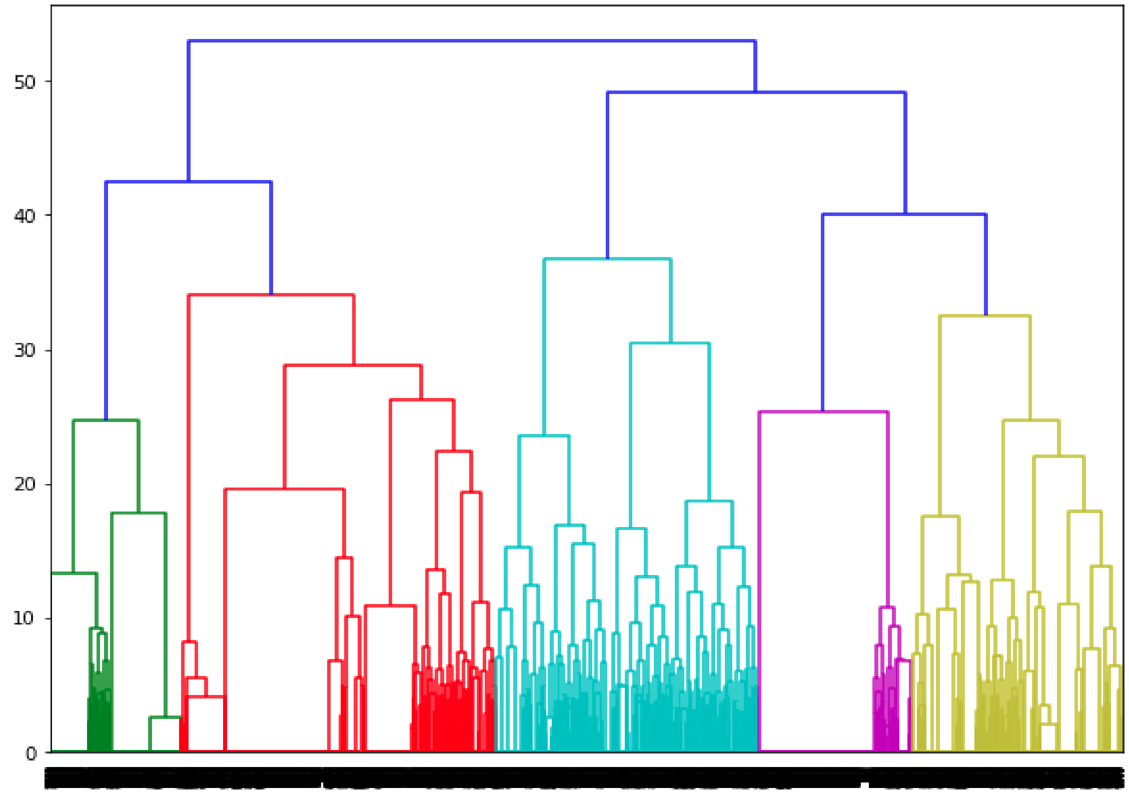
```
12607    597
12608    592
12609    116
12610    594
12611    328
12612    266
12613    596
12614    254
```

These movies can be recommended for all the people who watched the movie '457'.

9.Cluster Based Recommendation

Hierarchical clustering method :

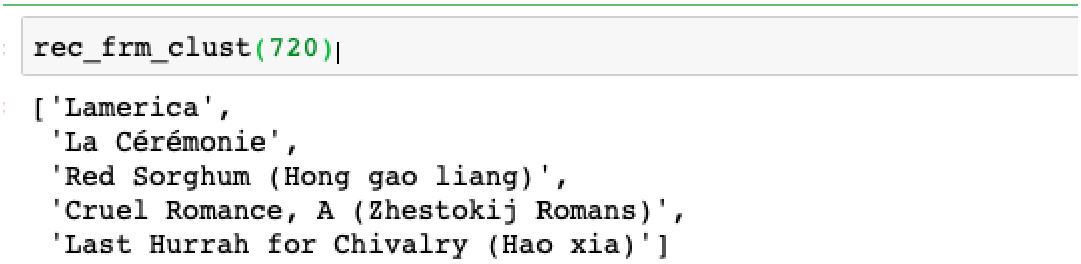
We used hierarchical clustering for genres in movies, movies were combination of 20 different genres like we a have a movie is a combined of different genres . Here we clustered the genres from movies and dendrogram is plotted.



By seeing this dendrogram, gave cut tree at 35 and extracted 6 clusters.

Now, we recommend movies based these clusters first give movie id and check it belongs to which cluster then extract movies from that cluster and recommend

But if user see movie old movie we cannot recommend a new movie for that reason we segregated movies based on the year and also we found mean ratings for movies and recommended top rated movies in that period of time.



```
rec_frm_clust(720)
```

```
['Lamerica',
 'La Cérémonie',
 'Red Sorghum (Hong gao liang)',
 'Cruel Romance, A (Zhestokij Romans)',
 'Last Hurrah for Chivalry (Hao xia)']
```

Challenges:

1. In collaborative filtering we find difficult in doing the data-matrix as we don’t have rating for some movies.
2. Cold-start :For a new movie or a new user because of lack of information.
3. It was tough extracting clusters using agglomerative clustering but the clusters turned out to be sequential.
4. In associative rule mining we are getting only few sets if we increase confidence.
5. We wanted to work with larger dataset but we found difficulties in creating data-matrix, clustering