Movie Recommendation System Using Machine Learning

Akhil Namboodiri
1814042
KJSCE
Information Technology

Jaykumar Panchal
1814044
KJSCE
Information Technology

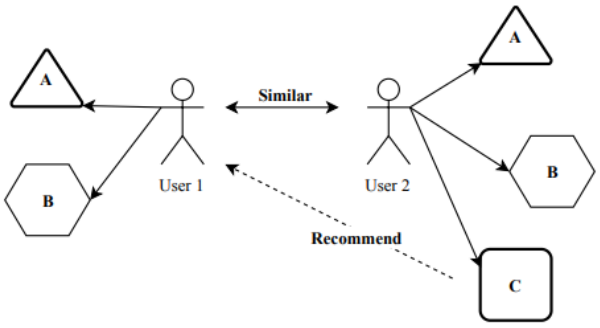
Pankti Nanavati
1814045
KJSCE
Information Technology

***Abstract*—The system that makes recommendations filters the data provided using multiple methodologies and recommends the most appropriate one for the customer's benefit. People use a powered recommendation system in many areas of their daily lives, such as movies, music, books and many other products. The capabilities of recommendation models are currently being relied on by the majority of internet-based businesses to increase product sales and even improve customer happiness. To recommend products, several content sites, such as YouTube, Netflix, and Amazon, use recommendation systems. As a result, numerous strategies for recommending desirable items to each user have been investigated.**

***Index Terms*—Machine learning, Recommendation system, Support vector machine, Naive Bayes, Content-based Filtering, Collaborative Filtering**

I. Introduction [1] [2]

In today's world, where human beings are always busy with some or the other activity, **recommendation systems** have become pervasive and important. Recommendation systems filter the preferences for the user from huge data and provide the relevant information only. It helps the user find the content which interests him/her the most based on a number of factors making the experience personalized to each user. Suggestion of the products on e-commerce sites such as Amazon, movies/serials on OTT platforms like Netflix are some real word examples where recommendation system is used. For example, a user might rate the movie on the scale of 1(disliked) to 5(liked), such ratings are stored in the database. Along with the user preferences, several factors like demographic details, product description, profile, browsing history, and so on help build up and identify well matched pairs between user and items using algorithms and predictive modelling.



II. What is a Recommendation System or Recommendation Engine? [1]

Recommender systems are software program that make recommendations to users based on a variety of parameters. These systems forecast the most likely product that users will buy and that they will be interested in.The recommender system works with a vast amount of data by filtering the most important information based on the data provided by the user and other criteria such as the user's preferences and interests. It determines the compatibility of the user and the object, as well as the similarities between users and items, in order to make recommendations.

III. Need for recommendation systems [1]

Recommendation systems are useful and beneficial to both the users and the providers.

- It minimizes the cost of browsing, finding and selecting the product from an online platform.
- It also helps to enhance the decision-making process thereby increasing the product sale and the revenue generated for the service provider.
- It also helps in saving people's time and effort in finding their choices in a complex domain easily. The users do not have the need to have in-depth knowledge of the things they are browsing or need. It is a information filtering system which solves the problem of information overload.

IV. Machine Learning algorithms used for Recommendation models

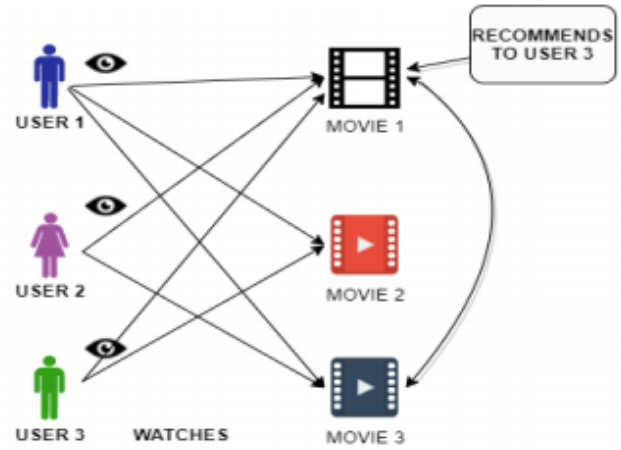
This section provides an overview of different machine learning methods that can be utilised to create effective recommendation systems. [1] [3]

- Support Vector Machine Algorithm
- Naive Bayes Classifier
- Collaborative Filtering
- Content Based Filtering
- Hybrid Filtering
- Clustering Approach like K-means algorithm, KNN
- Logistic Regression and many more ...

V. Machine Learning Algorithms for Movie Recommendation systems

A. Collaborative Filtering [1] [2] [4]

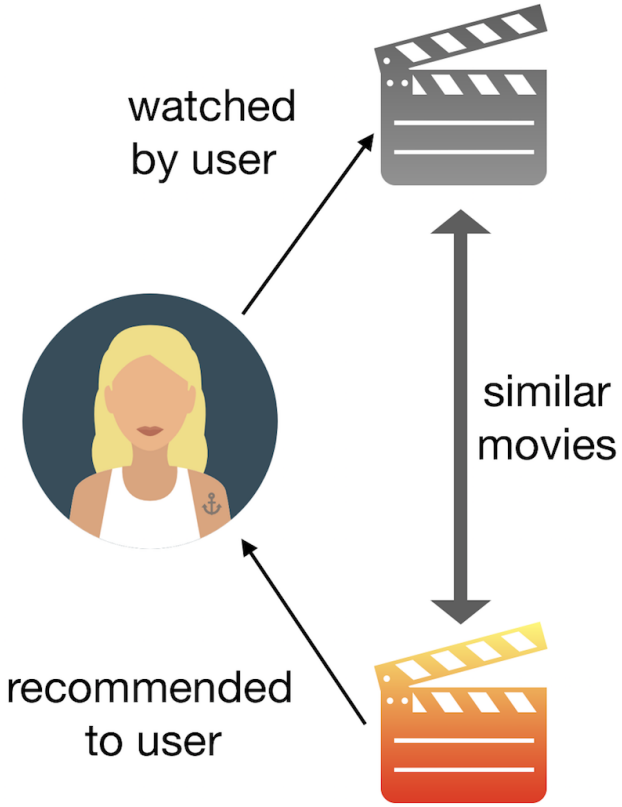
Collaborative filtering is a method wherein the attributes of each item are not taken into account. It uses the preferences provided by other users to provide recommendation. When the data which is available is huge and in adequate quantity, the results will be promising. There are two types of collaborative filtering: item-based filtering and user-based filtering. In item-based filtering item is taken into consideration while in user-based filtering user is taken into consideration.



There are various advantages of collaborative filtering. It does not require domain knowledge as opposed to content filtering where domain knowledge is required, this filtering method also helps the user discover and find new items and interests. Besides the advantages, there are few disadvantages like it needs to have enough information to start recommending i.e., cold start problem, it cannot handle new users and new items effectively and unpopular tastes are not supported strongly.Although there are a lot of algorithms, collaborative filtering is the most popular one used by the companies as it involves user's interactions more. Collaborative filtering can predict better because it analyses the user's browsing history and compares with other users and then suggests results.

B. Content Based Filtering [1] [2] [5]

To make recommendations, content-based frameworks analyse the depiction and features of an item, as well as the client's preferences. If a customer purchases an item, this algorithm will suggest another item that is similar to the one that was purchased. It takes into account user's previous ratings, comments, likes and dislikes of the various items. So, basically users' previous preferences are taken into account for recommending them the items. Such profile of preferences can be created by the user himself or can be created automatically. This profile is matched with the properties or attributes of each item, providing the score for each item.



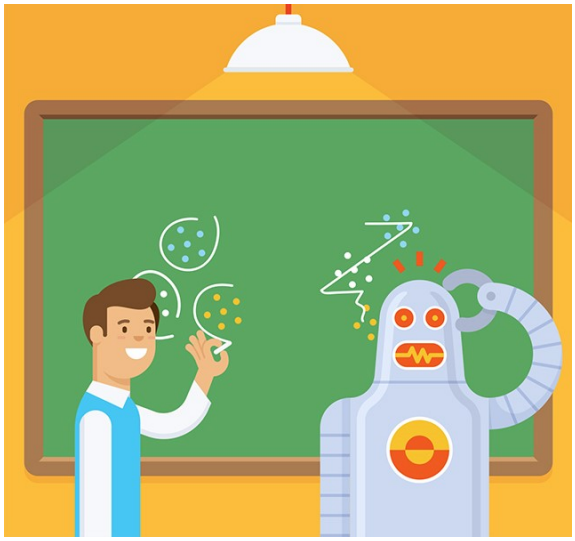
The key benefit of this method is that it does not necessitate the use of data from other users. Suitable and appropriate techniques should be employed for defining the user profile and the item attributes along with the algorithm which matches the profile with the attributes of the items in order to achieve the required accuracy. The content-based filtering method is specific to each user and hence can be applied to large number of users. It also suggests the new items which are not yet rated by other users. The drawbacks to this method are that the recommendations are limited to only existing interests of the user and this method requires a lot of domain knowledge since the attribute representation of the various items is hand-engineered.

C. K-Means Clustering Technique [3] [6]

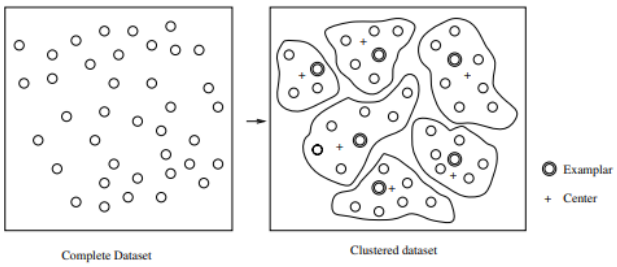
Consider the following scenario: we're developing a large recommendation system in which collaborative filtering and

content based filtering should last longer. Clustering is the initial suggestion. Clustering techniques have been dubbed the ”next best thing” to ”real” Artificial Intelligence in the sense of General AI, with many powerful applications.

It gives us a technique to characterise and categorise data when we don’t know how to do so beforehand. Different approaches have been taken to see which approach gives us the best result while clustering the users using clustering algorithms. The measure for generating recommendation will be on the basis of similarity of two items like vector distance between these items.



There are various advantages of clustering techniques used for recommendation systems. It is relatively simple to implement and efficiently scales well to large data sets. Besides the advantages, there are few disadvantages like scaling with number of dimensions. As the number of dimensions increases, a distance-based similarity measure converges to a constant value between any given examples. To overcome this, reduce dimensionality either by using PCA on the feature data, or by using “spectral clustering” to modify the clustering algorithm.



VI. Implementation and Results

A. Hybrid model of Collaborative Filtering and Content Based Filtering [1]

For the first implementation, it uses a hybrid combination of both the collaborative and content based filtering techniques for the movie recommendation system. The diagram below shows the flowchart for the implementation.

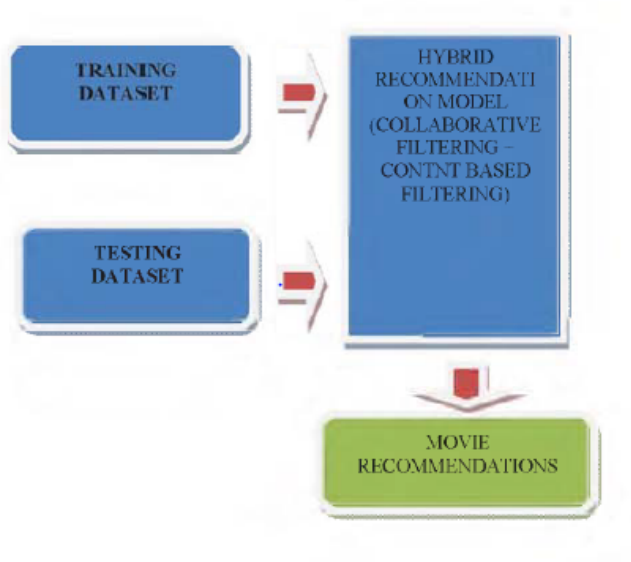


Fig. 1. Flowchart

The dataset used is the movielens dataset. The matplotlib library is used to visualise the data. The implementation uses a hybrid model of collaborative and content based filtering to make predictions. With the help of this method, the system can recommend users the movies that they may like. When it comes to the performance of this model, the hybrid model performs better than using either of the collaborative or content filtering techniques alone. The graph shown below shows how the hybrid performs better than using the two techniques independently. Performance is measured using precision (ratio of the true positive results to the sum of true positive and false positive results), recall (ratio of the true positive results to the sum of the true positive and false negative results), accuracy (ratio of number of predictions to the total number of predictions).

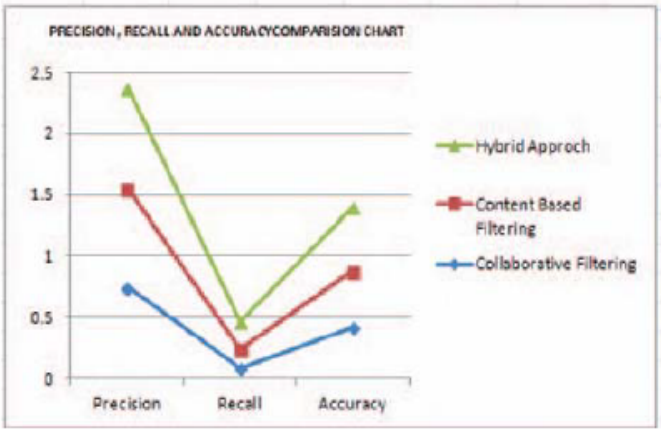


Fig. 2. Results

B. K-Means Clustering [3] [6]

The Movie recommendation system with the clustering technique uses K-means clustering and K-nearest neighbours. The implementation consists of data collection, data preparation, model creation, model training and model testing. In the data collection step, the right dataset is chosen. Here, the dataset chosen is the movielens dataset from Kaggle. The data preparation step includes data pre-processing. Also, a utility matrix is created which shows the movies rated by a user which is done by dividing the user data and movies data into separate dataframes.

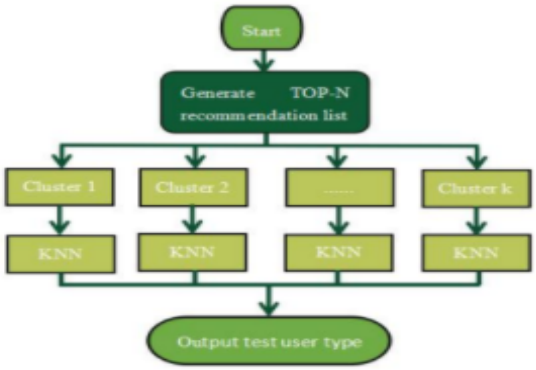


Fig. 3. Flowchart

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	5.000	3.000	4.000	3.000	3.000	5.000	4.000	1.000	5.000	3.000	2.000	5.000	5.000
1	4.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	2.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	4.000	0.000	0.000
4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	4.000	0.000	0.000	0.000	0.000	0.000	2.000	4.000	4.000	0.000	0.000	4.000	2.000
6	0.000	0.000	0.000	5.000	0.000	0.000	5.000	5.000	5.000	4.000	3.000	5.000	0.000
7	0.000	0.000	0.000	0.000	0.000	0.000	3.000	0.000	0.000	0.000	3.000	0.000	0.000
8	0.000	0.000	0.000	0.000	0.000	0.000	4.000	0.000	0.000	0.000	0.000	0.000	0.000
9	4.000	0.000	0.000	4.000	0.000	0.000	0.000	0.000	4.000	0.000	4.000	5.000	3.000
10	0.000	0.000	0.000	0.000	0.000	0.000	0.000	4.000	5.000	0.000	2.000	2.000	0.000
11	0.000	0.000	0.000	5.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
12	3.000	3.000	0.000	5.000	1.000	0.000	2.000	4.000	3.000	0.000	1.000	5.000	5.000
13	0.000	0.000	0.000	0.000	0.000	0.000	5.000	0.000	4.000	0.000	0.000	5.000	4.000
14	1.000	0.000	0.000	0.000	0.000	0.000	1.000	0.000	4.000	0.000	0.000	0.000	1.000
15	5.000	0.000	0.000	5.000	0.000	0.000	5.000	0.000	5.000	0.000	5.000	5.000	0.000
16	0.000	0.000	0.000	0.000	0.000	0.000	4.000	0.000	0.000	0.000	0.000	0.000	0.000
17	5.000	0.000	0.000	3.000	0.000	5.000	0.000	5.000	5.000	0.000	0.000	5.000	5.000
18	0.000	0.000	0.000	0.000	0.000	0.000	0.000	5.000	0.000	0.000	0.000	0.000	0.000

Fig. 4. Utility matrix

In the data creation step, the appropriate number of clusters i.e. k are selected for using the WCSS (Within-Cluster Sum of Square) method. After k is selected, the movie data is divided into clusters by using the K Means algorithm which creates the utility clustered matrix.

Now comes the data training step where the utility matrix is normalized and then the Pearson Correlation Matrix is used to calculate the similarity between users. After training, the model is tested on the testing data. Also, the model

	0
0	0
1	0
2	0
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	0
11	0
12	0
13	1
14	1
15	0
16	0
17	1
18	1

Fig. 5. Right number of clusters

	0	1
0	3.392	3.923
1	3.571	3.871
2	2.680	3.211
3	4.182	5.000
4	2.943	2.720
5	3.558	3.711
6	3.815	4.208
7	3.657	4.357
8	4.429	3.800
9	4.153	4.263
10	3.217	3.747
11	4.227	4.474
12	2.865	3.528
13	4.075	4.171
14	2.667	3.261
15	4.309	4.367
16	2.857	3.364
17	3.667	4.089
18	3.500	3.750

Fig. 6. Utility Clustered Matrix

is evaluated using some metric. Root Mean Squared Error (RMSE) will be the metric used here which is given by the formula shown in the diagram.

The results table below shows the RMSE obtained after using various number of clusters.

$$RMSD = \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}}$$

Fig. 7. RMSE formula

K-means + KNN	
Number of clusters	Root Mean Squared Error
19	2.504990
18	2.375555
17	2.337194
16	2.416212
15	2.256299
14	2.080751
13	1.994332
12	1.928682
11	1.861167
10	1.820095
9	1.625027
8	1.493939
7	1.441855
6	1.439451
5	1.269583
4	1.166091
3	1.141065
2	1.081648

Fig. 8. Final Result table

VII. Conclusion

The Paper starts by giving concise understanding of what are Recommendation systems in Machine Learning, its need, the various types of Machine Learning algorithms commonly used in Recommendation Systems. Then we went on to explain 3 types of algorithms used along with one of their applications which was in Movie Recommendation. For each algorithm, we briefed about its implementation involved for the various experiments carried on to demonstrate that specific algorithm. The flowchart and the final results from each implementation were carefully studied and mentioned in the paper.

References

[1] S. C. Mana and T. Sasipraba, "A machine learning based implementation of product and service recommendation models," in *2021 7th International Conference on Electrical Energy Systems (ICEES)*, 2021, pp. 543–547.

[2] N. Smitha, D. Anusha, C. Chaithanya, J. Sindhu, R. Tanuja, and H. S. Hemanth Kumar, "A review on movie recommendation system using machine learning," in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, 2021, pp. 769–773.

[3] R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbor," in *2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, 2019, pp. 263–268.

[4] M. Gupta, A. Thakkar, Aashish, V. Gupta, and D. P. S. Rathore, "Movie recommender system using collaborative filtering," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 2020, pp. 415–420.

[5] G. Sunandana, M. Reshma, Y. Pratyusha, M. Kommineni, and S. Gogulamudi, "Movie recommendation system using enhanced content-based filtering algorithm based on user demographic data," in *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, 2021, pp. 1–5.

[6] C. Cai and L. Wang, "Application of improved k-means k-nearest neighbor algorithm in the movie recommendation system," in *2020 13th International Symposium on Computational Intelligence and Design (ISCID)*, 2020, pp. 314–317.