# Comparative Study of Machine Learning Algorithms for Recommendation System

**D.O.I - 10.51201/12495**

**https://doi.org/10.51201/12495**

*1 Sonam Singh, 2kriti Srivastva

[1]D.J .Sanghvi College Of Engineering, Mumbai

[2]Department of Computer Engineering, D.J.Sanghvi College Of Engineering, Mumbai

***Abstract*—** *The role of recommender system is very vital in recent times for a lot of individuals. It helps in taking decisions without exploring physically. Broadly there are two types of recommender system: Content based and Collaborative Filtering. The first one focus on user's history and takes decisions. But there could be times when decisions based on only user history is not sufficient. For this, there is a need to analyze many parameters influencing the decision such as previous history, Age, gender, location etc. In the second approach it finds similar group of users based on several parameters and then takes decisions. Over the last few decades machine learning algorithms have proved their worth in this area because of their ability to learn from the given data and identify various hidden patterns. With this learning, these algorithms are able to generalize very well for unknown data. In this research work, a survey on three different machine learning based collaborative filtering methods are presented using Movie Lens dataset. The comparison of all three methods based on RMSE and MAE error is also discussed.*

Index Terms— Recommender system, root mean square error, matrix factorization, k nearest neighbor, single value decomposition.

## I. INTRODUCTION

Recommender systems have changed the way we interact with many services. Instead of providing static data, they bring an option to provide your feedback and to personalize the information which you have provided. It provides personalized informational flows independently for each user by considering feedback provided by user.

Recommender system are used to provide recommendations based on user's choice for a particular product which user might like. It is mainly divided into three main group known as content-based filtering, collaborative filtering and hybrid filtering method.

Content-based filtering is used to provide recommendations to user based on user's previous choice. Collaborative filtering is an approach in which a common group of people is determined who has the same choice for a particular product and based on the choice of one user in that group, recommendations are provided to other users. During filtering method two types of feedback are taken into considerations, user's previous choice related details are fetched based on explicit feedback and browsing related details are retrieved using implicit feedback. Hybrid filtering approach combination of content based and collaborative filtering.

The rest of the paper is organized as follows. Literature Review is explained in section 2. Dataset description is explained in section 3. Collaborative filtering approaches are presented in section 4. Comparison of all methods is discussed in section 5.Concluding remarks and future scope is given in section 6 &7.

## II. LITERATURE SURVEY

Davidson, James et al. [2] has mentioned how YouTube provide their recommendations using neighborhood collaborative filtering approach based on user's past history. Knn considers only last behavior of user and recommends the next item which are more similar to previous one where similarity is calculated using cosine similarity[3-4] .The author has explained that the shortcomings of K-nearest neighbor that it scales poorly when data increases as everyone is becoming dependent on internet for decision making [4].Recommender system focus on modelling the relationship between user and item based on historical feedback ,user's feedback can be implicit or explicit. Modelling implicit feedback can be challenging due to ambiguity of hidden data, Matrix Factorization (MF) methods can be used to uncover latent dimensions to represent user -item embedding's interactions through inner product[5-9].Netflix has built recommender

system based on single value decomposition (SVD) which has performed better than KNN.SVD++ is an extension of regular single value decomposition algorithm (SVD) which not only considers the implicit but also explicit feedback and performs better than SVD[10].Despite the success of traditional recommender system approach several limitations has been identified .Content based recommender system suffers with popularity biasness. Collaborative filtering approach KNN is not able to handle the data sparsity.SVD is able to deal with it but it don't consider the implicit feedback .SVD++ is able to deal with this problem but there is a need of deep learning based approach to build recommender system which can provide some quality recommendations

**III. Data Description**

Movie lens dataset has been used here which has 1 million information's with details of user, movie and rating. User details are user id, age, gender, occupation and zip-code. All demographic information is provided here where age lies between 1 to 56 .Movie details are also included here which have movie id, title and genre. All kinds of genre has been introduced in this dataset .it also consist of rating details which will have rating provided by user individual for specific movie.
In the subsequent subsection various methods of machine learning are discussed using this dataset.

**IV. Collaborative Filtering Approaches Using Machine Learning**

Machine learning algorithms are able to prove their worth in every area of research because of their ability to learn from given dataset and produce appropriate result .Recommender system has become an important part of every individual as it helps them to make a choice out of all available set of options. Recommender system is an Important part of machine learning algorithms that offers a relevant suggestions to user. Broadly there are two types of recommender system: content based and collaborative filtering .content based is not able to provide sufficient result as it only considers user's history for giving recommendations. To overcome this problem collaborative filtering approach is being used where it finds similar group of user for taking decisions. There are two ways to implement collaborative filtering approach:-

**1. Without Using Matrix factorization:-**

KNN is collaborative filtering based approach which don't use matrix factorization method for building recommender system. The main idea of this approach is to determine k-most similar neighbors in sample space which belongs to a particular product category then sample is taken into considerations with current sample. Let's consider a simple example to understand the use of knn ,Suppose there are two users Amar and Alit .Amar likes three movie A,B,C and alit likes C,B,D ,so based on their preferences we can say that they both same similar taste that means they both belong to similar group based on their choice. Since alit watches movie D recommender system can recommend it to Amar and alit watches A so accordingly that will be recommended to alit.

The idea of algorithm is to predict based on user's history record. KNN is used to find the neighbor of user u who have the same interest with target user, then recommend the things which neighbor of user u are interested in, predication is made based on the score calculation. The algorithm consists of three basic steps:-

i. User similarity calculation

ii. Nearest neighbor selection

iii. Predication score calculation.

In KNN algorithm, prediction is done based on similar group of user to the input user whose rating is to be predicted. Dataset is divided into 60-40 ratio for training and testing .In KNN algorithm results affects when we change the value of k .so following result shows that

| K value | RMSE | MAE |
|---|---|---|
| 10 | 0.9914 | 0.7818 |
| 30 | 0.9889 | 0.7811 |
| 50 | 0.9893 | 0.7847 |

Table 1 shows implementation result of KNN

KNN algorithm is simple to use but it has one drawback when size of dataset is getting increased its results gets affected because it's not able to handle data sparsity.
To overcome this problem new matrix factorization based approach has been introduced.

**2. With Matrix factorization:-**

Matrix factorization is a technique of factorizing matrix into two different category. It is a class of collaborative filtering in recommender system. It works by dividing matrix into two part user matrix and item matrix. And decomposing the user-item matrix interaction into the
Product of two lower dimensionality rectangle matrix. Here matrix m*n is decomposed into m*k and k*n .It is used to deal with linear data. It is used in determining the latent feature which are hidden and not known.

| | D1 | D2 | D3 | D4 |
|---|---|---|---|---|
| U1 | 5 | 3 | - | 1 |
| U2 | 4 | - | - | 1 |
| U3 | 1 | 1 | - | 5 |
| U4 | 1 | - | - | 4 |
| U5 | - | 1 | 5 | 4 |

Fig. 1. Example of matrix factorization

Assume we have 5 users and 10 items and their ratings which has a range of 1 to 5, as it has been given in the above figure. Now it's clearly visible in the above table that some user has not rated for a specific items .The main reason behind using matrix factorization approach is to solve the problem when there exist an latent feature through which we can determine the way user rates an item .For example if two user have given high ratings to certain movie if they have liked just actor and actresses of the movie or if the movie is an action movie, the genre preferred by both the users are same. Hence, if we are able to discover the latent feature, we can easily predict a rating with reference to a particular user and particular item.

- **Implementation of matrix factorization**

**SVD (single value decomposition)**

SVD is most popular matrix factorization technique which is used most widely. The model implementation is very straight forward .The mathematical representation of this model is as follows:-

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u$$

R^ui is predication calculation equation and μ is standard deviation and bu and bi are set to zero in first iteration .Here stochastic gradient descent are used for minimization. No of epochs are acting as a hyper parameter based on which the rmse (root mean square error) value gets affected.

Single value decomposition is model based collaborative approach where result depends on hyper parameter here epoch value is main parameter .Analysis has been done on algorithm by using different epochs to find the best result. Below table shows result of same

| n-epochs | RMSE | MAE |
|---|---|---|
| 20 | 0.9433 | 0.7448 |
| 40 | 0.9452 | 0.7458 |
| 60 | 0.9815 | 0.7709 |

Table 2 shows implementation result of SVD

Even though SVD performs really well with large set of data, it is not able to capture implicit data of user which can play an important role in user's selection of item. To overcome this problem new matrix factorization approach called SVD++ has been introduced.

**SVD++ (Enhanced single value decomposition)** SVD++ is an extension of SVD model which deals with both explicit feedback an implicit feedback which simply states that this model will not only take ratings given by user but also it will consider user's behavior's through implicit feedback at the time of generating result. The mathematical representation of this model is similar to SVD one, the only additional part is to consider implicit feedback. The prediction R^ui is set as:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T \left( p_u + |I_u|^{-\frac{1}{2}} \sum_{j \in I_u} y_j \right)$$

Where y is new set of j items which will capture implicit feedback. μ is standard deviation and bu and bi are set to zero in first iteration .here stochastic gradient descent are used for minimization. No of epochs are acting as a hyper parameter based on which the RMSE (root mean square error) value gets affected.

Single value decomposition is model based collaborative approach where result depends on hyper parameter here epoch value is main parameter .Analysis has been done on algorithm by using different epochs to find the best result. Below table shows result of same

| n-epochs | RMSE | MAE |
|---|---|---|
| 20 | 0.9213 | 0.7211 |
| 40 | 0.9314 | 0.7317 |

Table 3 shows implementation result of SVD++

Even though SVD++ performs really well with large set of data, it also capture implicit data of user which can play an important role in user's selection of item. But it is not able to capture user's temporal preference which helps in providing quality recommendations to user. To overcome this problem new deep learning based algorithm will be used in future.

## V. Comparison of all the methods

In this section a detailed comparison of various methods of recommender system is discussed the comparison is done based on error metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). Both this error is used to find correctness of predicted result by comparing it with actual value.

In the following subsequent subsection experimental setup and results has been discussed in detail

- **Experimental Setup:-**

Mainly train test split procedure is used to estimate the performance of algorithm when they are used to make prediction on data which is not used to train model .Normally random splitting has been used for analysis purpose where dataset is randomly divided into samples but here we have tested it based on both random and fixed splitting.


Fig 2.Experimental Setup

Here the algorithms are applied on movie lens (1million) dataset consist of 1 million ratings from 6000 users on 4000 movies.

a. User Rates movies, next pre-processing is performed to do data cleaning by replacing all NA values to zero.

b. A matrix with user, movies and the respective ratings is created.

c. Algorithms are applied to make predication of most preferable movie for targeted user by considering latent features .Based on target user's rating for movie is predicted by Comparing it with other similar user.

d. Here cross validation splits the initial dataset into training and testing with 60-40 criteria. Next evaluation metrics are used for performance evaluation of algorithms.

Figures shows execution of above mentioned algorithms on movie lens dataset

a. Root mean square error (RMSE)

Root mean square error (RMSE) is a prediction error which is used to find the difference between predicted and actual value .the lower rmse value is considered as a best fit and its range lies between 0 to 1000.

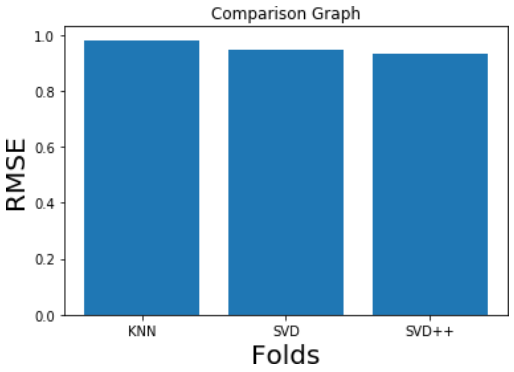


Fig .3.RMSE value of KNN, SVD, SVD++

Figure 3 illustrates the rmse value with respect to the folds which are generated through cross-validation.

Since values lies in the range of 1 to 1000, to get specified result we have scale it to the range of 0 to 1.

b. Mean absolute error(MAE)

It is averaged squared difference between actual and estimated value .it is basically used to capture how accurate model has given result .lower value considered as best fit.

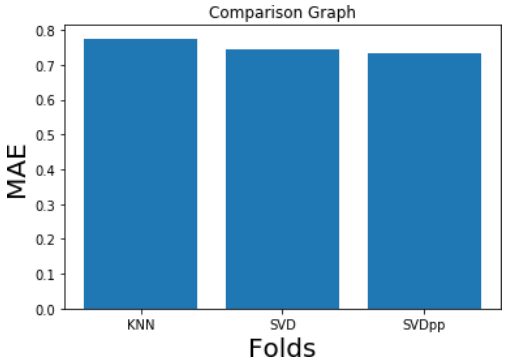


Fig .4.MAE value of KNN, SVD, SVD++

| Result | KNN | SVD | SVD++ |
|---|---|---|---|
| **RMSE** | 0.98 | 0.96 | 0.94 |
| **MAE** | 0.78 | 0.77 | 0.76 |

Table 4 shows the experimental result based on algorithms K-nearest neighbor(KNN),single value decomposition(SVD),Enhanced single value decomposition(SVD++). Based on output generated through different algorithm we can say that svd++ is performing better than existing two.

## VI. CONCLUSION

In this paper different recommendation Approaches of collaborative filtering has been discussed. It has been found that k-nearest neighbor cannot deal with sparsity so matrix factorization approach has been used to solve this issue. All the algorithms are explained in detail with proper steps involved in it .All the algorithms are compared based on Error rate and through experimental result we have found out that SVD++ performs better than other two algorithm.

## VII. FUTURE WORK

In this paper a comparative study of machine learning based algorithms of recommendation System has been discussed in detail .In the future, deep learning based recommender system can be built to enhance the performance and provide better recommendations to user.

## VIII. REFERENCES


*[1] I. Portugal,P. Alencar and D. Cowan,The use of machine learning algorithms in recommender systems:A systematic review, Expert Syst.Appl.,vol. 97,pp. 205–227,2018.*

*[2] Davidson, James et al. (2010). "The YouTube Video Recommendation System". In: Proceedings of the Fourth ACM Conference on Recommender Systems. RecSys '10. event-place: Barcelona, Spain. New York, NY, USA:ACM, pp. 293–296.*

*[3] X. Su and T. M. Khoshgoftaar, ''A survey of collaborative filtering techniques,'' Adv. Artif. Intell., vol. 2009, Aug. 2009, Art. no.421425.*

*[4] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing 7, 1 (2003), 76–80.*

*[5] Sarwar, Badrul, George Karypis, Joseph Konstan, and John Riedl (2000a). "Analysis of Recommendation Algorithms for e-Commerce", pp. 158– 167.*

*[6] M. J. Pazzani and D. Billsus, ''Content-based recommendation systems,'' in The Adaptive Web. Berlin, Germany: Springer, 2007,pp. 325–341.*

*[7] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in ICDM, 2008.*

*[8] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR:bayesian personalized ranking from implicit feedback," in UAI, 2009.*

*[9] F. Ricci, L. Rokach, B. Shapira, and P. Kantor, Recommender systems handbook. Springer US, 2011.*

*[10] Y. Koren and R. Bell, "Advances in collaborative filtering," in Recommender Systems Handbook. Springer, 2011.*

*[11] Adomavicius, G. and A. Tuzhilin (2005). "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions".In: IEEE Transactions on Knowledge and Data Engineering 17.6,pp. 734–749.*

*[12] Bell R.,Y. Koren and C. Volinsky (2009),"Matrix Factorization Techniques for Recommender Systems",In: Computer 42.8, pp. 30–37.*