# Movie Recommendation System

**Shalini Bhardwaj**
MT19045
IIIT Delhi
shalini19045@iiitd.ac.in

**Reecha Kumari Giri**
MT19134
IIIT Delhi
reecha19134@iiitd.ac.in

**Naman**
2017069
IIIT Delhi
naman17069@iiitd.ac.in

## 1 Problem Definition

While watching movies online on various platforms we get various recommendations as per various algorithms applied by websites.Websites and user both would be benefited by the quality of recommendation.If user will be engaged for more time while getting good recommendations website would have profit .Similarly user would also prefer the website as he can happily invest his time in movie watching as per his taste. Various techniques are used to recommend movies.Similar movie recommendation could be done by other user rating or reviews.

Review are also considered to do sentimental analysis to obtain actual user taste.As Sentiment analysis is usually used to extract the common feeling or emotion towards the product so that the organizations can understand whether the consumer's opinion is positive or negative. This can help them in improving their defects and change their strategies to get optimum results. By understanding why a particular product got positive reviews and what makes that product unique, any business or manufacturers can take advantage of this fact and create more products with similar features. However, it is tricky to explain to a machine the modulation, cultural variations, slang, and misspellings that are often found in online reviews. Thus, we have attempted to recommend the movie to the user by using both rating and reviews. As per his taste such that both user and organization are benefited.

## 2 Background

Collaborative filtering is a method of recommending products to customers by using their past behaviors or product ratings/reviews, as well as similar decision by other customers to predict which items might be appealing to the original customers. For an instance,assume two users Rachna and Sachi, who have very comparable tastes. If the ratings, which both have stated, are very similar, then their resemblance can be determined by the fundamental algorithm.In such cases, there is a high probability that the ratings where in just one of them has definite value, are also likely to be similar. This similarity can be used to make interpretations about partly stated values. Thus movie recommendation would be personalized for every user.

Two types of Collaborative filtering are:

(i) User based Collaborative Filtering : In this, we focus on similarity of user with community or other users and then recommend . It is seen if users had similar taste in past, they might have similar taste in future too .

(ii) Item based Collaborative Filtering : In this, instead of focusing similarity of user with community we focus on what item from all the options are more similar to what we know user enjoys. Collaborative Filtering is not a recently discovered technique.It is what all parameters and their combination we take into consideration while computing the similarity which makes it interesting field of research .In movie recommendation system it is usually seen that collaborative filtering is majorly done on the numeric rating or the content of movie like genre,actors,etc. We have used combination of textual review summary provided by user to movies and rating as the parameter . Motive of using reviews is to get actual sentiment of user as rating does not provide intense feeling. Hence the user reviews are also incorporated.

## 3 Dataset Used

The dataset chosen is the Amazon movie TV-5 core file, covering over 3,410,019 reviews. The 5 core implies that each video/movie has at least 5 ratings and each user has rated at least 5 videos/movies. This dataset includes asin value, helpfulness score e.g. overall rating (out of 5), reviewText, Review- Time, reviewerID, reviewerName, summary. The asin value for each movie is the actual ”id” of the movie or video on Amazon. For better visualization scrapping of movie names of corresponding asin is done.Major features given in dataset are used and analysed. For implementation 50,000 data of dataset is used. Dataset set link is present in poster.

## 4 Literature Review

Various methods are stated and tested for movie recommendation on the basis of numeric rating .It has been identified in studies that better recommendations are possible if review are also considered to get these emotions and feedback by users accurately. Reviews can be included by extracting the sentiment of the reviews.In the paper “Sentiment Analysis of Amazon Mobile Reviews” sentiment is analysed by using countvectorizer and tfidf vectorizer .We have further updated vector from countvectorizer by multiplying the polarity of different words. Logistic Regression is used in training model for analyzing the sentiment.This has helped us to differentiate the words like “good , “awesome” and “disappointing”. In our dataset summary of reviews was given.Hence we have used summarised reviews after cleaning and preprocessing.In sone studies KNN was used for recommendation.Hence various algorithms of Knn like ball,KD tree with different k value is done to obtain maximum accuracy. Recommendation is done based on both user based and item based collabrative filtering. In various studies RMSE is stated as the optimal metric to measure the recommendation.Hence RMSE is one of the metric used for evaluation.

## 5 Baseline

As mentioned previously, in collaborative filtering we do not need to know much about the products or the customers. We just need to know how many unique products and customers there as well as how the customers rated the products. This information is stored in a user-item table. The ratings matrix is another representation of the user-item table where each user and item have been assigned an unique integer value. The general rating matrix R is in R(nu*ni) where nu is the number of users and ni is the number of items. Let Ri,j entry corresponding to the ith row and jth column of the matrix R. Ri,j is then the rating by user i on item j. We format the matrix such that each row is a unique movie /video and each column is a unique reviewer. The value of each matrix will then be the rating that a reviewer.The entries that have missing values in the matrix will be field in with 0’s.

Evaluation:

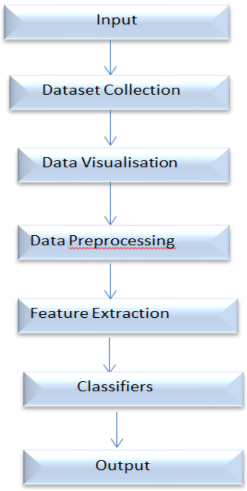
1.At first, naive approach is used in which missing value is filled with the mean rating of all users and on all products and then RMSE (root mean square error) has been calculated. RMSE= Root (sum (for all [‘overall’]-mean/)/Total ratings)

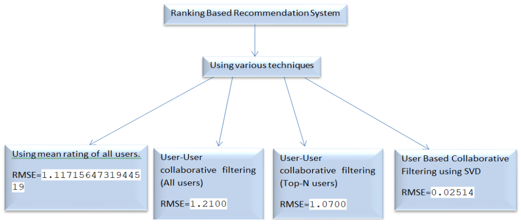
2.Then, to reduce the error user-user collaborative filtering is used which uses concept of cosine similarity. The cosine similarity between user u and user v is the normalized dot product of their row vectors, (ru and rv respectively) in the rating matrix R, To generate predictions for user u on product i from the cosine similarity function we then use a weighted average of some set of similar users (N) and those customers ratings of product i.We loop over all the users and all the products and treat each product for each user as a missing value and then predict its value. We then get the error between the predicted rating and the actual rating.

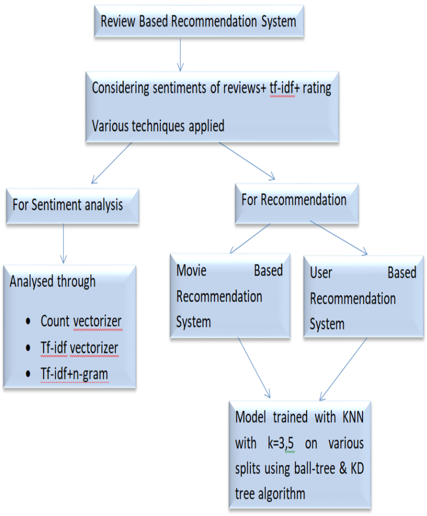
3.The next function performs user-user collaborative filtering just as above, but this time only uses the top N most similar customers to predict.the customer of interest’s missing rating. Further analysis is done on the reviews that is mentioned in proposed solution.

## 6 Proposed Solution

### 6.1 Overview of the Procedure




Ranking Based Recommendation System
Using various techniques
Using mean rating of all users.
RMSE=1.1171564731944519
User-User collaborative filtering (All users)
RMSE=1.2100
User-User collaborative filtering (Top-N users)
RMSE=1.0700
User Based Collaborative Filtering using SVD
RMSE=0.02514


Review Based Recommendation System
Considering sentiments of reviews+ tf-idf+ rating
Various techniques applied
For Sentiment analysis
Analysed through
Count vectorizer
Tf-idf vectorizer
Tf-idf+n-gram
For Recommendation
Movie Based Recommendation System
User Based Recommendation System
Model trained with KNN with k=3,5 on various splits using ball-tree & KD tree algorithm

## 6.2 Data Visualization

Data visualization is the graphical representation of information and data. Visual aids like charts, graphs, and maps help us visualize data by assisting us in understanding trends, outliers, and patterns in data. We have used Jupyter Notebook in Python in order to visualize our data.

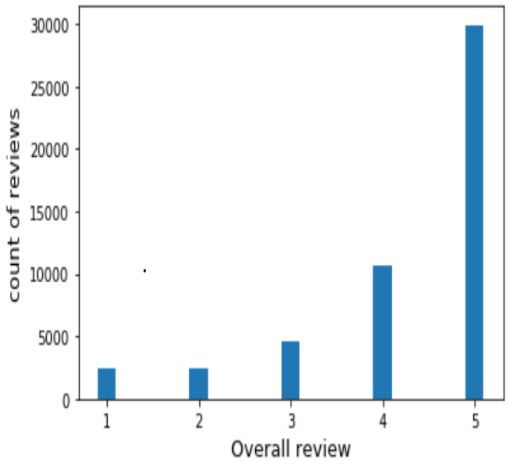


Figure 1: Graph showing total number of reviews for each type of rating

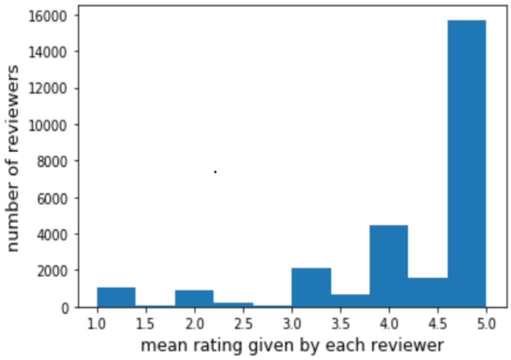
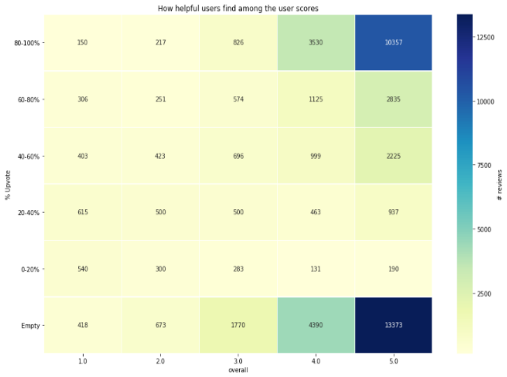


Figure 2: Graph showing mean rating given by each reviewer



| % Upvote | 1.0 | 2.0 | 3.0 | 4.0 | 5.0 |
|---|---|---|---|---|---|
| 80-100% | 150 | 217 | 826 | 3530 | 10357 |
| 60-80% | 306 | 251 | 574 | 1125 | 2835 |
| 40-60% | 403 | 423 | 696 | 999 | 2225 |
| 20-40% | 615 | 500 | 500 | 463 | 937 |
| 0-20% | 540 | 300 | 283 | 131 | 190 |
| Empty | 418 | 673 | 1770 | 4390 | 13373 |

Figure 3: Heat map showing total number of reviews getting upvote for each rating

## 6.3 Preprocessing

Data preprocessing can be thought as performing a function on raw data to prepare it for another processing procedure. It transforms the data into a simpler and effective format and also can be used for further processing. In our case, we have used various preprocessing techniques such as removal of HTML tags, non-characters such as digits and symbols, stop words such as “the” and “and”, and converted words to lower case. We have used stemming to convert certain words into their root words in order to maintain uniformity. For example, with stemming, a review which contained the word “watching” was converted to its root word “watch”.

## 6.4 Proposed Algorithms

In order to recommend products to the potential users, understanding textual review and quantitative review and also undermining the relationship between two distinct review methods is important. Recommending a movie to the audiences solely

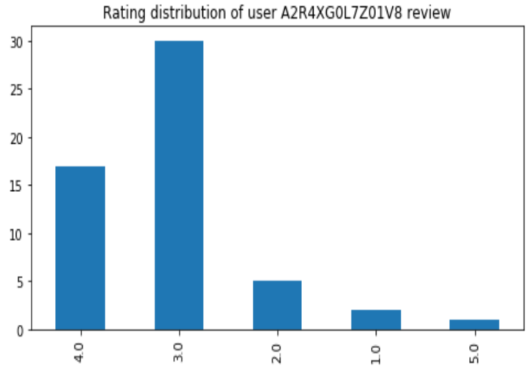


Figure 4: Graph showing rating distribution of user having ReviewerID: A2R4XG0L7Z01V8

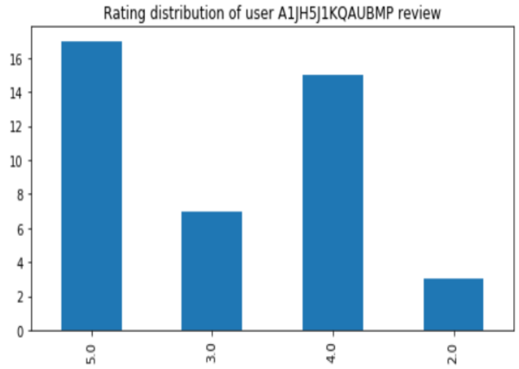


Figure 5: Graph showing rating distribution of user having ReviewerID: Aijh5jikquabmp

on the quantitative review could results in recommendation of an irrelevant movie. Recommending a movie to the audiences based on textual review can result in recommending a movie with similar quality but bad reputation. There are two goals of for the recommender system, first is to be able to effectively predict quantitative reviews based on qualitative reviews; second is to provide each audiences with relevant movies using textual mining; third is to combine the first and second step to allow for recommendation of similar movies, which are likely to be purchased and enjoyed by the audiences. We found that finding links between the quantitative and qualitative review and using the correlation between two review types are effective in finding a relevant and positive movies for the users. In order to do so, the techniques applied are:

### 6.4.1 Step 1: attempts to find similar movies based on rating given by users.

1. At first, naive approach is used in which missing value is filled with the mean rating of all users and on all products and then RMSE (root mean square error) has been calculated. RMSE= Root (sum (for all [‘overall’]-mean/)/ Total ratings) RMSE=1.2157

2.Then, to reduce the error user-user collaborative filtering is used which uses concept of cosine similarity To generate predictions for user u on product i from the cosine similarity function we then use a weighted average of some set of similar users (N) and those customers ratings of product i, we loop over all the users and all the products and treat each product for each user as a missing value and then predict its value. We then get the error between the predicted rating and the actual rating.RMSE= 1.2100

3.The next function performs user-user collaborative filtering just as above, but this time only uses the top N most similar customers to predict the customer of interest’s missing rating.RMSE=1.07.
4. SVD is used for predicting movies based on reviewerID taken as input. The advantage of SVD is that: users’ score matrix is a sparse matrix, so we can map the original data into a Low-dimensional space and then calculate the similarity of different items. This can help us reduce calculation complexity.

Note that for sparse matrices, we can use the sparse.linalg.svds() function to perform the decomposition.

### 6.4.2 Step 2: attempts to find similar movies based on rating given by users.

Movie Based Collaborative Filtering: Based on the input factors, sentiment analysis is performed on predicting the helpfulness of the reviews based on upvote percent. by using countvectorizer, tf-idfvectorizer and tf-idfvectorizer+n-gram Logistic Regression and SVM. Also, data visualisation for particular user is done to analyse the behaviour of ratings. Based on this, positive , negative and neutral sentiments of reviews are predicted and shown through word cloud. Like the word which is bigger in size has a high positive or negative score as compared to the word of smaller size shown below:

Further, countvectorizer is used to convert word to vector and sentiment score is also taken into account to find the other related movies. For prediction of rating based on reviews, K-nearest neighbour model (algorithm- ball tree and KD tree) is used on various parameters like k=3, k=5 on different splits.

the worst
film
horrible
movie
bad
this movie
boring
terrible
stupid
awful
waste of
worst movie
good
dvd

Figure 6: Showing sentiments of the words used for rating 1

bad
boring
great
quot
film
movie
good
disappointing
poor
this movie
really
review

Review Score two

Figure 7: Showing sentiments of the words used for rating 2

Input: Movie ID(asin)

Output: Similar movie IDs

#### 6.4.3 Step 3, compares quantitative review and qualitative review to create a connection between the two distinctive scales

#### 6.4.4 Step 4: User Based Collaborative Filtering

User Based Collaborative Filtering: analyze each user's qualitative review and recommend the movies using Step 1 and filter out the low ranking movies by incorporating Step 2 With these steps, the model is able to create a strong recommendation for the users by comparing the quantitative and qualitative review of the sample population with the specific user's reviews. Most of the steps used in movie based collaborative filtering are also followed here. It also uses KNN model to predict ratings based on reviews on various parameters to recommend the movie.

Input: User's id(Reviewer ID)

film
movie
good
story
fun
interesting
great
quot
decent
entertaining
action
okay
funny

Review Score three

Figure 8: Fig: Showing sentiments of the words used for rating 3

classic
blu ray
great
good
fun
film
movie
quot
excellent
good movie
the best
entertaining
thriller
story

Figure 9: Fig: Showing sentiments of the words used for rating 4

Output: Similar User's id and movie ids watched by him/her.

## 7 Results Produced

### 7.1 Results Based on rating

### 7.2 Results Based on Sentiment Analysis:

On the basis of rating, upvote percent. and number of reviews

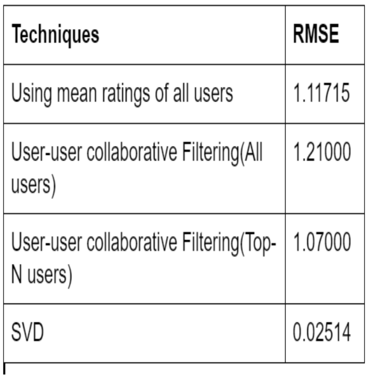
### 7.3 Results Based on Sentiment Analysis:

On the basis of the above table, we can conclude that prediction of sentiments through the count vectorizer model is more correct when we consider rating, upvote percent. and number of reviews.

Analysing the pattern of downvotes by users to predict upvotes of the products

### 7.4 Results Based on Sentiment Analysis:

On the basis of the above table, we can conclude that prediction of sentiments through the tf-idfvectorizer+n-gram model is more correct on



| Techniques | RMSE |
|---|---|
| Using mean ratings of all users | 1.11715 |
| User-user collaborative Filtering(All users) | 1.21000 |
| User-user collaborative Filtering(Top-N users) | 1.07000 |
| SVD | 0.02514 |

Figure 10: Fig:Results Based on rating

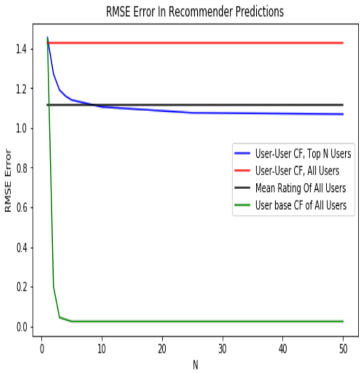


Figure 11: Comparison of techniques based on rmse values

analysing the pattern of downvotes by users to predict upvotes of the products.

Logistic Regression accuracy : 0.6453900709219859

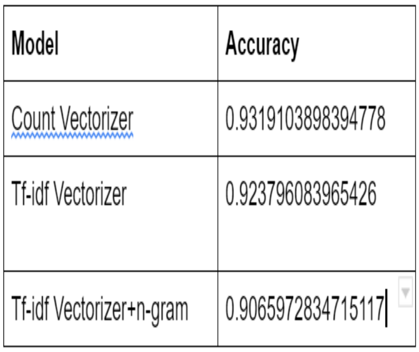
SVM accuracy: 0.5709219858156028

### 7.5 Result based on Movie Based Collaborative Filtering

It is observed that model trained with KNN=3 using ball-tree algorithm on train:test split=90:10 is giving highest accuracy with least rmse value. So, it can be concluded that the movies recommended by this trained model would be more correct.

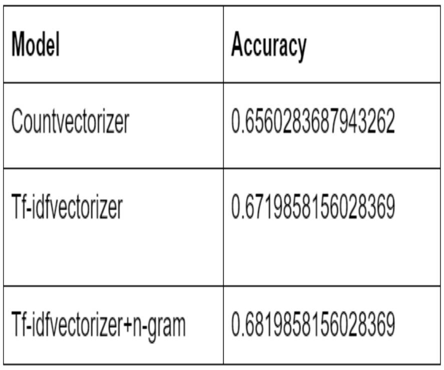
### 7.6 Result based on User Based Collaborative Filtering

It is observed that model trained with KNN=3 using ball-tree algorithm on train:test split=70:30 is giving highest accuracy with least rmse value. So, it can be concluded that the selection of similar users and the movies recommended by this trained model would be more correct.



| Model | Accuracy |
|---|---|
| Count Vectorizer | 0.9319103898394778 |
| Tf-idf Vectorizer | 0.923796083965426 |
| Tf-idf Vectorizer+n-gram | 0.9065972834715117 |

Figure 12: On the basis of rating, percent upvote and number of reviews



| Model | Accuracy |
|---|---|
| Countvectorizer | 0.6560283687943262 |
| Tf-idfvectorizer | 0.6719858156028369 |
| Tf-idfvectorizer+n-gram | 0.6819858156028369 |

Figure 13: Analysing the pattern of downvotes by users to predict upvotes of the products

## References

1.Alok Kumar A Collaborative filtering based sentiment analyzer to evaluate textual user feedbacks /opinions . 2003. *Computing Reviews*, 24(11):503–512.

2.MeenakshiArkav Banerjee 2020. Alternation. *Sentiment Analysis of Amazon Mobile Reviews*, 28(1):114–133.

3.Yeliz YENGİ 2016. *Distributed Recommender Systems with Sentiment Analysis*. Cambridge University Press, Cambridge, UK.