



International Journal of Research Publications

Personalized Recommendation of Movies Using a Combined Approach of Locality Sensitive Hashing, K-Nearest Neighbour and Collaborative Filtering.

U.M.R.C.M.Kumari^a, Thushari Silva^b

^aFirst affiliation, chathuri.uom@gmail.com, Faculty of Information Technology, University of Moratuwa, Sri Lanka

^bSecond affiliation, thusharip@uom.lk, Faculty of Information Technology, University of Moratuwa

Abstract

With the highly expanding usage of social media, a huge amount of data is being collected day by day. Proliferation of movies and their related views on social media has posed significant challenges in discovering the most suitable movies. Personalized recommenders that have been implemented to recommend movies use only a source of data and largely overlooked integration of several big data sources including tweets, you tube comments, posts on movies and comments. Moreover, most of the current approaches focus on one single aspect i.e. either content-based, collaborative filtering. Overlooking the integration of multiple aspects with many data provenances caused low effectiveness in current approaches. Hence, to overcome these deficiencies a novel, hybrid approach, KLC model which integrated user-based collaborative filtering, locality sensitive hashing and network-based approach is proposed. The New KLC model has been tested with 10000000 data items it outperforms benchmark methods.

© 2018 Published by IJRP.ORG. Selection and/or peer-review under responsibility of International Journal of Research Publications (IJRP.ORG)

Keywords: Collaborative Filtering; k-nearest neighbour classification; locality sensitive hashing; similarity metrics

1. Introduction

The recommendation systems are utilized in predicting interesting, likely items for a particular individual while using a large amount of information [1]. The Items can be news, music, books, movies, videos or any consumable thing. At present, the recommendation has become the most important in e-commerce and it is used by most popular e-commerce web sites such as amazon, e-bay, moviefinder Levis, CDnow.com etc. [2]. Mainly the recommendation can be classified as personalized and non-personalized systems. In movie recommendation for a certain user, as the first step the user behaviour should be taken into consideration. The user behaviour is analysed by observing the behaviours such as genres of previously watched movies, interactions or comments about the movies and mostly preferred etc. Another approach to find the direct rating information or history of the user. Collaborative filtering is an approach used to filter the items or movies [2]. First it gathers the ratings of movies gathered by users and it recommends movies for a particular user relied on the likely minded similarity and their tastes in the past. Some recommendations of movies are based on clustering. It is an unsupervised popular data mining technique which is used to partition a given dataset into homologous or similar groups relying on a similarity metric or else using a dissimilarity metric [3]. The widely used and most popular clustering technique is the k-means clustering. An integrated approach of user-based collaborative filtering, k nearest neighbor and locality sensitive hashing is used to predict the movies for a certain user which predicts more accuracy results because the approach is not based only to one recommendation approach.

The remaining of the paper is arranged as follows. Literature review describes the review on existing methodologies in Movie recommendation. Following the literature review reports the proposed method for personalized movie recommendation including the design, proposed models, algorithms and experimental results and analysis, benchmark analysis and finally the research offers concluding remarks.

2. Literature Review

The earliest implementation of the recommender system of collaborative filtering was the Tapestry [1]. The Collaborative Filtering term was firstly used by David Goldberg at Xerox PARC in 1992 [4]. It was a revolutionary mail and a repository system. This system was based on opinions of people like office work group. With the enhancement of the technology, rating-based recommendation systems are implemented. The GroupLens Researchers [8] have applied a collaborative filtering approach for Usenet news and movies. The collaborative filtering utilizes for finding the huge number of persons and searching a smaller set with similar tastes. This is performed by the comparison of each user with another user by calculating a similarity score. The similarity score can be computed using the Euclidean distance and the Pearson correlation.

Movie recommendation systems are mainly built on collaborative filtering approaches and clustering. In movie recommendation, the target user is able to rate items or movies that the user has seen already and those ratings are used to recommend the movies to the user that has not perceived based on similar ratings. Collaborative filtering [4] is tremendously spreading very fast while affecting to the other systems used in recommendation. Collaborative filtering is categorized mainly into two main classes namely memory based and model based. Memory based collaborative filtering searches for nearest neighbours from the user for an active user and recommend the movies dynamically. The disadvantages related to the above method are data sparsity and complexity in computation. The model based approach uses a prebuilt model to gather the patterns of ratings in the users which helps to scalability and data sparsity issues.

Different other technologies were also being used for recommender systems. Clustering models, Bayesian

networks and Horting techniques were included among them [7]. Bayesian networks create a model. It is based on a training set with a decision tree. The each node and edges represents the user information. The model based approach is minimal but accurate and fast as the nearest neighbour method. It is hard to customize for environments where user preferences are updated frequently. Clustering is worked on where the identification of similar users is done based on their similar preferences [3]. By forming the clusters, predictions or forecasts for a certain individual can be done using the information based on users where they are in the same cluster. In some cases, the clustering produces results less accurate for personalized recommendations than the other procedures. Moreover, the accuracy is worst that nearest neighbour algorithms. Horting can be describes as a graph-based technique in which the users are shown using nodes edges connected the nodes are provided the degree of similarity between two users. By using synthetic data, horting produces better predictions over nearest neighbour algorithms.

Another approach in movie recommendation is user-based collaborative filtering recommendation systems (CFRS) [2]. Items that are given similar ratings by various users are grouped by using the similarity measure of a clustering technique. Fuzzy C-means is a frequently used clustering techniques in this category. Among the algorithms used CFRS, TYCO algorithm [3] is significant. It comprises the opinion of typicality of the object from cognitive psychology. CFRS also follows K-means clustering technique to group users based on the user ratings by initializing centroids randomly. After repeating until convergence users are labelled based on the clusters. Finally softmax regression is used with hypothesis function to classify the users [5]. In traditional collaborative filtering techniques, work amount to be done is increased with the growth of number of participants. Hence for a large scale problems item-based techniques are used. Memory based algorithms use entire user-item matrix to give a prediction. The statistical techniques are used to find the users called neighbours [9].The movie prediction systems are developed using machine learning algorithms such as baseline-predictor, K-nearest Neighbour (KNN), Stochastic Gradient Descent and Support Vector Machine. In the baseline predictor [10], the estimation is done based on particular users. The unknown ratings are estimated by the below equation,

$$b_{ui} = \mu + b_u + b_i$$

Equation 1: The baseline predictor equation (1)

Where μ is the overall average score, b_u and b_i are training deviations which are estimated by a decoupling method which is lesser in complex but the accuracy is high [10]. Addressing the movie recommendation at large-scale the back-propagation classification/ranking method is used. This algorithm is based on weights given to input, hidden and output layers. During training process this model learns a prediction function $y(x)$, where x is a set of input values and y categories x to one predefined classes and computes the ranking value x using the weights across the model. The hidden layer has the responsibility for smooth tuning the weights which are assigned to various nodes of the BP through many iterations. Given each by a training instance I , the BP model updates the weights that are associated with the corresponding pairs of (input to hidden and hidden to output, respectively) nodes that are based on the difference between the actual $y(I)$ and desired output of I , called prediction error [11].

3. Proposed Approach

This paper proposes a hybrid and novel approach combining k-nearest neighbor, collaborative filtering and approaches in large scale social data analysis techniques such as locality sensitive hashing.

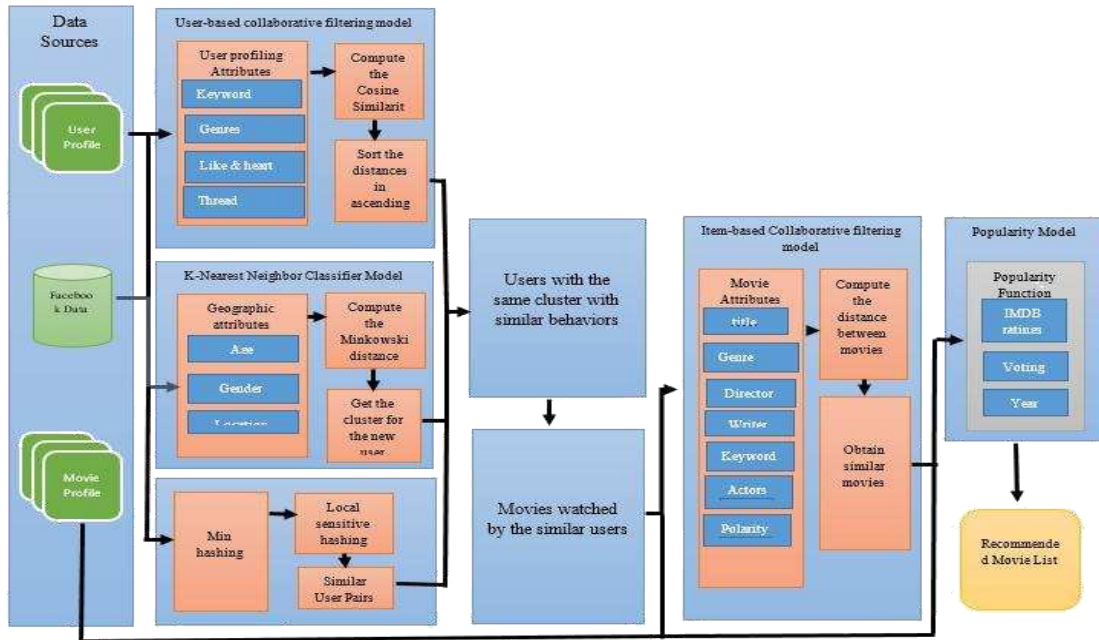


Fig.1. Combined Approach (KLC Model)

The data source is obtained by analysing the user behaviours and data of social media is accessed using Facebook API. User profiling module is consisted the demographic features and derived features. The derived features are obtained by analysing the user available information in Facebook social network and the generation of user profiling data is obtained in the user profiling module.

- i. Demographic Features: Age, Gender, Location.
- ii. Features from user profiling: keywords, movie category, average likes count, average hearts count, average review polarity, average thread polarity and cluster of the user.

Keywords are obtained by the comments that a particular user has posted on Facebook social network.

Movie category is derived from the movies that the user has commented on Facebook.

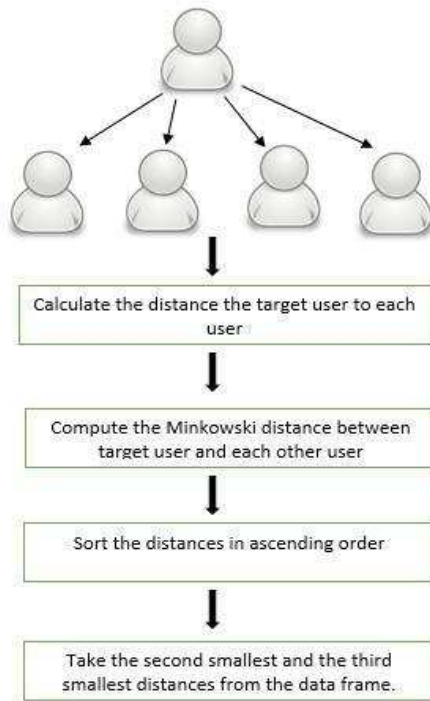


Fig.2. Searching for similar users by using demographic features of a new user

Average Likes count is the count of likes given by a particular user and it is taken as an average value.

Average Heart count is the count of heart given by a particular user and it is taken as an average value.

User demographic features such as age, gender and location are applied to find the closest users when registering to the system as a new user. The similarity of the demographic features are computed using the Minkowski distance function. For finding the closest users all the time the best distance function is the Minkowski because it is based on an absolute value and the absolute values give the robust outcomes than other metrics such as Euclidean. The Euclidean outcome values are influenced by abnormal values and it does not gives the always accurate results [5]. The equation for Minkowski distance function is given below.

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=0}^{n-1} |x_i - y_i|^p \right)^{1/p}$$

Equation 1: Minkowski Equation (2)

In this approach, let X and Y are represented by feature vectors $A = (x_1, x_2 \dots x_m)$ and $Y = (y_1, y_2 \dots y_m)$, where m is the dimensionality of the feature space. To compute the distance between X and Y , Minkowski distance [2] function is used. The pseudo code for searching similar users for a new user by demographic information is mentioned below.

Input Data $X = (x_1, x_2 \dots x_m)$

Normalize X

Input User Feature Vector $Y = (y_1, y_2 \dots y_m)$ for $I = 1$ to n do

 Compute MinkowskiDistance $d(X_i, y)$

end for

 Compute set I containing indices for k smallest

 distance $d(X_i, y)$

Return the lowest 2 distance vectors.

3.1 Analysing Facebook profile using K-Nearest Neighbour classification.

After registering into the system, user's profiles are analysed and the user profile is generated. Generating the user profile is not focused in this paper. To have a better personalized recommendation, the demographic features are not sufficient. Hence, more detailed features about the users should be utilized when doing recommendations. The K-nearest neighbour approach is done considering many user attributes. They are as keywords, movie category, average likes count, average hearts count, and average review polarity and average thread polarity. In the user profile's data, users are already clustered. By using the KNN classifier we can predict the cluster that a new user belongs. The clusters are formed in which the similar users are put into together. When predicting the cluster of the new user, it can be considered that users within the same cluster behaves in similar way.

3.2 User Based Collaborative Filtering Algorithms.

After getting the cluster of a certain user, it would be most effective to find the closest users within the cluster because the less similar users can be removed and the accuracy of the predicted movies will be increased when finding the most similar users within the cluster.

The user based collaborative filtering approach (UB-CF) is relied on the users and the recommended results are based on similarity of the user behaviours. The algorithm is mainly consisted of two steps.

Step 1: Define a user-user similarity matrix between various users and the targeted user.

In this step, key functionality is to calculate the similarity among two users. The features from the user profiling are used in order to define the matrix.

Here, the user's behaviours are considered to calculate the similarity. The mostly used distance function is used with the cosine function similarity to search for similar users. The Euclidean distance is the straight line distance between two points $A = (X_1, Y_1)$ and $B = (X_1, Y_1)$ is given by the formula

The above Euclidean distance is combined with the cosine similarity for a better recommendation. The cosine similarity score of two users is given by equation mentioned below. The angle between the two non-zero vectors (A and B) can be given as follows.

$$\frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|_2 \|\mathbf{B}\|_2} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Equation: Cosine Similarity (3)

After finding the similar users the second step is proceeded.

Step: 2 In the second step, the movies watched by the most similar users are obtained.

3.3 Item based Collaborative Filtering.

After getting the movies watched by the similar users, item based collaborative filtering is done to obtain the similar movies for those movies. This part is added to increase the movie variation when recommending movies for the user. So, user can have a high selection of movies from the recommended movie list. The features such as released Year, Genre, Director, Writer, actors, keywords and Polarity Confidence are used when finding similar movies. Cosine similarity is used to find the similarity between the movies. But it is good to recommend the most popular movies first than others. Therefore, the movie list is passed to a popularity model in order to get the most popular movies first.

3.4 The popularity Model for Movie list

In this approach most popular movies are shown. For its computation, the equation mentioned below is used based on movie popularity. Then the average probability of liking a movie is increased. For the movie popularity finding the using of ratings of movies will not give the basic results as the recommendations are filtered here. Hence, as a modified step we can pass the movie category of the user profile as an argument to retrieve only the best movies for a particular user. Mathematically it is represented as follows.

Rating Function (RF) =

$$\frac{V.R}{(V+M)} + \frac{M.C}{(V+M)}$$

Where,

V= Like count for the movie (4)

M= minimum likes required to be in the chart

R = Average Review Polarity of movie

C = Mean vote given by the users.

Combined recommendation system model based on the user and the item and data from social media is given below. As in that diagram, the movies are recommended for a certain user based on both collaborative, k-

nearest neighbour and the popularity model

3.5 Locality Sensitive Hashing For Finding Similar Users (LSH).

This algorithm is utilized in optimizing the similar user users in many dimensional space. In addition, when number of users are increasing as social media is generating more and more data. There should be an efficient and accurate way to respond with similar users without analysing all the rows in data set.

Minhashing is done to convert large-scale sets as short signatures while preserving the similarity. Locality sensitive hashing is focused on pairs of signatures that are likely to be similar. The big picture of Minhashing and LSH is shown below.

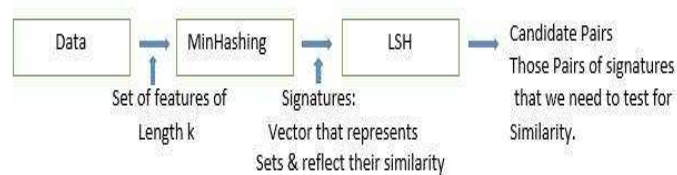


Fig.3. big picture of minhashing and LSH

The main steps in the algorithm is mentioned below.

Input: set of objects X

for $i = 1..m$

for each $x \in X$

stack k hash functions and form $x_i = (h_1(x), \dots, h_k(x))$

store x in bucket given by $f(x_i)$ On query time Input: query object q

$Z = \emptyset$

for $i = 1..m$

stack k hash functions and form $q_i = (h_1(q), \dots, h_k(q))$

$Z_i = \{ \text{objects found in bucket } f(q_i) \}$

$Z = Z \cup Z_i$

Output all $z \in Z$ such that $s(q, z) \geq s$

The personalized recommendations have been an effective tool for solving the information overflow problem, and improving the accuracy of recommendations have always been the aim to improve efficiency of the algorithm. By relying on the two classical recommendation algorithms and a popularity model this paper proposes an improved UB-CF and IB-CF recommendation algorithm, and the accuracy is higher than the single recommendation algorithm.

4. Evaluation

For the evaluation of Minkowski distance function, the Root Mean Squared Error (known as Mean Squared Error and Root Mean Squared, RMS) is applied. It is a measure which is used to calculate or measure the

deviation between the predicted values by the model and the environment observed. The RMSE finds out how similar on average of the given two lists denoted as d and p. The used equation for the calculation is given below.

$$RMS = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_i - p_i)^2}$$

Equation: Root Mean Squared Error (5)

This combined approach gives more trustable and accurate results because relying on one equation is not giving better results because according to situation each has consequence. K-nearest neighbour, user based collaborative filtering and item based collaborative filtering are combined to achieve the most accurate personalized recommendation movies for a user. For the K-nearest neighbour approach, high dimensionality of data is considered and the accuracy of classifier is checked with the increasing of the dimensionality. The below table (see Table 1) shows the results obtained when doing experiments with the k-nearest neighbour classifier.

Table 1. Accuracy of KNN vs number of user profile attributes

Number of User Data Attributes	KNN Accuracy
1	0.6
2	0.6
3	0.7
4	0.6
5	0.8
6	0.8

In addition, when there are a huge amount of data in data set, it is time consuming and reduces the response of the system, so the locality sensitive hashing is used to get the similar users with higher efficiency. The accuracy of K nearest neighbour classifier is tested with the k value is shown in below table (Table 2).

Table 2. Accuracy of KNN with k values

K value	Accuracy
1	0.754
2	0.809
3	0.809
4	0.792
5	0.774
6	0.789
7	0.703
8	0.663
9	0.683
10	0.698
11	0.698
12	0.683
13	0.683
14	0.683

15	0.683
16	0.658
17	0.642
18	0.609
19	0.622
20	0.622
21	0.622
22	0.605
23	0.562
24	0.505

In finding the optimal k value for the KNN classifier the below figure(Fig.3) shows that the maximum recall for the k value is between 1 and 10. The recall is value of the KNN classifier is plotted against the k-value.

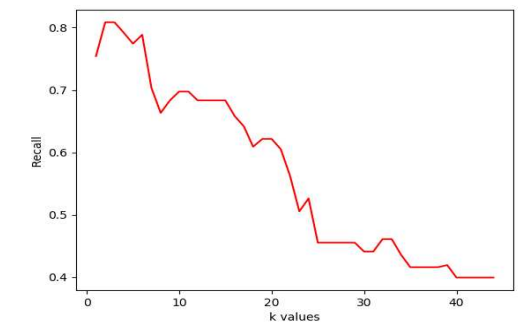


Fig.4. The recall of KNN classifier against k value

Acknowledgements

I would like to acknowledge my sincere gratitude to supervisor of Final Year Research Project, Dr.A.T.P.Silva, and Senior lecturer in Department of Computational Mathematics of Faculty of Information Technology University of Moratuwa, for her guidance, valuable suggestions and her time allocated for me which inspired me on this accomplishment.

References

- [1] David Goldberg, David Nichols, Brian M. Oki and Douglas Terry, Using collaborative filtering to weave an information Tapestry, Dec 1992 v35 n12 p61 (10).
- [2] User based Collaborative Filtering using fuzzy C-means Hamidreza Koohi, Kourosh Kiani.
- [3] A Fast Collaborative Filtering Approach for Web Personalized Recommendation System.
- [4] Programming Collective Intelligence by Toby Segaran.

- [5] Movie Recommendations from User Ratings by Hans Byström.
- [6] A Movie Recommender System: MOVREC
- [7] John S. Breese David Heckerman Carl Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, Microsoft Research Redmond, WA 98052-639.
- [8] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J. (1997). GroupLens: Applying Collaborative Filtering to Usenet News. Communications of the ACM, 40(3).
- [9] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, Item-Based Collaborative Filtering Recommendation Algorithms, Army HPC Research Center Department of Computer Science and Engineering University of Minnesota, Minneapolis, MN 55455.
- [10] Zhouxiao Bao, Haiying Xia, Movie Rating Estimation and Recommendation.
- [11] Yiu-Kai Ng, MovRec: a personalized movie recommendation system for children based on online movie features, Department of Computer Science, Brigham Young University, Provo, UT, USA
- [12] Rahul Katarya, Om Prakash Verma, An effective collaborative movie recommender system with cuckoo search, Department of Computer Science & Engineering, Delhi Technological University, Delhi, India
- [13] Ioannis Konstantas, Vassilios Stathopoulos, Joemon M Jose, On Social Networks and Collaborative Recommendation.
- [14] Yunnan Song, Shi Liu, Wei Ji, Research on Personalized Hybrid Recommendation System.

