Proceedings of IOE Graduate Conference, 2019-Summer
Peer Reviewed
Year: 2019 Month: May Volume: 6
ISSN: 2350-8914 (Online), 2350-8906 (Print)

Collaborative Filtering Recommender System Using Genetic Algorithm

Manoj Bhusal [a], Aman Shakya [b]

[a] *Computer System and Knowledge Engineering*
[b] *Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, Tribhuvan University, Nepal*
Corresponding Email: [a] manoj071mscs654@ioe.edu.np, [b] aman.shakya@ioe.edu.np

Abstract
Information overload in the internet has caused users to rely on Recommender Systems for information filtering. However the quality of the recommendations has always been a challenge. Many techniques have been developed for the improvement of quality and performance of recommender systems. Collaborative Filtering is the most used approach in recommender systems. This paper presents a technique that combines the idea of collaborative filtering with genetic algorithm. In this approach Genetic Algorithm is used to find the optimal similarity value between users. Each individual in the population represents the similarity matrix between the users. Thus, the proposed system does not directly compute any similarity metric but learns the similarity among users which helps to minimize the effect of sparsity and cold start problems common in collaborative filtering. A series of experiments have been conducted that demonstrate the effectiveness of the approach in terms of the quality of recommendations.

Keywords
Genetic Algorithm, Collaborative Filtering, Recommender System

1. Introduction

Information filtering system that involves suggesting options to users is termed as Recommender System (RS), for example, offering news to on-line newspaper readers or offering movies that the viewer might like. In this recent time RS plays an important role in the utilization of overloaded information from the internet. The suggestions provide relevant information from the collective knowledge of all other users in the domain. Recommender systems use a number of different technologies. Broadly we can classify these systems into three groups.

Content-based system is based on the properties of the items itself. These systems recommend items similar to those that a user liked in the past. For example, a person who watch cowboy movie gets the suggestion to watch the same genre movies.

Collaborative filtering whereas recommend items based on the similarity measures between the users or items. The items which are preferred by similar users are recommended to the user in the group with similar tastes. It is further divided into Item-based filtering and User based filtering. Item-based system usually recommends for user depending on the relationship between different items. Content-based system uses the relationship of different users to find the suggestion.

Hybrid approaches combine collaborative and content-based methods which helps to avoid certain limitations of content-based and collaborative system.

Despite all the achievements, the current generation of recommendation system still requires further improvements to make suggestion more effective and applicable to an even broader range of real-life applications. Meanwhile, the Recommendation Systems are having a hard time in generating an accurate and qualitative suggestions.

In collaborative filtering it is hard to calculate similarity values among either users or items. The problem arises during clustering users and items. Thus Genetic Algorithm can be used as a technique to calculate the similarity values between the users or items group so as to improve the recommender system. As similar users are clusters in a same group it is much more effective in the suggestion if the similar objects are grouped together. In this paper, a

user-user collaborative filtering is considered. The collaborative based recommender system is based on the similarity measure. Here, the similarity values are calculated using Genetic algorithm. Genetic algorithm is used to find the optimal similarity vector in this problem domain.So the objective of this paper is to develop an system which learn the similarity values between users using Genetic Algorithm.

2. Literature Review

In most cases GA is used in recommender system in order to find optimal similarity matrix, similarity function or optimal feature weight. Some of the work done by other researchers are discussed in this section.

Jesus Bobadilla et al. [1] proposed an approach to calculate similarity between users using genetic algorithm for improving collaborative filtering recommender system results and performance. They formulated a similarity function which is a simple linear combination of values and weights. Values are calculated for each pair of users between which the similarity is obtained, whilst weights are calculated once, making use of a prior stage in which genetic algorithm extracts weightings from the recommender system which depends on the specific nature of data.

Fong et al. [2] worked on a features based approach using Genetic Algorithm for hybrid modes of collaborative filtering in online recommenders and coded the input variables into genetic algorithm chromosomes in various modes.

Zhang et al. [3] introduced a genetic clustering algorithm to partition source data using similarity computation techniques to ameliorate scalability issue in collaborative filtering.

Xiao et al. [4] proposed an item-based collaborative filtering system which involved a novel similarity function using average rating for each user instead of the overall average rating for all users.

LV et al. [5] developed an item-based RS depending on the features of a recommended item with corresponding weights. GA was used to find the feature weight.

The proposed system in this paper helps to reduce the cold start and sparsity problems along with increasing the quality of recommendations by considering an individual chromosome as the entire similarity matrix among users. The similarity value among users thus would be generated using Genetic Algorithm.

3. Methodology

3.1 Overview

The utility matrix (user-item rating matrix) is represented by U_I (u, i), where u is the user and i is the item. The value of U_I (u, i) is between 0 and 5, where 0 means the user u has not rated the item i and 5 means the user u liked the item i to the fullest. The optimum similarity value between each user pair is obtained using genetic algorithm, the initial population being the random similarity values between all pairs. The active user who is targeted for the generation of recommendation uses the similarity value of top k most similar users obtained from genetic algorithm to calculate the prediction rating value of the unrated items. Top n high rated items are shown as recommendations for the active user.

Algorithm 1 Procedure of the system

1. Read user-Item rating matrix (utility matrix).
2. Generate the initial population that contains M individuals.
3. While(Current Generation ¡= Max Generation)
   (a) Predict rating value for utility matrix using each individual.
   (b) Calculate MAE for each individual.
   (c) Select top x best individual.
   (d) Apply Crossover with probability Cp.
   (e) Apply mutation with probability Mp.
   (f) Populate select individuals for new population.
4. Predict rating for each unrated item of active user using the optimum result from GA
5. Select the top K best items for the active user for recommendation

3.2 Genetic Representation

The genetic algorithm starts with an initial set of (random) solutions called a population. Each individual in the population is called a chromosome which represents a solution to the problem. Each individual is evaluated by its fitness function. The

chromosomes evolve through iterations called generations. Maximum number of iterations can be predefined or a stopping condition such as finding a desired fitness function can be assigned. One of the most important features of GA is representation of the problem with a string of symbols known as genes. The representation is usually formed by binary, real-valued or integer valued arrays. Binary encoding occurs by 1-0 strings and needs more memory space while computing. Integer encoding contains real values that represent the solution [6]. A Matrix representation is used in this research work. The common genetic operators (selection, crossover and mutation) are used in the experiment.

3.3 Encoding

Matrix representation has been applied to a single stage transportation problem by Vignaux and Michalewicz [7] for the first time. Two-dimensional structure is a natural representation for the recommendation problem to represent similarity between users, as shown in the example Table 1. In this work, the user-user similarity matrix is considered as a chromosome.

Table 1: Users similarity matrix

User	Ram	Shyam	Krishna	Bishnu
Ram	1	0.458	0.364	0.364
Shyam	0.458	1	0.833	0.644
Krishna	0.364	0.833	1	0.515
Bishnu	0.446	0.644	0.515	1

The population is encoded in a vector form. The lower or upper triangular matrix excluding the diagonal value is sufficient to extract the required information for the GA. The diagonal values are the self similarity values which is always 1. So the required values can be encoded in vector form as in Table 2.

Table 2: Chromosome encoding

SN	User	SimValue
1	Shyam − Ram	0.458
2	Krishna − Ram	0.364
3	Krishna − Shyam	0.833
4	Bishnu − Ram	0.446
5	Bishnu − Shyam	0.644
6	Bishnu − Krishna	0.515

3.4 Population Size

In order to choose the population size, the criterion of using of individuals in the population which is the double of the number of gene used to represent each individual is considered.

3.5 Fitness Function

The fitness of chromosome is determined by iterating on the rating matrix and prediction matrix and calculating the deviation of the rating value, which is the Mean Absolute Error (MAE). The fitness function of the genetic algorithm is used to prescribe the optimal similarity value. We assume that $(p_1, p_2, .., p_n)$ is the predicated ratings, obtained by the calculation of equation 2, for the user and $(q_1, q_2, ..., q_m)$ is the actual ratings of the user and the MAE is formulated as equation 1,

$$Fitness = MAE = \frac{\sum_{i=1}^{n} |p_i - q_i|}{M} \tag{1}$$

Here $M = mxn$ is the product of number of items and number of users.

3.6 Selection

Roulette wheel method is used for the selection. The roulette wheel is a fitness-proportionate selection method.

3.7 Crossover and Mutation

Crossover is a structured yet randomized information exchange between bits. The single-point crossover and single point mutation technique is used. High crossover probability and low mutation probability is taken to make the genetic algorithm perform better [8]. As the representation of the chromosome is in the matrix form, Matrix Crossover [9] is used for creating new individual generation wise. The representation of the chromosome is naturally suitable to two dimensional representation. Therefore, a two-dimensional encoding representation is designed and the traditional genetic algorithm is modified to fit the representation [10]. A random position of the matrix is selected as a crossover point and each gene above the point from an individual is mixed with the genes below that point from other individual to form a new individual. Similarly two individuals are formed from two parent chromosomes.

3.8 Recommendation

Once comparison between the user and the rest of the community of recommenders is completed, we have a set of similarity values, and the predicted ratings of unrated content can be computed. The equation 2 gives the predicted ratings, where multiplier k is a normalizing factor and is usually selected as $k = \sum_{c' \in C} |sim(c,c')|$. Here $r_{c,s}$ is the rating prediction for item s by user c. . After the generation of the predicted ratings for the items, and sorted according to the predicted value, the top-n items can be proposed to the end user as recommendations.

$$r_{c,s} = \bar{r}_c + k \sum_{c' \in C} sim(c,c')(r_{c',s} - \bar{r}_{c'}) \quad (2)$$

Example 1 *Suppose that there are 4 users and 4 items. The utility matrix(rating matrix) is shown in table 3. Table 4 and Table 5 are two individuals of any generation in genetic algorithm. Using the two individuals M and N the prediction ratings of the items in the utility matrix is calculated which is shown in Table 6 and Table 7.*

Table 3: Utility Matrix (User Rating Matrix)

User	CabinBoy	Picnic	KingPin	Jack
Ram	3	.	5	1
Shyam	1	2	.	3
Krishna	.	.	4	.
Bishnu	.	2	.	2

Table 4: Individual M

User	Ram	Shyam	Krishna	Bishnu
Ram	1	0.145	0.681	0.359
Shyam	0.145	1	0.362	0.674
Krishna	0.681	0.362	1	0.482
Bishnu	0.359	0.674	0.482	1

Table 5: Individual N

User	Ram	Shyam	Krishna	Bishnu
Ram	1	0.214	0.533	0.244
Shyam	0.214	1	0.606	0.016
Krishna	0.533	0.606	1	0.949
Bishnu	0.244	0.016	0.949	1

From this prediction ratings, the MAE of the two individuals is calculated to be 0.480 and 0.495 for similarity matrix M and N respectively. From this result individual M is considered better than individual N.

Table 6: Prediction ratings of M

User	CabinBoy	Picnic	KingPin	Jack
Ram	2.5	.	4	2.19
Shyam	3.61	2	.	1.91
Krishna	.	.	4.54	.
Bishnu	.	2	.	2.17

Table 7: Prediction ratings of N

User	CabinBoy	Picnic	KingPin	Jack
Ram	2.34	.	4	2.33
Shyam	3.51	2	.	1.53
Krishna	.	.	4.51	.
Bishnu	.	2	.	1.76

4. Experiment

Experiments were performed to compare the proposed system with other approaches that use GA and traditional similarity metrics. For comparisons, Bobadilla et al.'s work was chosen as the baseline system using GA.

4.1 Experimental Data

The MovieLens dataset was used for the experiments. MovieLens is a dataset collected by a research group from University of Minnesota called GroupLens for conducting research in the field of recommender systems [11]. The real rating data provided by GroupLens was used for experiments. It consists of 1,00,000 ratings with 943 users and 1682 movies. The data is anonymized to protect user identity. The possible values for ratings are 1,2,3,4 and 5 with 5 indicating that the user liked the movie the most and 1 that the user liked the movie the least. The dataset was divided into different set randomly. 20 percentage of data was used for training set whereas the remaining data was used for testing purpose. The data was divided into different chunks for multiple sets of experiment.

4.2 Experimental Setup

All experiments were conducted in a PC having i3-7100U, 2.40 GHz of processor, and 8 GB of RAM. Experiments were carried on the same machine and same data for both the proposed system and Bobadilla et al.'s system. The results of the proposed system were compared with other approaches using prediction accuracy and Mean Absolute Error (MAE) as recommendation quality metrics . Both systems were executed for 5 times each and average of the results was taken.

The parameters of the GA for the experiments was set as:

- Roulette-wheel selection is used to select parents.
- Number of individuals in each generation is same.
- Beside the elite individual that is carried on to the next generation, 80% of the next generation is created by crossover.
- Each gene has a 0.02 probability of being mutated.

The experiment on different set of initial population is shown in figure 1. The experiment using the individuals with ten elements shows that the optimum number of individuals for the population is 20 which is also the double of the number of elements.

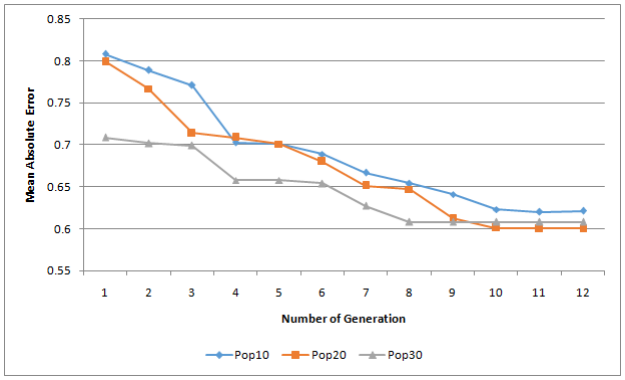

Figure 1: Variation of Initial population size

The optimum crossover and mutation probability is obtained from the experiment for numbers of generations using different values of probability. The result of the experiment is shown in figure 2.

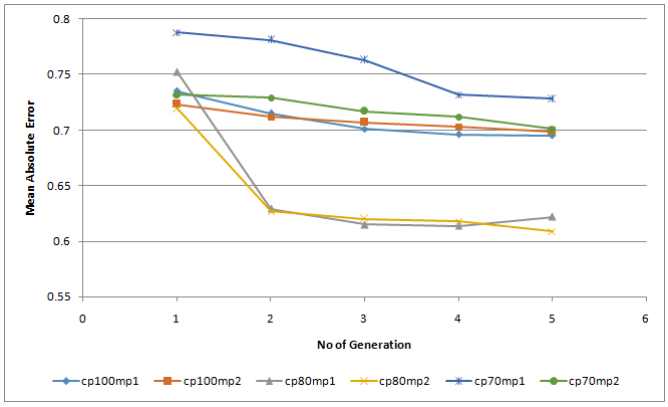

Figure 2: Selection of optimum crossover probability and mutation probability

5. Results and Analysis

Similarity values obtained from the genetic algorithm is expected to provide better quality and quicker results than the ones provided by the traditional metrics. Improvements are expected in the systems accuracy (MAE), in the coverage and ratings and prediction ratings as recommendation quality measures.

The experiment is carried out multiple times to see the deviation of values from average value. The graph in figure 3 shows the three runs of the proposed system. The experiment for the Bobadilla's system is also carried out for the validation of the proposed system shown in figure 4.

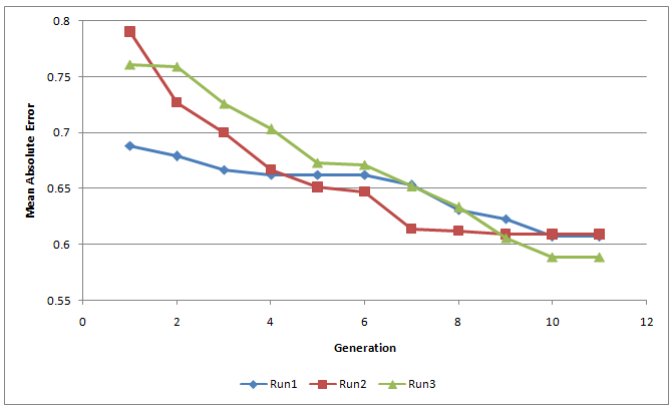

Figure 3: The MAE measures of Proposed System for 3 runs.

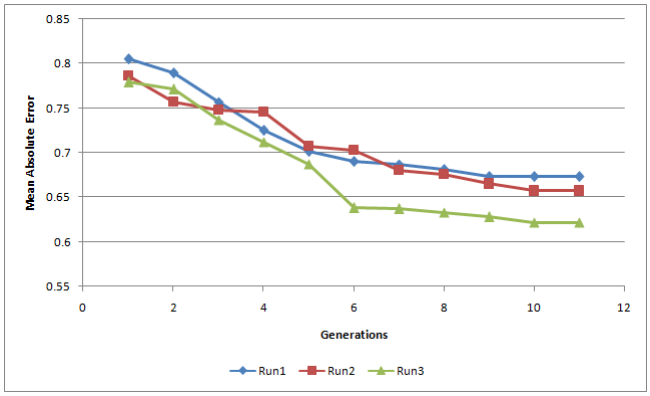

Figure 4: The MAE measures of Bobadilla System for 3 runs.

The figure 5 shows the average MAE measure of the proposed system compared with the average MAE measure of the Bobadilla's sytem after running the experiment for multiple times. The accuracy measure of the proposed system is better than than Bobadilla's system in some extent.

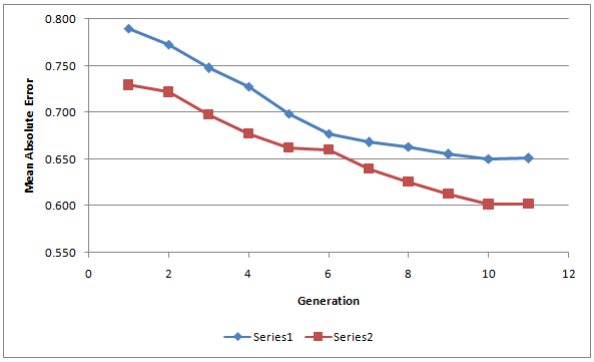

Figure 5: The Average MAE measures of proposed system and Bobadilla System.

6. Conclusions And Future Works

In collaborative filtering recommeder systems, the main challenge is to overcome cold start and sparsity problems while assuring quality of recommendations. A Genetic algorithm based Recommender System has been developed in this work to address these problems. The system adjusts the user-to-user similarity values based on training data using genetic algorithm. The result shows that the proposed system achieved high performance in accuracy and quality compared to the current state-of-the-art genetic based algorithms proposed by Bobadilla et. al. and the traditional cosine similarity metric.

There are many possibilities to further continue to improve the algorithm to increase accuracy of the results and efficiency. One of the approaches is to apply the adaptive crossover and mutation probability in the genetic algorithm. Also using the demographics data the recommendation quality and relevance can be further improved. The system can further developed in future in the below discussed points in future.

- The developed system is based on the single dataset, Movielens. In future, the system may be modified to work with other dataset like e-commerce, media etc.
- The rating information is only the information used for the evaluation of similarity values between user in the developed system. In future the user, item information along with demographics data can be used to improve the prediction accuracy of the recommender system.
- The system is developed and tested in a normal machine configuration so small set of data is used through out the experiment and the similarity evaluation process is offline. In future a highly configured machine may be used to work with the large dataset and the system may be modify to work on realtime.

References

[1] Jesus Bobadilla, Fernando Ortega, Antonio Hernando, and Javier Alcalá. Improving collaborative filtering recommender system results and performance using genetic algorithms. *Knowledge-based systems*, 24(8):1310–1316, 2011.

[2] Simon Fong, Yvonne Ho, and Yang Hang. Using genetic algorithm for hybrid modes of collaborative filtering in online recommenders. In *2008 Eighth International Conference on Hybrid Intelligent Systems*, pages 174–179. IEEE, 2008.

[3] Feng Zhang and Hui-you Chang. A collaborative filtering algorithm employing genetic clustering to ameliorate the scalability issue. In *2006 IEEE International Conference on e-Business Engineering (ICEBE'06)*, pages 331–338. IEEE, 2006.

[4] Jing Xiao, Ming Luo, Jie-Min Chen, and Jing-Jing Li. An item based collaborative filtering system combined with genetic algorithms using rating behavior. In *International Conference on Intelligent Computing*, pages 453–460. Springer, 2015.

[5] Gang Lv, Chunling Hu, and Shengbing Chen. Research on recommender system based on ontology and genetic algorithm. *Neurocomputing*, 187:92–97, 2016.

[6] Ayşe Dosdoğru, Faruk Geyik, and Mustafa Gocken. Genetic algorithm representation types for a two-stage supply chain distribution problem. pages 383–394, 01 2014.

[7] G.A. Vignaux and Zbigniew Michalewicz. A genetic algorithm for the linear transportation problem. *Systems, Man and Cybernetics, IEEE Transactions on*, 21:445 – 452, 04 1991.

[8] Nuwan I Senaratna. Genetic algorithms: The crossover-mutation debate. *Degree of Bachelor of Computer Science of the University of Colombo*, 2005.

[9] Mohamed Elsersy, Mohammed Zaki Abdelmagid, and Mahmoud Marie. Matrix based representation genetic algorithm for solving optical network design problem. 11 2008.

[10] Ming-Wen Tsai, Tzung-Pei Hong, and Woo-Tsong Lin. A two-dimensional genetic algorithm and its application to aircraft scheduling problem. *Mathematical Problems in Engineering*, 2015, 2015.

[11] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):19, 2016.