Journal of Engineering and Applied Sciences 11 (10): 2213-2221, 2016
ISSN: 1816-949X

Improved Efficiency of Social Recommender Systems Using ANN and GA

[1]Seyed Mohammad Hosseini and [2]Ali Asghar Safaei
[1]Department of Computer Engineering, Qeshm International Branch, Islamic Azad University, Qeshm, Iran
[2]Department of Biomedical Informatics, Faculty of Medical Sciences, Tarbiat Modares University, Tehran, Iran

Abstract: Currently, the rapid growth of internet and huge volume of informationrequire systems which are able to recommend the most appropriate products and services to the user. Using data mining tools, recommender systems can make appropriate suggestions to choose from large amounts of data. Traditional recommender systems, particularly collaborative filtering,used similarity criteria to selectsimilar neighbors for the active user and made suggestions based on estimate of their feedbacks. Similarity criteria considerably influence performance of these systems. Therefore, it is challenging to select the proper criteria. The main objective of this study is to develop a SRS which can make appropriate suggestions by feedbacks of other people and trust to them without having to find similar users. ANNs are known as one of the common methods used in thesesystems to explore relationships and trust. In fact, ANNs explore implicit relationships (trust) between trusted target users to make accurate suggestions. To increase efficiency of ANN in the suggested method, GA is used as feature selection method to find optimal set of features. By implementing and comparingthis hybrid algorithm with other similar algorithms, the results indicate error reduction (mean absolute error and root mean square error) in making suggestions to users.

Key words: Social recommender systems, artificial neural network, implicit relationships, feature selection, genetic algorithm

INTRODUCTION

Recommender system is an innovative system which provides useful information and can be used in various domains (Hsu, 2011). By analyzing behaviors of users, recommender system recommends the most suitable items (data, information, product, etc). This system is an approach to deal with problems caused by large and growing volume of information and helps usersachieve their goals faster among huge volume of data (Kangas, 2002). Different definitions have been developed for recommender systems. For example, Ting-Peng Liang presented a holistic and summarized definitionin which recommender systems are known as a subset of Decision Support Systems (DSS) and are defined as information systems which are ableto analyze past behaviors and make recommendations for current issues. Recommender systems are the most common personalized information retrieving technology which often act in two ways (Cornelis *et al*., 2007): predicting whether a particular item is preferred by a certain user (customer); predicting a batch of goods which may be preferred by a user. Recommender systems will both satisfy customers and increase sales (Li *et al*., 2005). Recommender systems are generally divided into three main categories. In the most common categorization, they are divided into three groups: content-based; knowledge-based and Collaborative Filtering (CF); however, a fourth group is also considered as hybrid recommender system (Li *et al*., 2005). Recommender systems are techniques and software tools which suggest a series of items (products and services) to users (Mahmood and Ricci, 2009; Resnick and Varian, 1997; Burke, 2007, 2002). These suggestions help users make decisions and facilitate the decision-making. These decisions may vary from buying a product, visiting a web page or listening to music. However, recommender systems usually focus on a particular type of items (such as books, web pages, music) in accordance with their designs. This means that items influence the design of a recommender system. With increasing spread of social

Corresponding Author: Seyed Mohammad Hosseini, Department of Computer Engineering, Qeshm international Branch, Islamic Azad University, Qeshm, Iran

networks and the need to use recommender systems in these networks to suggest appropriate items for the user such as friendship suggestions, an interesting domain have emerged for recommender systems in social networks. Social Recommender Systems (SRS) eliminate information overload on users of social media by providing the most attractive and most relevant contents. SRSs also increaseadoption, employment and participation of new and existing usersof social media. SRS refers to a recommender system which use social interactions as input data to increase accuracy and efficiency of the system. Content (blogs, wikis) (Guy *et al.*, 2009), tag, people and community recommenders often use customization methods which are compatible with needs and preferencesof a single user or a set of users.

Trust has been widely addressed in recommender systems, known as Trust-Aware Recommender Systems (TARSs). The main idea in these systemsis based on the fact that trusted users may have similar preferences. In fact, researchers have found a strong and direct correlation between trust and user preferences (Singla and Richardson, 2008). In social networks such as Facebook, people relate to each other;this relationship is called trust. These relationships form a trust network between users (Fig. 1). In trust matrix, $T_{u,v}$ = 1 if the user U trusts the user; otherwise, $T_{u,v}$ = 0 (Fig. 2). TARSs use profile information of users of a trust network who are related tothe target user and recommend appropriate items to the target user.

The extent to which information of these users influences recommendations to the target user depends on trustof target user to these users. In general, there are explicit and implicit trusts. Explicitly, trusted users can be imported manually or through links from social networks. Implicitly, trust can be achieved by history ofuser ratings (Li *et al.*,2005). Both explicit trust (expressed by the user) (Lee and Brusilovsky, 2009; Lewicki *et al.*, 1998) and implicit trust (Resnick and Varian, 1997; Kamvar *et al.*, 2003) have been used in this area. Although explicit trust is more accurate than implicit trust, this information often cannot be received directly from the user. Hence, the approach suggested in this study focuses on implicit trust relationships. In these methods, trust value is automatically generatedby the system rather than users. These methods usually estimate trust value based on scoring behavior of the user in the past. Music recommender system uses user interactions within social networks and other data published within scope of the linked open data as well as semantic web technology to

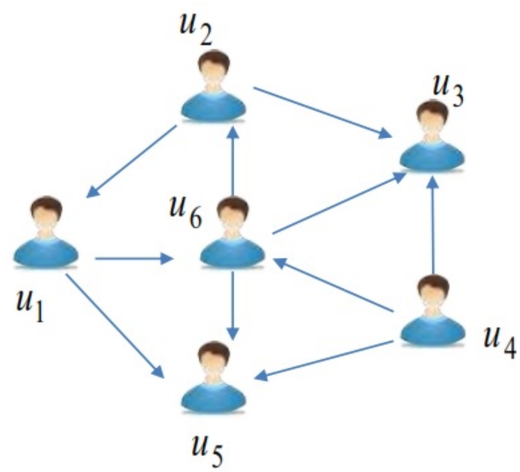

Fig. 1: Trust network

	u_1	u_2	u_3	u_4	u_5	u_6
u_1	0	0	0	0	1	1
u_2	1	0	1	0	0	0
u_3	0	0	0	0	0	0
u_4	0	0	1	0	1	1
u_5	0	0	0	0	0	0
u_6	0	1	1	0	1	0

Fig. 2: Trust matrix

extract RDF triples from music websites and semantic query and offers suggestions. New recommending method mentioned in (Kazienko *et al.*, 2011) provides preferences of other users in multimedia sharing system based on knowledge discovered in Multi-dimensional Social Networks (MSN). This system considers activities of users in separate layers of MSN. This process and personal weighting to each layer customize recommendations. In addition to these operational systems, Balabanovic, Bruke and Nguyen have studied recommender systems and developed the most efficient system possible (Burke, 2002). Some works categorize user tags in social networks. SRS introduced in (Cantador *et al.*, 2011) divides tags into four categories. This system initially assumes that the concept of tag exists in databases such as Yago (Suchanek *et al.*, 2008);

then it maps the concept to the relevant meaning. If the tag is not available in any database, the system maps it to a semantic unit available in database and finds the batch equivalent to that unit by using natural language processing techniques, separating tags and recognizing the tag functionsuch as noun, verb, etc. The main objective of this study is to obtain accurate results in discovering relationships and trust of recommender systems by: reviewing the types of recommender systems; collecting different parameters used to explore relationships and trust using data mining techniques; improving efficiency of recommender systems in predicting recommended items using neural network and feature selection; reducing the mean absolute error in prediction of recommended items.

Literature review: Emergence of recommender systems dates back almost to the mid-1990's. At the time, researchers were focused on voting structures. Since, the mid-1990's, a large number of recommender systemshave been implemented to help users achieve their favorite information. GCBR and GroupLens, article and Usenet news recommender systems and Bellcore Video Recommender, videos recommender system were designed and implemented afterwards. In traditional collaborative filtering systems, similarity criterion is one of the most influential factors of the performance of recommender systems. Due to sparseness of data in the rating matrix, similarity of the users cannot be estimated correctly. Trust-based collaborative filtering methods are a solution to this problem. In cases where there is trust explicitly, one can be confident that this additional information can compensate for deficiencies of rating matrix while trust calculation is important in cases where trustis to be achieved implicitly byrating matrix. In the suggested method, all people can help to recommendation by considering preferences of their ratings. In other words, even someone who is not similar tothe target user may have valuable information. The suggested algorithm uses an Artificial Neural Network (ANN) to discover implicit relationships in rating matrix which are used to find the appropriate recommendation. Because ANN is used to find trust, this system is not related to a specific formula. In the following, some works which have used trust to make suggestions to the target user are reviewed.

Simple user-centric collaborative filtering algorithm: Simple user-centric collaborative filtering algorithm first measures similarity of any user (who hasrated the target item) to the target user and use their weighted rates to the target item in predicting ratings of target user to the target item. Weight of ratings of any user denotes similarity of that user to the target user. Coefficient of Pearson correlation is one way to measure similarity of two users in a recommender system.

Coefficient of Pearson Correlation (CPC): CPC is generally considered as a main component of a basic recommender system which can only use information and previous ratings within the system to predict potential user ratings toitems. CPC can estimate similarity of two users by their ratings. In fact, closer ratings of two users to the same items indicate their higher similarity;thus, they will give nearly identical ratings to an item in a near future. For example, if two users,a andn, give close ratings to the books 1 and 2, they will probably have similar interest to the Book 3 and will rate the book similarly. Therefore, the user rating to the Book 3 is used to predict potential rating of the User A to the Book 3.$UCPC^{ab}$ denotesuser CPC between User a and User b as shown below (Eq. 1):

$$UCPC^{ab} = \frac{\sum_{i=1}^{I_{a,b}} (r_{a,i} - \bar{r}_a)\times(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i=1}^{I_{a,b}} (r_{a,i} - \bar{r}_a)^2} \times \sqrt{\sum_{i=1}^{I_{a,b}} (r_{b,i} - \bar{r}_a)^2}} \quad (1)$$

Where:

$r_{a,I} \in \{1, 5\}$ = Rating of the user a tothe item aindependently of the user b

$I_{a,b}$ = Set of items rated by both users a and b

Given similarity of the target user and any other users who rate the target item and actually form the set, $N^{sUCF} \subseteq I$, the ratings of the target user to the target item can be predicted. Equation 2 shows this prediction:

$$P_{a,i}^{N^{sUCF}} = \bar{r}_a + \frac{\sum_{u=1}^{N^{sUCF}} UCPC_{a,u}(r_{u,i} - \bar{r}_u)}{\sum_{u=1}^{N^{sUCF}} UCPC_{a,u}} \quad (2)$$

where, r_u is average rating of the User u to the item i. UCPC denotes CPC between the User a and target user u. In above equation, ratings of all users who rated the target item are considered for prediction. However, only ratings of the users who are highlysimilar to the target user are effective in prediction. For this purpose, the highest n can be used to isolate n users with highest similarity and only consider their ratings. CPC measures similarity of two users based on their ratings on only common items. This is a key weakness because two users who are common only in similar ratings on an itemcan generally be identified similar. To overcome this weakness, CPC and Jaccard can be combined to ensure that two users who rated many common items are preferred over those who rated just a few common items.

Tidal trust algorithm: Tidal Trust algorithm uses trust network of users. This algorithm has two criteria: shorter route between target user and members of trust network increases accuracy of recommendation; high trust increases accuracy of recommendation. These two criteria are used to determine trusted users of the target user. Trust between indirect user u and v is calculated by Eq. 3: where k is direct neighbor of the user u and v.Feedback of the trusted users, u or trust factor is used to predict rating of the user u to the item i. In Eq. 4, v denotes trusted users, u:

$$T_{u,v} = \frac{\Sigma T_{u,k} R_{k,i}}{\Sigma T_{u,k}} \tag{3}$$

Note that Tidal Trust algorithm uses implicit trust to provide recommendations:

$$R_{u,v} = \frac{\Sigma T_{u,k} R_{k,i}}{\Sigma T_{u,k}} \tag{4}$$

Mole trust algorithm: Mole trust algorithm is similar to Tidal trust algorithm. This approach has two basic steps. First, loops are removed from trust network. This speeds up algorithm because each user is required to visit once to determine trust. Therefore, trust network becomes a loop-free straight graph. Second, trust value or trust network exploration is calculated in the edge-free network.

Trust walker algorithm: Trust Walker algorithm uses random walk in trust network to predict rating of the target user to item i. Starting from target user, this algorithm calculates rating of item i by searching in trust network. Trust Walker algorithm improves accuracy of prediction by considering users in the shortest route and improves generality by items which are similar to the target item. As algorithms introduced, this algorithm uses implicit trust to provide recommendations.

MATERIALS AND METHODS

The suggested method: As noted above, one of the promising approaches for user rating prediction problem in recommender systems isANNwhich has exhibited good potential to predict rating. A major problem is training timewhich requires development or improvement of the model disregarding the scope of features. Unrelated data and additional features lead to poor ratings and largecomputations and reduced performance. In general, feature selection for machine learning algorithms reduces data size and space used for the algorithm and increases its speed. In some cases, it increases accuracy of classification. Using feature selection techniques, one can reduce the size of features and undermine those which are unused in decision-making to achieve a particular class. In order to find the optimal feature set for recommender system, a solution is to reduce datasetand weigh learning algorithm for analysis in order to reduce leaning time and use learning features of any taste. In fact, feature selection finds a subset which maximizes efficiency and accuracy of classification. Using all available parameters in rating matrix to evaluate and explore taste patterns and user preferences increases processing overhead, prolongs recommending process and thereby reduces efficiency of recommender systems. Since removal of worthless and useless data simplifies the problem and items are recommended faster and more accurately, feature selection is considered vital in a recommender system. Genetic Algorithm (GA) is one of the most optimal data mining techniques used to extract and select features. The model suggested in this study uses GA for feature selection. The GA researches based on evaluation function based on classification error -random generator function. In this method, a population is regenerated from candidate subsets. In each iteration, new elements are regenerated by using mutation and crossover operators on elements of previous population. Using an evaluation function, fitness of elements of the current population is determinedto select better elements as next generation. This method does not guarantee the best solution but always finds a good solutionrelative to the time in which the algorithm is allowed to run. Recommender system suggested in this study uses GA combined with ANN to develop a new model to reduce errors in making suggestions to the target user. Important parameters used for classification lead to higher accuracy and significantly reduce training and experiment time. The suggested algorithm works as follows.

Step 1: Initialization, generation of the initial population to the desired number, crossover and mutation on population, determination of the number of GA iterations and the number of survivors in each iteration.

Step 2: Each individual of the population shows a different solution. Using ANNfor each individual, fitnessis calculated for the test set (after a sufficient number of training). Finally, μ best individuals are selected as parents of next generation.

Step 3: Following steps are iterated M times to the end.

Step 4: Regeneration of a new population (children) from current population using crossover and mutation operators.

Step 5: Calculation offitness of children using ANN by following equation:

$$f = \frac{1}{[\sum_{i=1}^{k}(y_i - t_i)^2]} \tag{5}$$

Step 6: Selection of the best among children and parents as survivors of this generation; this in fact approaches towards optimal solution. Then, the algorithm moves toward next generation by new population. Note that, error of survivors is obtained beforehand; the algorithm continues if error is lower than a certain value (considering the problem); otherwise, the algorithm moves toward the next generation. In other words, the algorithm ends by usingcertain number of iterations and by achieving a certain amount of error.

Step 7: The algorithm ends.

Once final reduced feature subset is found, all features which do not exist in the reduced setare removed from the feature set and imported to ANN for training and testing.

Using GANN, the suggested algorithm discovers implicit relationships in rating matrix and uses them to find appropriate recommendations in the next step. This system consists of three general steps: findingk recommending users; training GANN to predict user ratings; integrating ratings of krecommenders to estimate user ratings. In fact, this system does not depend on specific relations because it uses ANN to find trust.

Finding k recommending users: At this point, k users are selected for recommendation. Since information of these users is input to ANN, the users are selected which have the most trust to each other. In fact, those users are selected whichare most similar, so that the network has enough training data. Similarity is definedas the number of items rated by both users (regardless of the rating).

Training GANN to predict user ratings: At this point, GANN can be usedto discover the relationship between feedbacks of active user and trusted users and store them for later use. The used network contains K input neurons and one output neuron. In design of GANN, the goal is to prevent growth of divergence and relate the shape and structure of the network to one or more numerical parameters, so that the structure of networks changes by changing these parameters. Evolutionary methods such as GA are widely used in various stages of ANN design because of their unique capabilities to find optimal values and ability to search in unpredictable spaces (Zadeh *et al*., 2002). In this study, ANN is used as classifier function to evaluate the population regenerated by GA.

The main dataset consists of N featureswhich is divided into different subsets with n_1, n_2, ... for use in GA. Then,these subsets are imported in backpropagation neural network with a fixed number of neurons in the hidden layer used for training. The number of neurons existing in the input layer depends on the number of features in the considered subset. Each subset of data is divided into two parts: training and testing. After training, network is tested by using some new data (test datasets). The number of incorrectly classified samples is considered as error of the relevant subset. Inverse error is a measure of fitness of that subset (individual of the population). Therefore, any subset of features has an estimation error which will help to determine the best subset. Thus, ANN directsindividuals in a GA population to find the most optimal solution. The last subset derived by GA (which is the most optimal) is trained again by an ANN for a larger number. Figure 3 shows the suggested hybrid model in details. The pseudo-code of the suggested algorithm is as follows:

Step 1: Find the k nearest users using similarity measure

Step 2: Transpose the created matrix to feed to GANN

Step 3: Dofor all training data

Step 3.1: Dofor all features
Step 3.1.1: DoFor all generations
Evaluate the fitness of all the chromosomes of the population.
Select the best chromosomes to reproduce using mutation and crossover.
Create a new generation with the new chromosomes created from the fittest of the previous generation

Step 3.1.2: Evaluate the fitness for all the chromosomes of the population

Step 3.1.3: Select the fittest chromosome of the population as the new subset of features

Step 4: Dofinal training for selected subset of features

Step 5: Predict the rate of target user

To implement GA, the GA Toolkit available in MATLAB is used (Mitchell, 1997). MATLAB is originally developed by using C++ language. The most serious problem in GA is to set parameters. The standard GA is used in test; the parameters are shown Table 1. Once GA selects important features, the feature.

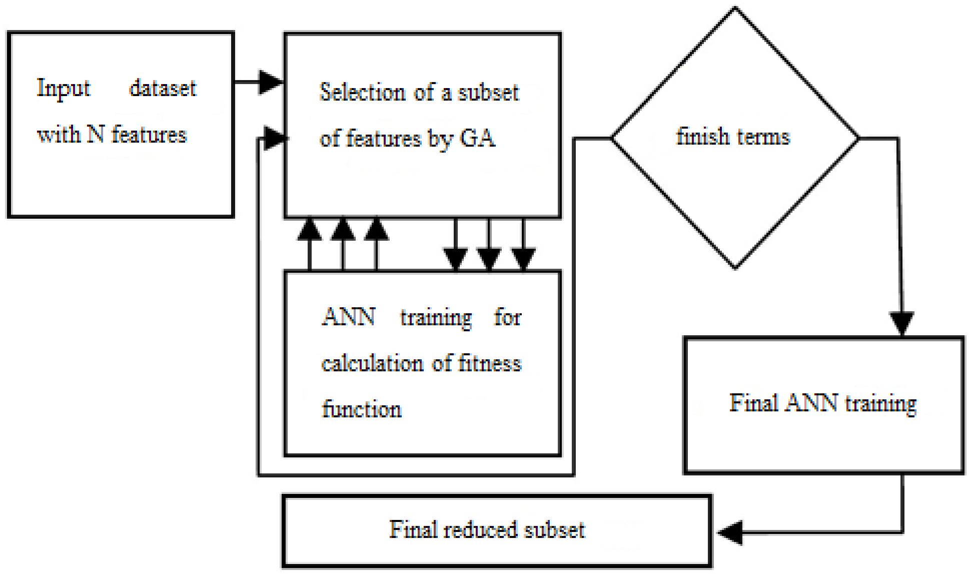

Fig. 3: Steps of the suggested method for implementation of product recommendation task

Table 1: Parameters used in GA

No. of initial population	No. of iterations	No. of children (λ)	Survivor selection method (μ)	Mutation rate	Crossover rate
75	30	λ = 7 μ	λ+μ	0.2	0.8

vector is made and given to ANN for training or testing. ANN is used as classifier in the suggested recommender system. ANN is used to evaluate fitness of subsets developed by GAin each iteration and to test final subset. In this study, ANN is trained by backpropagation algorithm. Various implementations have been developed for ANNs in recent years. Most of these packages are designed and implemented in academic environments; however, commercial products arealso available. ANN toolkit available in MATLAB is used in the suggested recommender system (Mitchell, 1996).

Evaluation of the suggestedmethod: In this study, the suggested method is evaluated byMovielenz dataset (https://movielens.org/) toidentify strengths and weaknesses of the suggested approach and analyze its performance compared with basic methods.

Data description: As noted earlier, this study uses Movie Lens dataset. MovieLensis related to Group Lens project at the University of Minnesota. This dataset contains 100000 ratings given by 943 users on 1682 movies; each user rated at least 20 movies. Demographic information of the dataset includes age, gender, occupation and region.The data was collected in 3 months from 19 September 1997 to 22 April 1998 from www.movielens.umn.edu. Then, those users who rated <20 movies or lacked demographic information were excluded. Users rate the movies from 1 to 5 (1 for bad, 2 for average, 3 for good, 4 for very good and 5 for excellent). For further details, see http://www.grouplens.org/. Considering the number of ratings, users and available items, sparsity of the MovieLens dataset is equal to 0.937 which is close to 1; thus, this dataset is highly sparse and favorable to test recommending algorithms which intend to overcome sparseness of rating matrix.

Evaluation criteria: Evaluation of a recommender system involvesmeasurement of accuracy of the system in predicting ratingsgiven by an active user to the target item. A recommender system can be evaluated in two ways.

Online evaluation: Measurement of similarity of the suggested items to sets rated by the user in the future is an indirect way to calculate user satisfactionin an online evaluation. In this method, the suggested set is formed and it waits for visits and new ratings of users. By calculating similarity in this set, evaluation of the system isrelatively accurate. In online evaluation of a recommender system, however, one must be careful about changes in user preferences

Offline evaluation: For offline evaluation of a recommender system using existing dataset, it is not required to recognize real user preferences. In fact, error rate and accuracy of recommender system can be achieved by predictinga percentage of ratings of a user and comparing it with actual values of ratings.

One of the major problems with online evaluation is the long time to wait for users to register in the system, visit the items and rate the visited items. This method is usually so time consuming that most studiesuse offline evaluation. This study also uses offline evaluation. Therefore, the evaluation criteria selected are related to offline evaluation. In the following,the suggested method is quantitatively compared with similar methods by two types of errors, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) which are two general criteria for evaluating accuracy of recommender systems (Herlocker *et al*., 2004). MAE calculates mean deviation between the predicted ratings ($pred_{u,i}$) and real ratings ($r_{u,i}$) for all users u?Uand all items in their test set ($testset_u$). In this equation, U representsuser sets:

$$MAE = \frac{\Sigma_{u\in U}\Sigma_{i\in testset_u}\left|pred_{u,i} - r_{u,i}\right|}{\Sigma_{u\in U} testset_u} \quad (6)$$

To evaluate more accurately, another criterion is defined here as Mean Absolute User Error (MAUE), given by. To measure MAUE, MAE is calculated separately for each user and averaged on these independent user errors (Eq. 7):

$$MAUE = \sum_{u\in U}\frac{MAE_u}{|U|} \quad (7)$$

Root Mean Square Error (RMSE) is another well known criterion used to evaluate the accuracy ofrating prediction. The error between actual and predicted ratings is expressed by:

$$RMSE = \frac{\sqrt{\Sigma_{u\in U}\Sigma_{i\in testset_u}(pred_{u,i} - r_{u,i})^2}}{\Sigma_{u\in U} testset_u} \quad (8)$$

RESULTS AND DISCUSSION

To predict the user-centric algorithm suggested in this study more accurately, similar and basic algorithms are considered as competitor algorithms. Thesecompetitor algorithms include:

- Simple Collaborative Filtering (SCF) (Resnick *et al*., 1994)
- Trust-based Recommender System (TRS) (O'Donovan and Smyth, 2005)
- Trust Walker algorithm

Note that comparisons (Fig. 4 and 5) also involve the suggested approach (GANN) without genetic algorithm Gain reducing errors. In experiments carried out,u.baseandu.testdatasets are used which divide data into training set and test set by 80% and 20%. In this dataset, each of the different sets u_1, u_2, ..., u $_5$randomly made from the original setseparately include training and test data. These separate sets allow the use of 5-Fold Cross Validation algorithm (Mitchel, 1997) to evaluate the system. In this way, part of the data (1/5) are considered as test data at each iteration. Figure 4 and 5 show test results using this algorithm as well as MAE and RMSE, respectively.

Obviously, thesuggested method reduces error rate of the recommender system. Since, collaborative filtering alone does not consider trust between users, it only covers a small fraction of predictable items. It does not offer high-quality recommendations. In comparison to other methods, it suffers severely from cold start problem. For this reason, trust models are used to improve quality further. TRS outperforms net implicit trust which only uses directly trusted neighbors. However, TRS fails for

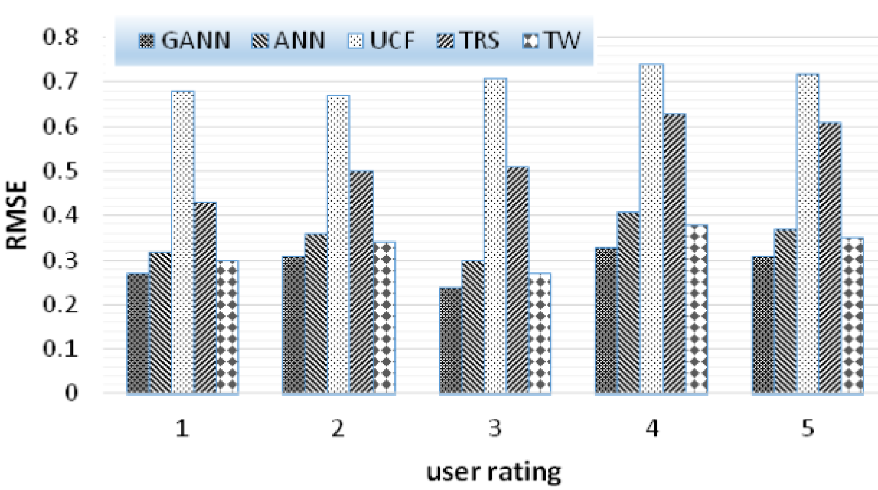

Fig. 4: Comparison of the suggested algorithm and competitor algorithms in terms of MAE

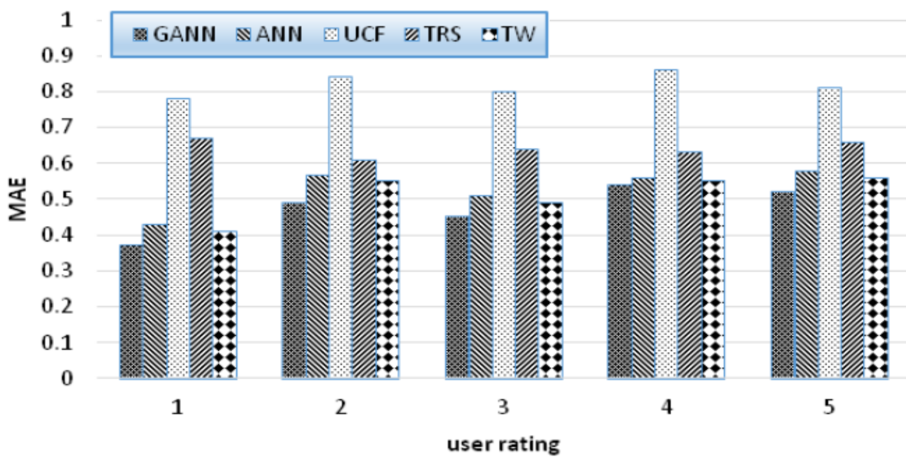

Fig. 5: comparison of the suggested algorithm and competitor algorithms in terms of RMSE

cold sers; this is due to dependence on collaborative filtering to find similar users before trust. The implicit trust learning method presented in this study obtains trust value based on GANN. Using a network of implicit trusts between users, prediction errors ofcollaborative filtering significantly decrease. In fact, similarity of the users to each other is detected better by both basic algorithm and trust network which implicitly exist in the data. The real-world trust to different people is a result of comparison and analysis oftheir feedbacks. The human mind can do this well and results of this analysis are recorded in the memory to be used in the future. The suggested system uses this process. Using ANN, one can discover the relationship between feedbacks of active user and trusted neighbors and store the result for future use.Implicit trusts discovered by ANN significantly improve recommendations and significantly reduce prediction errors of recommender system. Moreover, power and accuracy of classification increases after selection of important features and removal of irrelevant data. GA reduces incorrect classification rate, increases detection accuracy and acceleratesranking of parameters.

CONCLUSION

This study suggested a method to improve recommendations of a recommender system. The suggested method used a trust-based approach to solve problems of collaborative filtering. Using implicit trusts between users,prediction errors of the improved user-centric collaborative filtering were reduced. In fact, similarity of the users to each other is detected better by both basic algorithm and trust network which implicitly exist in the data. The real-world trust to different people is a result of comparison and analysis of their feedbacks. The human mind can do this well and results of this analysis are recorded in the memory to be used in the future. The suggested system used this process. Using ANN, one can discover the relationship between feedbacks of active user and trusted neighbors and store the result for future use. Implicit trusts discovered by ANN significantly improve recommendations and significantly reduce prediction errors of recommender system.Hence,this system does not depend on specific relations because it uses ANN to find trust. Evaluation platform was a dataset consisting of data collected by GroupLensproject at the University of Minnesota from www.movielens.umn.edu which containsratings of users onany item (video). Moreover, evaluation criteria includedMAE and RMSE. The results of evaluation indicated that the error was reduced and accuracy of the suggested method increased. Following remarks can be considered for future studies. This study usedGA as a feature selection techniqueas well as ANN to improve its efficiency for recommender system problem. ANN is a general technique used for solving various problems in different research areas.The hybrid model suggested in this study can be used for classification task in other applications. In addition, strengths of several classification methods can be used and combined with GANN to achieve higher accuracy bydiversifying the classifications used. Future works also can test thesuggested method on other types of datasets which have been developed objectively.

REFERENCES

Burke, R., 2002. Hybrid recommender systems: Survey and experiments. User Model. User-Adapted Interact., 12: 331-370.

Burke, R., 2007. Hybrid Web Recommender Systems. In: The Adaptive Web, Brusilovsky, P., K. Alfred and N. Wolfgang (Eds.). Springer, Berlin, Germany, ISBN:978-3-540-72079-9, pp: 377-408.

Cantador, I., I. Konstas and J.M. Jose, 2011. Categorising social tags to improve folksonomy-based recommendations. Web Semant. Sci. Serv. Agents World Wide Web, 9: 1-15.

Cornelis, C., J. Lu, X. Guo and G. Zhang, 2007. One-and-only item recommendation with fuzzy logic techniques. Inf. Sci., 177: 4906-4921.

Guy, I., I. Ronen and E. Wilcox, 2009. Do you know?: Recommending people to invite into your social network. Proceedings of the 14th International Conference on Intelligent User Interfaces, February 8-11, 2009, Sanibel Island, Florida, USA., pp: 77-86.

Herlocker, J.L., J.A. Konstan, L.G. Terveen and J. Reidl, 2004. Evaluating collaborative filtering recommender systems. ACM Trans. Infrom. Syst., 22: 5-53.

Hsu, M.H., 2011. Proposing a charting recommender system for second-language nurses. Expert Syst. Appl., 38: 9281-9286.

Kamvar, S.D., M. T. Schlosser and H. Garcia-Molina, 2003. The eigentrust algorithm for reputation management in p2p networks. Proceedings of the 12th International Conference on World Wide Web, May 20-24, 2003, Budapest, Hungary, pp: 640-651.

Kangas, S., 2002. Collaborative filtering and recommendation systems. VTT. Inf. Technol., 2002: 1-29.

Kazienko, P., K. Musial and T. Kajdanowicz, 2011. Multidimensional social network in the social recommender system. IEEE. Trans. Syst. Man Cybern. Part A Syst. Hum., 41: 746-759.

Lee, D.H. and P. Brusilovsky, 2009. Does trust influence information similarity?. Recommender Syst. Social Web, 10: 71-74.

Lewicki, R.J., D.J. McAllister and R.J. Bies, 1998. Trust and distrust: New relationships and realities. Acad. Manage. Rev., 23: 438-458.

Li, Y., L. Lu and L. Xuefeng, 2005. A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce. Expert Syst. Appl., 28: 67-77.

Mahmood, T. and F. Ricci, 2009. Improving recommender systems with adaptive conversational strategies. Proceedings of the 20th ACM Conference on Hypertext and Hypermedia, June 29-July 1, 2009, ACM, New York, USA., ISBN:978-1-60558-486-7, pp: 73-82.

Mitchell, M., 1996. An Introduction to Genetic Algorithms. 1st Edn., Massachusetts Institute of Technology, A Bradford Book, Cambridge, Boston, ISBN: 0262133164.

Mitchell, T.M., 1997. Machine Learning. 1st Edn., McGraw-Hill, New York, ISBN: 978-0070428072.

O'Donovan, J. and B. Smyth, 2005. Trust in recommender systems. Proceedings of the IUI`05, Jan. 9-12, ACM Press, San Diego, California, USA., pp: 167-174.

Resnick, P. and H.R. Varian, 1997. Recommender systems. Commun. ACM., 40: 56-58.

Resnick, P., N. Lakovou, M. Sushak, P. Bergstrom and J. Riedl, 1994. Group lens: An open architecture for collaborative filtering of Netnews. Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, Oct. 22-26, Chapel Hill, North Carolina, United States, ACM Press, pp: 175-186.

Singla, P. and M. Richardson, 2008. Yes, there is a correlation: From social networks to personal behavior on the web. Proceedings of the 17th International Conference on World Wide Web, April 21-25, 2008, ACM, New York, USA., ISBN:978-1-60558-085-2, pp: 655-664.

Suchanek, F.M., G. Kasneci and G. Weikum, 2008. Yago: A large ontology from wikipedia and wordnet. Web Semant. Sci. Serv. Agents World Wide Web, 6: 203-217.

Zadeh, N.N., A. Darvizeh, M. Darvizeh and H. Gharababaei, 2002. Modelling of explosive cutting process of plates using GMDH-type neural network and singular value decomposition. J. Mater. Process. Technol., 128: 80-87.