Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 62 – 77
© TÜBİTAK
doi:10.3906/elk-2004-138

Evolutionary neural networks for improving the prediction performance of recommender systems

Berna ŞEREF[1,*], **Erkan BOSTANCI**[2], **Mehmet Serdar GÜZEL**[2]
[1]The Scientific and Technological Research Council of Turkey, Ankara, Turkey
[2]Department of Computer Engineering, Faculty of Engineering, Ankara University, Ankara, Turkey

Received: 20.04.2020 • **Accepted/Published Online:** 10.08.2020 • **Final Version:** 27.01.2021

Abstract: Recommender systems provide recommendations to users using background data such as ratings of users about items and features of items. These systems are used in several areas such as e-commerce, news websites, and article websites. By using recommender systems, customers are provided with relevant data as soon as possible and are able to make good decisions. There are more studies about recommender systems and improving their performance. In this study, prediction performances of neural networks are evaluated and their performances are improved using genetic algorithms. Performances obtained in this study are compared with those of other studies. After that, superiority of this study is shown. While multilayer perceptron, generalized feed-forward network, and coactive neuro fuzzy inference systems were used as neural network algorithms, Movielens 100K and Movielens 1M datasets, which are widely preferred in recommender system studies, were used to train and test the system in the present study. Mean square error and root mean square error were employed as performance metrics. As a result, it was observed that genetic algorithm improves performance of neural networks, and prediction performance of hybrid combination of neural networks and genetic algorithm is superior to prediction performance of recommender systems available in the literature.

Key words: Recommender systems, prediction, neural networks, genetic algorithms

1. Introduction

Rapid development of the network causes exponential growth of information and as a result, it is hard to get useful information. This situation is called "information overload" problem. To solve this problem, information retrieval systems which are represented by search engine and intelligent recommendation systems are used. When these systems are compared based on criterion of satisfying personalized needs of information, it is observed that intelligent recommendation systems show better results. The reason behind this is when search engines are used and when the users enter the same keywords as an input to the search engine, they are likely to get similar results [1, 2]. As a conclusion, information retrieval systems cannot deal with the personalized needs of information completely. On the other hand, intelligent recommender systems focus on personalized needs and interests to recommend information or product. These systems use user data as an input, analyze the interests of user and according to the interests, make recommendations [1].

Recommender systems are a kind of information filtering systems that aim to make sensible recommendations by using user preferences, environmental contexts, and user contexts [3]. Such systems predict rating or preference of an item [4, 5]. As a result, they enable making a decision on an item in a large space of

*Correspondence: berna.seref@tubitak.gov.tr

possible items in a short time. Recommender systems are used to recommend movies, music, news, books, social tags, products, restaurants, financial services, life insurance, Facebook friends, and Twitter followers [4]. These systems are also used in a variety of web sites. For example, YouTube, an online video community, uses video recommendation system and recommends videos based on past activities of user on YouTube. LinkedIn, a business-oriented social networking site, recommends people, jobs, groups, and communities that can attract the user. Amazon, an e-commerce site, sells a variety of products such as books, electronic devices, and clothes based on item to item collaborative filtering (CF) [4, 6].

Recommendation systems are classified into the following groups: content-based filtering systems, CF systems, context-aware recommendation, knowledge-based recommendation, demographic filtering systems, and hybrid recommender systems [7]. These classifications are based on information and knowledge sources that the systems use to recommend items. For example, while content-based filtering systems use information about active user and items [4] and recommend items similar to the ones preferred in the past [7], CF systems need information about a set of users and their relations with the item to make recommendation for the active user [4]. CF is divided into two classes, which are user-based CF and item-based CF. User based CF finds the users who have similar interests to the target users. Item-based CF finds the similarity between items by analyzing the user behavior [8]. Context is any information which is used to analyze situation of a person, place, or object [9, 10]. Context-aware recommender systems focus on predicting how a given user will like an item. These systems use user rating or item interactions, context in which ratings/interactions were produced, and user's aspects and context at request time to make predictions [11, 12]. Knowledge-based recommendation systems use knowledge about users and products, and analyze which products satisfy the user's requirements [7]. Demographic filtering systems utilize demographic classes such as college students, teenagers, women, and men to make recommendations [7]. Hybrid recommender systems make recommendations by combining two or more recommendation techniques [7].

In this study, recommender systems improved by using artificial neural networks (ANNs) and genetic algorithms are proposed. After that, performances of the proposed systems are compared with performances of the recommender systems available in the literature. Mean square error and root mean square error are preferred to measure performance of the system. As a result, it is observed that prediction performance of improved neural networks is better than prediction performance of recommender systems avaliable in the literature. This situation is shown using example studies.

The contributions of our study can be highlighted as follows:

1. A hybrid system combining genetic algorithms with neural networks for recommendation systems is proposed.

2. The optimization performed by the genetic algorithm reduces the mean squared error (MSE).

3. A statistical performance comparison with the existing literature is performed and the superiority of the hybrid approach is demonstrated.

The rest of the paper is structured as follows. Section 2 presents a literature review and background. Section 3 is dedicated to the explanation of the proposed evolutionary neural network recommendation system. Section 4 presents experimental design such as dataset and evaluation metrics. While Section 5 contains results such as accuracy of the proposed approach, Section 6 compares the proposed approach with other studies. Finally, the last section concludes the paper.

2. Literature review and background

Recommender systems are a type of information filtering systems that assist people in decision making process. As a result, people choose one product or service in a short time. Some methods are used to improve performance of recommender systems. Fuzzy logic, genetic algorithms, and hybrid approaches can be given as example of such systems. A lot of experiments are carried out to increase accuracy rate of predictions and make better recommendations.

Siddiquee et al. [13], Verma et al. [14], and Jeon et al. [15] focus on applying fuzzy logic in their studies. On the other hand, Yigit Sert et al. [16], Parvin et al. [17], Alhijawi and Kilani [18], and Shou-Qiang and Ming [19] use genetic algorithms to make better recommendations. Siddiquee et al. [13] measure the similarities using Euclidean distance, Manhattan distance, Pearson coefficient, and cosine similarity measures and observe that Euclidean distance shows better performance. Verma et al. [14] propose a system which uses CF and fuzzy c-means clustering algorithms and handles sparsity and cold start problems. After the experiments, they observe that the fuzzy clustering algorithm yields better results than the k-means method. Jeon et al. [15] use CF and fuzzy system for their prediction system and they observe that the fuzzy system improves the performance of the CF system. Yigit Sert et al. [16] aim to improve prediction accuracy of recommender systems using artificial bee colony and genetic algorithms. Parvin et al. [17] study on a recommender system which uses genetic algorithm and trust statements and propose a two-step CF method called TCFGA. After the experiments, they observe that this proposed method shows a good performance. Alhijawi and Kilani [18] use genetic algorithm to calculate similarity between users and they propose a recommender system called "Simgen". After the experiments, they get 46% and 38% improvements in prediction quality. Shou-Qiang and Ming [19] focus on scalability problem and propose Hadoop cluster recommender system. They use a hybrid CF algorithm based on genetic algorithm optimization.

There are more studies which focus on genetic algorithm to improve performance of recommender systems. Ar and Bostanci [20] use genetic algorithm to adjust similarity weights using genetic operators. After the experiments, they observe that their proposed method gives better accuracy. Kim et al. [21] and Inoue et al. [22] propose a recommender system for music. While Kim et al. [21] recommend a novel recommender system for music data by combining content-based filtering technique and genetic algorithm with the aim of adapting and responding to immediate changes in users' preferences, Inoue et al. [22] propose distributed genetic algorithm for music recommendation systems and get better results. Soliman et al. [23] use genetic algorithm to predict interest of the user for unvisited locations. Hassan and Hamada [24] and Hamada et al. [25] use an adaptive genetic algorithm to improve prediction accuracy performance of multicriteria recommender systems. Agrawal and Jain [26] propose a movie recommendation system using support vector machine and genetic algorithm and get an improvement in the accuracy, quality, and scalability. Hosseinpourpia and Oskoei [27] use genetic algorithm with the aim of estimating parameters of multifaceted trust model. After the experimens they observe that prediction accuracy is increased. Janjarassuk and Puengrusme [28] use genetic algorithm to develop a product recommendation system. This system recommends the best recommendation for a combination of products to the customers. Ciaramella et al. [29] focus on adapting a resource recommender to the behavior of the specific user to improve the effectiveness and reliability in suggesting the correct resources to the user. As a result, they use genetic algorithm to make this idea real.

The literature presents several studies focusing on using neural networks for recommender systems. For example, Devi et al. [30] focus on trust between users and use probabilistic neural network to calculate it.

Hassan and Hamada [31] prefer using ANN to model the criteria ratings and analyze performance of neural networks for multicriteria recommender systems. Chakrabarti and Das [32] discuss neural network methods used for CF systems. They observe that using neural networks for CF improves efficiency of the personalization process. Da'u and Salim [33] work on a sentiment-aware deep recommender system with neural attention network based on aspects of products and user sentiments. After the experiments, they observe that the model shows better results when compared to the state-of-the-art methods. Gong and Ye [34] propose a recommendation system based on backpropagation (BP) neural networks and item-based CF. While they fill the vacant ratings using BP neural networks, they form nearest neighborhood using item-based CF. Mai et al. [35] focus on e-commerce recommendation systems and propose a neural networks-based clustering CF algorithm with the aim of overcoming the sparsity problem and getting more effective and more accurate recommendation results. Almaghrabi and Chetty [36] propose a new deep learning-based framework with the aim of capturing the deep and hidden interactions between users and items and improving the performance of recommender system. As a result, they improve the performance of CF. Sanandaj and Alizadeh [37] work on a hybrid recommender system based on CF techniques and content-based filtering combined with ANN.

Our approach takes a different direction from the studies presented above. In our approach, firstly, we predicted the rates using neural networks such as multilayer perceptron, generalized feed-forward, and coactive neural fuzzy inference system (CANFIS). After that we applied genetic algorithms to inputs and processing elements of these networks with the aim of improving accuracy of rate prediction system. While conjugate gradient and momentum methods were used as learning rules, the Takagi-Sugeno-Kang and Tsukamoto models were used as fuzzy models. Our approach will be evaluated below.

2.1. Multilayer perceptron

Multilayer perceptrons (MLPs) are feed-forward neural networks. They have one or more hidden layers between input layer and output layer and they can form arbitrarily complex decision function thanks to the nonlinearity activation function with each node [38–40].

In MLP, input $s(k)$ and desired $d(k)$ are applied to the network which produce output $y(k)$. Output $y(k)$ is defined in Equation (1) [38].

$$y(k) = f(\sum_{j=0}^{n-1}(s_j w_j) + b) \tag{1}$$

In (1), while b is bias, $\mathrm{w_j}$ is the weight matrix associated with input line j. Error signal $e(k)$ is defined in (2) [38].

$$e(k) = d(k) - y(k) \tag{2}$$

Error signal is propagated back to the hidden layers to get less errors. BP algorithm is used to update weights and thresholds. This situation continues until getting minimum number of mean square error [38, 41].

2.2. Generalized feed-forward network

Such networks are generalization of MLPs trained by backpropagation algorithm. In this network, connections can jump over one or more layers. In theory, a problem can be solved by both MLP and generalized feed-forward network, but in practice, generalized feed-forward network shows better performance. For example, for two-spiral problem, we need more training epochs when MLP is used. Generalized feed-forward network solves the problems more quickly and more efficiently [42].

2.3. CANFIS

Such networks are integration of fuzzy inputs with a neural network [43]. In this system, every node in the first layer is represented as x_1, x_2, ..., x_n. They show membership grade of the fuzzy set and the degree to which the input vectors belong to one of the fuzzy set. Product of all the outputs which belong to the first layer is computed to be transmitted to the second layer. After that, weight normalization is calculated [44].

2.4. Learning rules

In this study, conjugate gradient and momentum methods were used as learning rules to train and test the systems for all network algorithms. These rules are explained below.

The conjugate gradient method, which produces numerical solution, is used to optimize linear and nonlinear systems. In addition, it can be used as both iterative algorithm and direct method. This method enables solving large systems[1].

The momentum algorithm can accelerate gradient descent using previous gradients in the update rule[2] and it accelerates learning in the situations such as facing high curvature, small but consistent gradients, and noisy gradients[3].

2.5. Fuzzy models

In this study, the Takagi-Sugeno-Kang and Tsukamoto fuzzy models were preferred as fuzzy models to train and test the system when CANFIS was used as a neural network algorithm. These models are explained below.

In the Takagi-Sugeno-Kang model, consequence parts are represented by linear functions of input variables. This model is able to approximate complex nonlinear systems. To achieve that, it uses a few rules but gives higher modeling accuracy [45].

The Tsukamoto model can be used to solve nonbinary and nonlinear problems. Fuzzyfication, rule "IF-THEN", inference, and defuzzyfication processes are steps of the model. The end result is calculated using weighted average [46].

3. Evolutionary neural networks recommendation system

In this study, prediction systems were developed using neural networks. After that, prediction performances of the neural networks were improved using genetic algorithms. As neural networks; MLP, generalized feed-forward network, and CANFIS were used. For MLP and generalized feed-forward network, all the experiments were carried out for 3, 5, and 7 hidden layers. For all the neural networks; momentum and conjugate gradient learning rules were tried to get better results. For CANFIS model, while Bell function was used as a membership function, the Takagi-Sugeno-Kang and Tsukamoto were used as fuzzy models. MSE and root mean square error (RMSE) were used to measure the performance of the system.

[1] Website `https://optimization.mccormick.northwestern.edu/index.php/Conjugate_gradient_methods` accessed[19 December 2019]

[2] Website `https://medium.com/konvergen/momentum-method-and-nesterov-accelerated-gradient-487ba776c987` accessed[19 December 2019]

[3] Website `https://cedar.buffalo.edu/~srihari/CSE676/8.3%20BasicOptimizn.pdf` accessed[19 December 2019]

Genetic algorithm was applied to the inputs because their importance to the model was not known. As a result, genetic algorithm would define if these inputs would be selected or deselected for the model to find combination of inputs that produces the lowest error rates. In addition, genetic algorithm was applied to processing elements to optimize the number of hidden layer processing elements. Fitness calculation was carried out using the lowest MSE achieved with the cross-validation test. Tournament selection was chosen to select the best individuals. The best 10% of the population joined the tournament. While crossover rate was defined as 0.15, mutation rate was defined as 0.7.

All the experiments were carried out for ten times and the average MSE and RMSE values for cross-validation results were calculated to compare the performance obtained in this study with the others. The proposed prediction system is depicted in Figures 1–3 . The same steps were carried out to apply genetic algorithms to inputs and processing elements.

4. Experimental design

This section gives information about datasets and performance evaluation criteria used in this study to train and test the model and evaluate the performances of the proposed neural network-based prediction system and evolutionary neural network-based prediction system.

4.1. Dataset

MovieLens 100K Dataset which has almost 100,000 ratings from 1000 users on 1700 movies and MovieLens 1M Dataset [4], which has almost 1 million ratings from 6000 users on 4000 movies were preferred as a dataset to train and test the system. While 10% of the dataset was defined randomly for cross-validation, 20% of the dataset was defined as a test data randomly to test the system.

GroupLens Research Project from the University of Minnesota collected Movielens datasets. In Movielens 100K dataset, ratings are numbered from 1 to 5. This dataset has been collected through Movielens web site for 7 months. This dataset does not include any information of those who rated less than 20 movies and who did not complete demographic information. Movielens 1M dataset consists of ratings, movies, and users files.

4.2. Evaluation metrics

In this study, MSE and RMSE values were used to evaluate performances of the proposed systems. They are explained below.

MSE and RMSE are defined in (3) and (4) . In (3) and (4), while p_i is prediction, a_i is true numerical rating.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2 \tag{3}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2} \tag{4}$$

[4] Website https://grouplens.org/datasets/movielens/ accessed[16 December 2019]

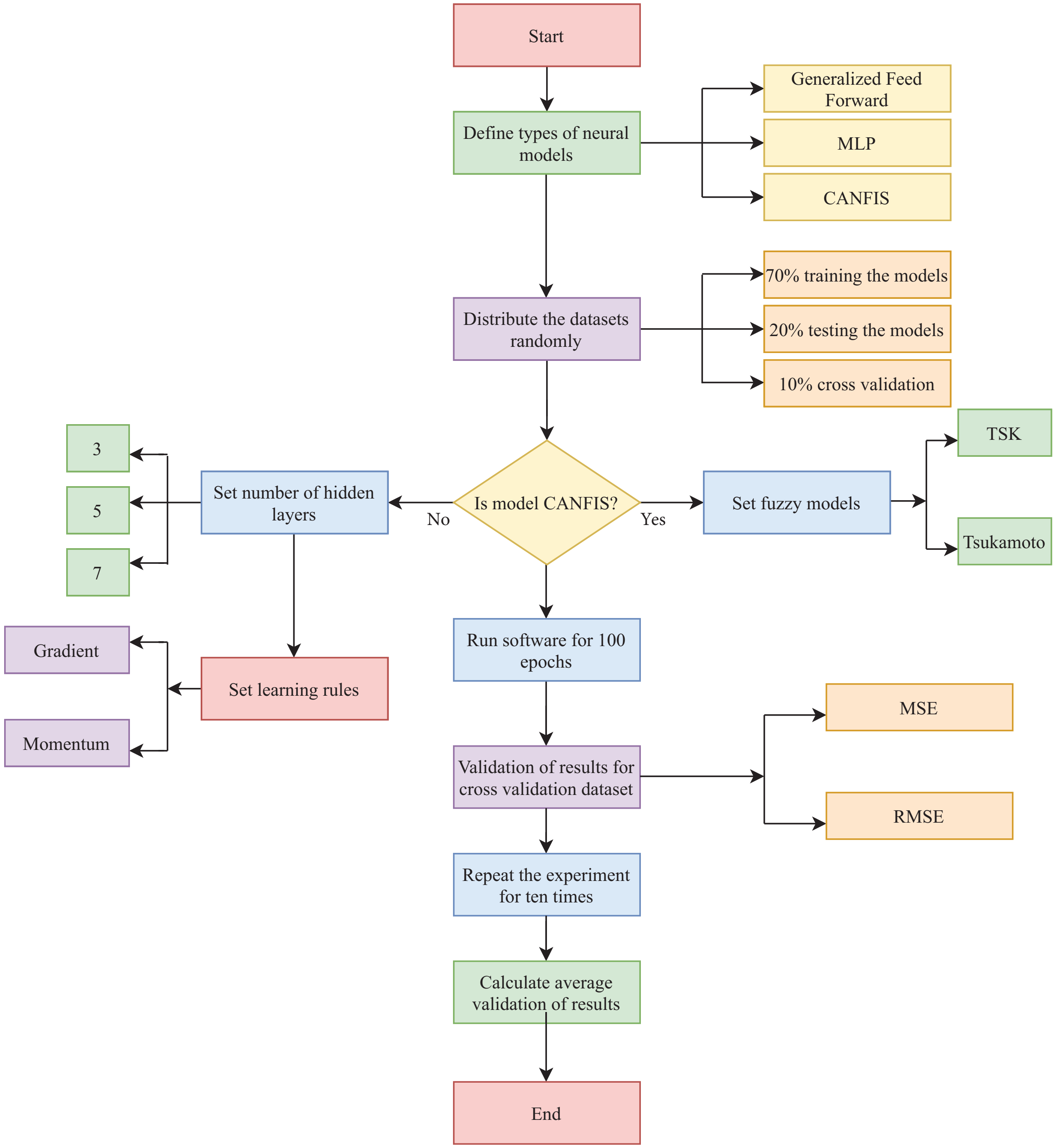

Figure 1. Flowchart of the proposed neural network-based prediction system.

5. Results

In this section, prediction performance of MLP, generalized feed-forward network, and CANFIS methods are compared using MSE values for the Movielens 100K and 1M datasets. The results are given in two groups which are before the optimization and after the optimization. As a result, we get the best methods for the

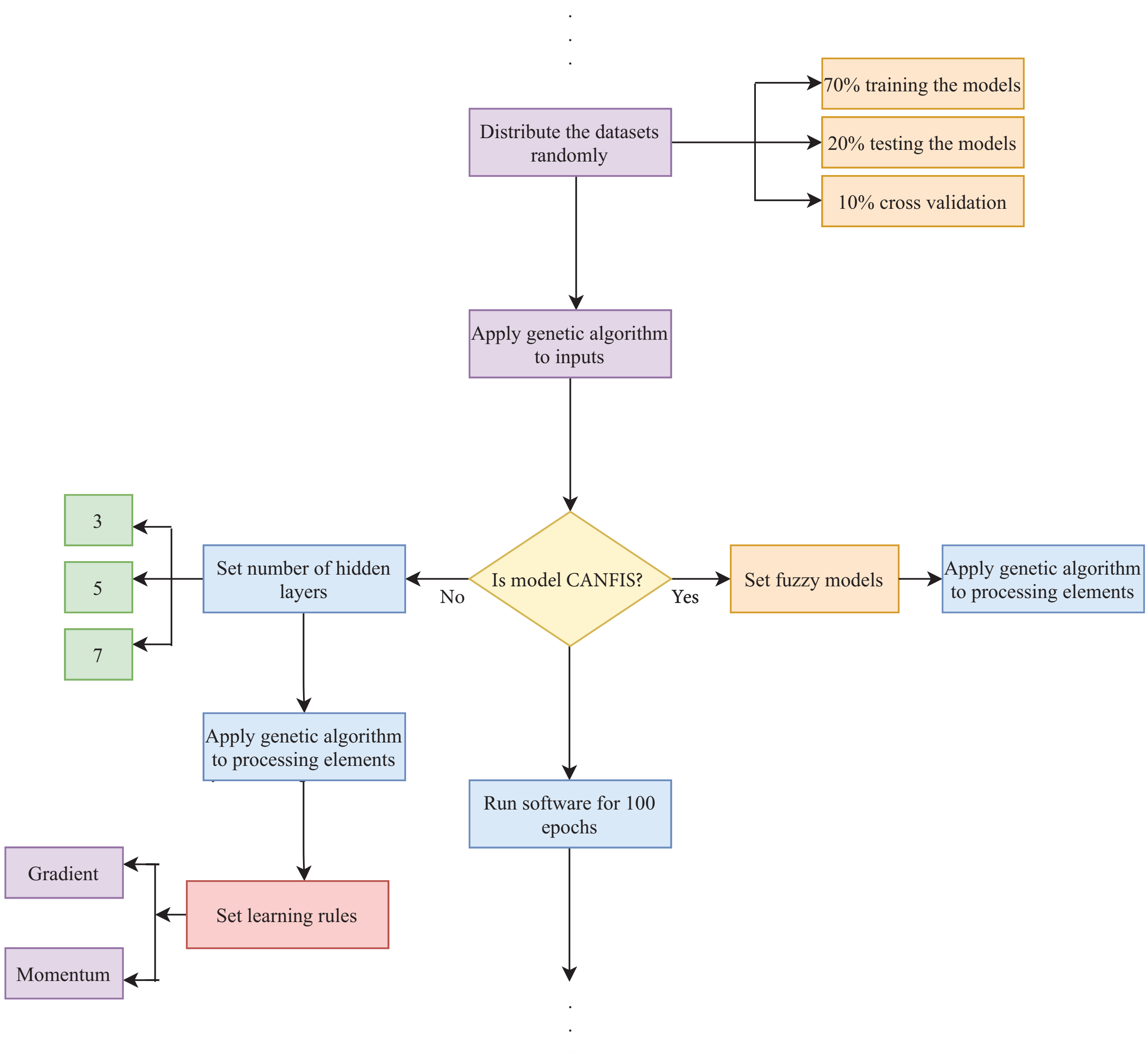

Figure 2. Flowchart of the proposed evolutionary neural network-based prediction system.

two datasets which give the lowest MSE and RMSE values. After that, prediction performances of these two datasets are compared using t-Test.

5.1. Performance evaluation

MSE cross-validation results for the MovieLens 100K Dataset are given in Tables 1–3. While Table 1 represents MSE results of MLP algorithm, Tables 2 and 3 represent MSE results of generalized feed-forward network and CANFIS algorithms before and after the optimization. When we focus on Table 1, we observe improvements after the optimization by using momentum and gradient learning rules with 3 and 5 hidden layers. There are improvements when generalized feed-forward network is preferred as a neural network instead of MLP. These

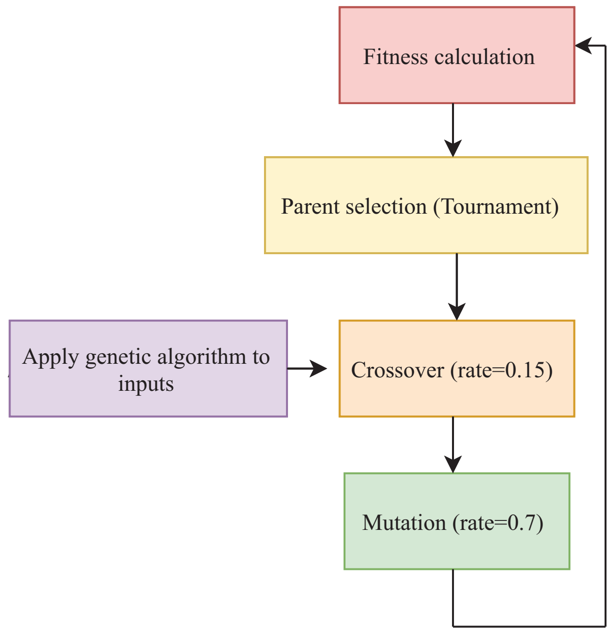

Figure 3. Flowchart of genetic algorithm's details for the proposed evolutionary neural network-based prediction system.

improvements are observed when momentum and gradient learning rules are applied with 5, and 3 and 5 hidden layers respectively. There is no improvement after applying genetic algorithms to the CANFIS network.

Table 1. MSE results using MLP for Movielens 100K dataset.

	MLP					
	Momentum			Gradient		
	# of hidden layers			# of hidden layers		
	3	5	7	3	5	7
Before optimization	0.31728	0.26993	0.25796	0.29472	0.47740	0.26429
After optimization	0.25788	0.25735	0.25796	0.24838	0.26872	0.26429

Table 2. MSE results using generalized feed-forward for Movielens 100K dataset.

	Generalized feed-forward network					
	Momentum			Gradient		
	# of hidden layers			# of hidden layers		
	3	5	7	3	5	7
Before optimization	0.40421	0.61841	0.29510	1.15396	0.70673	0.79529
After optimization	0.40421	0.45736	0.29510	0.46800	0.58902	0.79529

MSE cross-validation results for MovieLens 1M Dataset are given in Tables 4–6. When we focus on these tables, we observe improvements after the optimizations using both of the learning rules and fuzzy models. While the lowest MSE value is observed by applying momentum learning rule with 3 hidden layers to MLP after the optimization, the lowest MSE value is obtained by applying momentum learning rule with 3 hidden layers to generalized feed-forward network before the optimization. For CANFIS, the lowest MSE value belongs to momentum learning rule applied with the Takagi-Sugeno-Kang fuzzy model.

Table 3. MSE results using CANFIS for Movielens 100K dataset.

	CANFIS			
	Takagi-Sugeno-Kang		Tsukamoto	
	Momentum	Gradient	Momentum	Gradient
Before optimization	0.25219	0.25587	0.33036	0.28264
After optimization	0.25219	0.25587	0.33036	0.28264

Table 4. MSE results using MLP for Movielens 1M dataset.

	MLP					
	Momentum			Gradient		
	# of hidden layers			# of hidden layers		
	3	5	7	3	5	7
Before optimization	0.37336	0.25774	0.25824	0.27702	0.38233	0.27054
After optimization	0.24820	0.25680	0.25824	0.24948	0.25092	0.24955

Table 5. MSE results using generalized feed-forward for Movielens 1M dataset.

	Generalized feed-forward network					
	Momentum			Gradient		
	# of hidden layers			# of hidden layers		
	3	5	7	3	5	7
Before optimization	0.21458	0.26927	0.29676	0.32991	0.70883	0.84200
After optimization	0.21458	0.24986	0.25639	0.25490	0.31410	0.28433

Table 6. MSE results using CANFIS for Movielens 1M dataset.

	CANFIS			
	Takagi-Sugeno-Kang		Tsukamoto	
	Momentum	Gradient	Momentum	Gradient
Before optimization	0.25825	1.09930	0.37259	0.28741
After optimization	0.25825	0.43654	0.33754	0.28741

The best MSE and RMSE values are given in Tables 7 and 8. These tables summarize the results of Tables 1–6. According to Tables 7 and 8, while the lowest MSE and RMSE values for Movielens 100K dataset are obtained creating MLP using gradient learning rule with 3 hidden layers and applying genetic algorithms to this network, the lowest MSE and RMSE values for Movielens 1M dataset are obtained creating generalized feed-forward network using momentum learning rule with 3 hidden layers.

Table 7. The lowest MSE and RMSE values for Movielens 100K dataset for our study.

100K dataset		
Method	MSE	RMSE
MLP+Genetic algorithm (3 hidden layers, gradient)	0.24838	0.49838

Table 8. The lowest MSE and RMSE values for Movielens 1M dataset for our study.

1M dataset		
Method	MSE	RMSE
Generalized feed-forward network (3 hidden layers, momentum)	0.21458	0.46322

While overall training accuracy values of the hybrid system for 100K dataset are 0.24001, 0.49132, and 0.270110 for MLP, generalized feed-forward-network and CANFIS; these values are 0.24210, 0.25246, and 0.31849 for 1M dataset, respectively.

t-test one-tailed and two-tailed results of Movielens 100K and 1M datasets are given in Table 9. When we focus on this table to analyze one-tailed results of Movielens 100K dataset, we observe that t value is smaller than critical one-tail value. As a result, it can be said that difference between the groups is not significant. When we look at the P value of this dataset, we see the value of 0.14477. It means that there is only a 14% probability that the results from an experiment happened by chance.

When we look at this table to analyze one-tailed result of Movielens 1M dataset, we see that t value is bigger than critical one-tail value. As a result, it can be said that difference between the groups is statistically significant. When we look at the P value of this dataset, we see the value of 0.02557. It means that there is only a 2% probability that the results from an experiment happened by chance. As a conclusion, we can comment that there is more improvement after optimization when Movilens 1M dataset is used. Results of two-tailed t-tests are similar to one-tailed results for two datasets.

Table 9. t-Test one-tailed and two-tailed results assuming equal variances for Movielens 100K and 1M datasets.

	t Stat	P(T<=t) One-Tail/Two-Tail	t Critical One-Tail/Two-Tail
100K	1.07367	0.14577/0.29153	1.69726/2.0061
1M	2.03150	0.02557/0.04812	1.69726/2.0061

6. Comparison of our study with others

Pradeep and Bhaskar [47] use Movielens 100K dataset and observe that SVD approach has less RMSE. Chen and others [48] use Movielens 100K and Movielens 1M datasets and propose UCEC&CF algorithm. Katarya and Verma [49] use Movielens 100K dataset and propose K-means Cuckoo algorithm. Mohapatra et al. [50] use Movielens 100K dataset and propose a recommender system which use k-means clustering and cuckoo search optimization algorithm. Jaia Priyankka et al. [51] use Movielens 100K dataset and propose PSO-KM-FCM method. Comparison of RMSE values for Movielens 100K dataset is given in Figure 4.

Siddiquee et al. [13] focus on improvement in recommendation system when fuzzy logic is used and they use Movielens 1M dataset. Sanandaj and Alizadeh [37] present a hybrid recommender systems based on CF and content-based filtering. They use Movielens 1M dataset. Comparisons of RMSE values for Movielens 1M dataset are given in Figure 5.

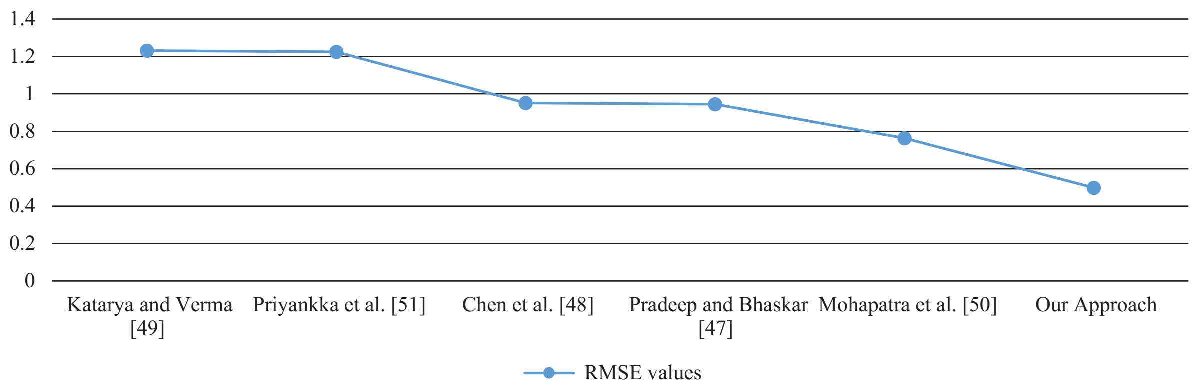

Figure 4. Comparison of RMSE values with different studies for Movielens 100K dataset.

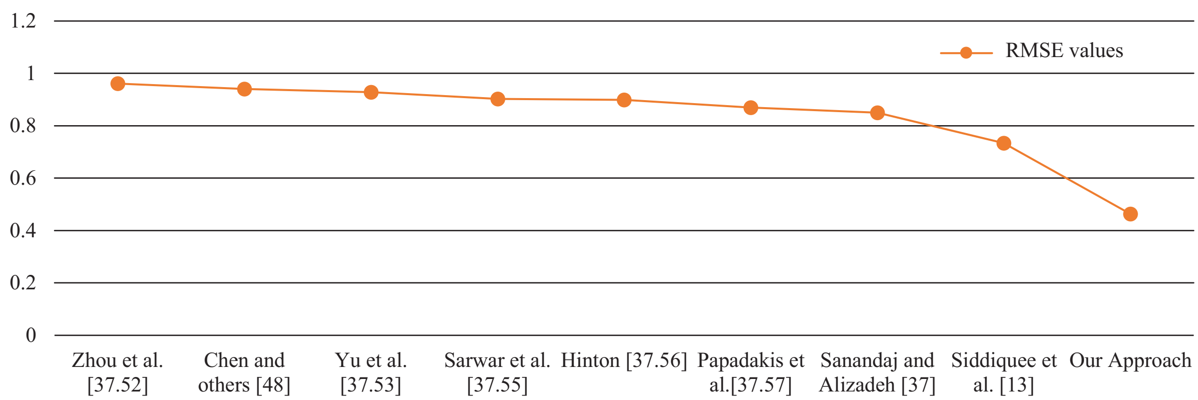

Figure 5. Comparison of RMSE values with different studies for Movielens 1M dataset.

7. Conclusion

Recommender systems are information filtering systems that aim to predict ratings of users to an item and recommend relevant and sensible items to the users according to their interests in a short time. These systems are used in a variety of sites such as e-commerce, business-oriented social networking, and online video community.

In this study, prediction performances of recommender systems were improved using ANN and genetic algorithms. After that, prediction performance of recommender systems were compared with the prediction performance of neural networks and neural networks which were combined with genetic algorithms. As a neural network; MLP, generalized feed-forward, and CANFIS were used. Firstly, predictions were carried out using a neural network-based prediction system. After that, an evolutionary neural network-based prediction system was proposed. Genetic algorithms improved the prediction performance of the CANFIS, MLP, and generalized feed-forward network algorithms since we aimed to minimize the prediction error. In addition, input parameters (i.e. which inputs to choose for training) as well as number of processing elements (i.e. number of neurons in layers) were also optimized to reduce the prediction error. As it is shown in Section 6, prediction performances of the proposed systems are better than those of a lot of studies. We get smaller MSE and RMSE values.

In the future, it is planned to propose a recommender system using fuzzy genetic algorithms with the aim of improving prediction accuracy performance of the system and handling sparsity and cold start problems. This proposed system will be evaluated using different datasets and performance evaluating metrics.

References

[1] Xiang B, Zhang Z, Dong H, Wu Q, Hu L. Research of mobile recommendation system based on hybrid recommendation technology. In: 3rd International Conference on Consumer Electronics, Communications and Networks; Xianning, China; 2013. pp. 508-512.

[2] Guo-xia W, He-ping L. Summary of personalized recommendation system. Journal of Computer Engineering and Applications 2012; 3: 66-76.

[3] Hirakawa G, Sato G, Hisazumi K, Shibata Y. Data gathering system for recommender system in tourism. In: 18th International Conference on Network-Based Information Systems; Taipei, Taiwan; 2015. pp. 521-525.

[4] Jain S, Grover A, Thakur PS, Choudhary SK. Trends, problems and solutions of recommender system. In: International Conference on Computing, Communication and Automation; Noida, India; 2015. pp. 955-958.

[5] Ricci F, Rokach L, Shapira B. Introduction to recommender systems handbook. In: Ricci F, Rokach L, Shapira B, Kantor PB (editors). Recommender Systems Handbook. Boston, US: Springer, 2011, pp. 1-35.

[6] Jones MT. Introduction to approaches and algorithms of recommender system. IBM Corporation 2013; 1-7.

[7] Peerzade SS. Web service recommendation using pearson correlation coefficient based collaborative filtering. In: International Conference on Energy, Communication, Data Analytics and Soft Computing; Chennai, India; 2017. pp. 2920-2924.

[8] Liu G, Wu X. Using collaborative filtering algorithms combined with Doc2Vec for movie recommendation. In: IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference; Chengdu, China; 2019. pp. 1461-1464.

[9] Mukherjee D, Banerjee S, Bhattacharya S, Misra P. A context-aware recommendation system considering both user preferences and learned behavior. In: 7th International Conference on IT in Asia; Kuching, Sarawak, Malaysia; 2011. pp.1-7.

[10] Dey AK. Understanding and using context. Personal and Ubiquitous Computing 2001; 5: 4-7.

[11] Inzunza S, Juárez-Ramírez R. Building context-aware recommendation systems: a software engineering point of view. In: 4th International Conference in Software Engineering Research and Innovation; Puebla, Mexico; 2016. pp. 175-184.

[12] Codina V, Mena J, Oliva L. Context-aware user modeling strategies for journey plan recommendation. In: International Conference on User Modeling, Adaptation, and Personalization; Dublin, Ireland; 2015. pp. 68-79.

[13] Siddiquee MMR, Haider N, Rahman RM. A fuzzy based recommendation system with collaborative filtering. In: The 8th International Conference on Software, Knowledge, Information Management and Applications; Dhaka, Bangladesh; 2014. pp. 1-8.

[14] Verma SK, Mittal N, Agarwal B. Hybrid recommender system based on fuzzy clustering and collaborative filtering. In: 4th International Conference on Computer and Communication Technology; Allahabad, India; 2013. pp. 116-120.

[15] Jeon T, Cho J, Lee S, Baek G, Kim S. A movie rating prediction system of user propensity analysis based on collaborative filtering and fuzzy system. In: IEEE International Conference on Fuzzy Systems; Jeju, Jorea; 2009. pp. 507-511.

[16] Yigit Sert S, Ar Y, Bostanci E. Evolutionary approaches for weight optimization in collaborative filtering-based recommender systems. Turkish Journal of Electrical Engineering & Computer Sciences 2019; 27: 2121-2136.

[17] Parvin H, Moradi P, Esmaeili S. A collaborative filtering method based on genetic algorithm and trust statements. In: 6th Iranian Joint Congress on Fuzzy and Intelligent Systems; Kerman, Iran; 2018. pp. 13-16.

[18] Alhijawi B, Kilani Y. Using genetic algorithms for measuring the similarity values between users in collaborative filtering recommender systems. In: IEEE/ACIS 15th International Conference on Computer and Information Science; Okayama, Japan; 2016. pp. 1-6.

[19] Shou-Qiang L, Ming Q. Research and design of hybrid collaborative filtering algorithm scalability reform based on genetic algorithm optimization. In: 6th International Conference on Digital Home; Guangzhou, China; 2016. pp. 175-179.

[20] Bostanci E, Ar Y. A genetic algorithm solution to the collaborative filtering problem. Expert Systems with Applications 2016; 61: 122-128.

[21] Kim HT, Kim E, Lee JL, Ahn CW. A recommender system based on genetic algorithm for music data. In: 2nd International Conference on Computer Engineering and Technology; Chengdu, China; 2010. pp. 414-417.

[22] Inoue M, Takenouchi H, Tokumaru M. Music recommendation system improvement using distributed genetic algorithm. In: Joint 8th International Conference on Soft Computing and Intelligent Systems and 17th International Symposium on Advanced Intelligent Systems; Sapporo, Japan; 2016. pp. 627-630.

[23] Soliman THA, El-Moemen Mohamed SA, Sewisy AA. Developing a mobile location-based collaborative recommender system for geographic information systems applications. In: Tenth International Conference on Computer Engineering & Systems; Cairo, Egypt; 2015. pp. 267-273.

[24] Hassan M, Hamada M. Improving prediction accuracy of multi-criteria recommender systems using adaptive genetic algorithms. In: Intelligent Systems Conference; London, UK; 2017. pp. 326-330.

[25] Hamada M, Abdulsalam LO, Hassan M. Adaptive genetic algorithm for improving prediction accuracy of a multi-criteria recommender system. In: IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip; Hanoi, Vietnam; 2018. pp. 79-86.

[26] Agrawal S, Jain P. An improved approach for movie recommendation system. In: International Conference on IoT in Social, Mobile, Analytics and Cloud; Palladam, India; 2017. pp. 336-342.

[27] Hosseinpourpia M, Oskoei MA. Genetic algorithm based parameter estimation for multi-faceted trust model of recommender systems. In: 5th Iranian Joint Congress on Fuzzy and Intelligent Systems; Qazvin, Iran; 2017. pp. 160-165.

[28] Janjarassuk U, Puengrusme S. Product recommendation based on genetic algorithm. In: 5th International Conference on Engineering, Applied Sciences and Technology; Luang Prabang, Laos; 2019. pp. 1-4.

[29] Ciaramella A, Cimino MGCA, Lazzerini B, Marcelloni F. Using context history to personalize a resource recommender via a genetic algorithm. In: 10th International Conference on Intelligent Systems Design and Applications; Cairo, Egypt; 2010. pp. 965-970.

[30] Devi MKK, Samy RT, Kumar SV, Venkatesh P. Probabilistic neural network approach to alleviate sparsity and cold start problems in collaborative recommender systems. In: IEEE International Conference on Computational Intelligence and Computing Research; Coimbatore, India; 2010. pp. 1-4.

[31] Hassan M, Hamada M. Performance analysis of neural networks-based multi-criteria recommender systems. In: 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering; Yogyakarta, Indonesia; 2017. pp. 490-494.

[32] Chakrabarti N, Das S. Neural networks and collaborative filtering. In: IEEE International Conference on System, Computation, Automation and Networking; Pondicherry, India; 2019. pp. 1-4.

[33] Da'u A, Salim N. Sentiment-aware deep recommender system with neural attention networks. IEEE Access 2019; 7: 45472-45484.

[34] Gong SJ, Ye HW. An item based collaborative filtering using back propagation neural networks prediction. In: International Conference on Industrial and Information Systems; Haikou, China; 2009. pp. 146-148.

[35] Mai J, Fan Y, Shen Y. A neural networks-based clustering collaborative filtering algorithm in electronic-commerce recommendation system. In: International Conference on Web Information Systems and Mining; Shanghai, China; 2009. pp. 616-619.

[36] Almaghrabi M, Chetty G. A deep learning based collaborative neural network framework for recommender system. In: International Conference on Machine Learning and Data Engineering; Sydney, Australia; 2018. pp. 121-127.

[37] Sanandaj DD, Alizadeh SH. A hybrid recommender system using multi layer perceptron neural network. In: 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium; Qazvin, Iran; 2018. pp. 7-13.

[38] Guha DR, Patra SK. Cochannel interference minimization using wilcoxon multilayer perceptron neural network. In: International Conference on Recent Trends in Information, Telecommunication and Computing; Kochi, Kerala, India; 2010. pp. 145-149.

[39] Gibson GJ, Siu S, Cowan CFN. Application of multilayer perceptrons as adaptive channel equalizer. Adaptive Systems in Control and Signal Processing 1989; 23: 573-578.

[40] Meyer M, Pfeiffer G. Multilayer perceptron based decision feedback equalizers for channels with intersymbol interference. IEEE Transactions 1993; 140: 420-424.

[41] Ibnkahla M, Yuan J, Boutros R. Neural networks for transmission over nonlinear channels. In: Ibnkahla M (editor). Signal Processing for Mobile Communications Handbook. Boca Raton, Florida: CRC Press LLC, 2005, pp. 1-28.

[42] Jadhav S, Nalbalwar S, Ghatol A. Performance evaluation of generalized feedforward neural network based electrocardiogram arrhythmia classifier. International Journal of Computer Science Issues 2012; 9: 379-384.

[43] Hemachandra S, Satyanarayana RVS. Co-active neuro-fuzzy inference system for prediction of electric load. International Journal of Electrical and Electronics Engineering Research 2013; 3: 217-222.

[44] Chiroma H, Abdulkareem S, Abubakar A, Zeki A, Gital AY et al. Co – active neuro-fuzzy inference systems model for predicting crude oil price based on organization for economic co-operation and development inventories. In: 3rd International Conference on Research and Innovation in Information Systems; Kuala Lumpur, Malaysia; 2013. pp. 232-235.

[45] Kwak SI, Choe G, Kim IS, Jo GH, Hwang CJ. A study of an modeling method of takagi–sugeno fuzzy system based on moving fuzzy reasoning and its application. CoRR 2015; 1511.02432: 1-24.

[46] Sayuti M, Khairani Sofyan D. Manufacturing optimization using tsukamoto fuzzy inference system method: A case study in block paving and solid concrete block industry. In: The 2nd International Mechanical and Industrial Engineering Conference; Malang, Indonesia; 2018. pp. 2-9.

[47] Pradeep IK, Bhaskar MJ. Comparative analysis of recommender systems and its enhancements. International Journal of Engineering & Technology 2018; 7: 304-310.

[48] Chen J, Zhao C, Uliji, Chen L. Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering 2019. Complex & Intelligent Systems 2020; 6: 147-156.

[49] Katarya R, Verma OP. An effective collaborative movie recommender system with cuckoo search. Egyptian Informatics Journal 2017; 18: 105-112.

[50] Mohapatra P, Mohapatra RK, Sandhibigraha B. Movie recommender system using improvised cuckoo search. International Journal of Innovative Technology and Exploring Engineering 2019; 8: 2869-2873.

[51] Priyankka RPJ, Arivalagan S, Sudhakar P. Performance analysis of effective optimization algorithm based collaborative movie recommender systems. International Journal of Scientific Research and Review 2019; 8: 511-527.

[52] Zhou Y, Wilkinson D, Schreiber R, Pan R. Large-scale parallel collaborative filtering for the netflix prize. In: International Conference on Algorithmic Applications in Management; Shanghai, China; 2008. pp.337-347.

[53] Yu HF, Hsieh CJ, Si S, Dhillon I. Scalable coordinate descent approaches to parallel matrix factorization for recommender systems. In: IEEE 12th international conference on data mining; Brussels, Belgium; 2012. pp. 765-774.

[54] Ekstrand MD, Riedl JT, Konstan JA. Collaborative filltering recommender systems. Foundations and trends in human-computer interaction 2011; 4: 81-173.

[55] Sarwar B, Karypis G, Konstan J, Riedl J. Item-based collaborative filtering recommendation algorithms. In: 10th international conference on world wide web; Hong Kong; 2001. pp. 285-295.

[56] Hinton G. A practical guide to training restricted Boltzmann machines. Toronto, Canada: UTML TR 2010–003, 2010.

[57] Papadakis H, Panagiotakis C, Fragopoulou P. A synthetic coordinate based recommender system. Expert Systems with Applications 2017; 79: 8-19.