Recommended System Optimization in Social Networks based on Cooperative Filter with Deep MVR Algorithm

Qiang Gao (14438120210041@sdmu.edu.cn)
Shandong University

Research Article

Keywords: Social Networks, Recommended System, Deep Learning, Reinforcement Learning, Modified Vector Rotational (MVR) algorithm

Posted Date: June 9th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1615797/v2

Recommended System Optimization in Social Networks based on Cooperative Filter with Deep MVR Algorithm

Qiang Gao

Shandong Management University, School of Labor Relations, Jian Shandong 250357, China

14438120210041@sdmu.edu.cn

Abstract

Today, social networks have become very popular due to their high usage in communicating with each other. But this popularity requires the development of backend to communicate with each other. Hence, a topic called identifying users is created by making recommendations or propositional systems, and so on, link prediction. The most important issue is the new users' social networks, so that they can offer suggestions. In this research, we tried to provide a system of recommendations for introducing new users to previous users and vice versa based on the principles of machine learning. The proposed method is that the data is entered into the program and then the keywords are extracted from them. Then a sampling between the data is performed based on the Pearson and Cronbach method. In the process of extraction operations along with diminishing dimensions, selection and finally extraction of the best features is done using the cooperative filter which named here based on deep learning- Modified Vector Rotational (MVR) algorithm and its operators. In the following, due to the lack of probabilistic and statistical training of the Deep and Reinforcement Learning with random structure that is used to select users and also to offer users with respect to the tastes of the extracted, there is an optimization algorithm for MVR consider the best features with training. In the following, a series of evaluation criteria have been used to ensure the proposed approach, indicating the appropriate results of the proposed method.

Keywords: Social Networks, Recommended System, Deep Learning, Reinforcement Learning, Modified Vector Rotational (MVR) algorithm

Introduction

The unprecedented growth of the new Internet technology in recent years has created many applications in the field of social networking. People can only face one problem, and that information and items are extremely infinite. Given the needs of today's virtual world and Internet technology, researchers are trying to research to improve and make it as easy as possible and provide new ways to solve problems and challenges [1].

One of the most important applications in social networks is the recommended systems. Offer systems are a special kind of information refinement systems that refine items from a large set of items and users based on which item is attractive to the user. This system is an approach that is presented to meet the challenges of a large and growing amount of information, and helps its user to approach their goal within a massive amount of information. The proposing systems try to guess the user's thinking and then identify and offer the closest and most suitable product to the user's taste. Also, a recommended system can prevent waste of time by displaying the list of items that are of interest to users. Many proposed bidding systems focus on items based on text information such as documents, website addresses, and text messaging [1].

The new user is one of the main problems and concerns of the system provider, which requires a large amount of user data to provide the exact proposal. While at the time a new user enters the system, no information is available to them in the system and the system information provider does not have a new users to compare with other users for the preparation of new user suggestions. At the same time, this issue is important because the new user, if not receiving attractive suggestions from the system, may lose the desire to reuse the system. It's hard to offer suggestions based on user interests, especially for newly logged-in users. In proposing systems, it is important to calculate the similarity of the product with the user and the user with the user. The proposer systems are divided into four categories based on the estimation of the rate and how the proposals are made [1]:

- ✓ Recommended-oriented content
- ✓ Coordinator (or subscriber)
- ✓ Combined refinement
- ✓ Knowledge-based approach

Literature Review

A lot of research done in recommender systems in social networks. In [2] they are focused on certainty. They provide a new confidence-driven approach with the name of integration by combining explicit neighbors identified by users in the system with the goal of improving the overall performance of the proposals and improving the data and the new user problem of refining the partnership. In particular, merging the neighboring rankings of an active user by averaging rankings on commonly ranked items according to the extent to which trusted neighbors are similar to those of users [3].

In [4], he has impliedly assured the user. Less research has been attempted to derive more user rankings from user interface angles. Extracting more user ratings can reduce the issues involved. The method of this paper is to better model user performance by using user behaviors (rankings) and social relationships (trusted friends) on social networks. It is intended to provide a reliable feature to achieve a positive and strong correlation with performance. The reliability of the trust network is mainly disseminated and can provide a recommender or suggestion system [5].

In [6], the present-day informed assurance methods have been ranked for an unobserved item by ranking users who are accessed through a trusted-reliant link in the target user trust network. So that a recommender system can be created. However, in this method it is not possible to apply the ranking of users in the opposite direction, they also have similar interests. In this paper, this issue has been considered and this possibility has been investigated by identifying and adding these users to existing methods at the time of estimating the rank for the target user.

In [7], a study has been conducted in a propositional system that ensures users' improved reassurance receives suggestions from items ranked by individuals in their confidence range, or even by those who are members of their social networking community are trusted as friend. Hence, by providing a little more subtle connectivity in the secure network, new users can instantly access a wide range of offers. In [8], new users are grouped into a specific group using effective techniques such as C4.5 and simple parsing algorithms as well as random selection. In this way, they have important characteristics (demographic data, such as age, gender, and occupation) to test the user for other users in the group that best matches the user. [9] have examined the extent to which the likelihood of users accepting a high quality item offer is acceptable and choosing neighbors based on their performance sharing with the target user. This method represents a kind of bid approach (based on user-centric diversity) that explicitly searches neighbors with the same functionality for the user, and uses these functions to predict the rankings of the user's item. This approach is based on the principle that a user-specific ranking is not equally applicable to other users as a customized suggestion input of an item [10].

In [11], they presented a reduction-based approach to the multi-dimensional advisor model that combined textual information such as time and location in the recommendation process. [12] provides an informative text system that predicts user preferences in different text positions depending on what similar users have done in similar cases and can be used to predict a user's priority relative to Current text to be used. In [13], they proposed a collaborative label refinement in which two models were created to aggregate labels in the stage of calculating the user's similarity and the stage of prediction. However, the system still encountered the problem of not reusing users from labels. In [14] used labels to find similar users to create a candidate label set for each user. Using Bayesian clustering, she then implemented the recommendation approach based on the same set of candidate labels.

In the method used in [15], three user matrices use the user's tag and user tag and item tag matrix to generate a general mechanism that aggregates the tags in the standard collaborative refinement model, which is based on collaborative refinement The user's matrix of the user's item should be expanded with the user's user tag matrix and expanded to the user-item matrix based on the item's matrix system with the label item matrix [16]. In [17] and [18] examined the use of CiteLike data for the purpose of recommending scientific articles. Three collaborative refinement algorithms (traditional method and user-based and item-based method) were examined and found that user-based collaborative refinement of the item-based collaborative refinement in the proposed systems was better.

Some of the work done was based on time information, such as user purchase time and supply time, which was intended to increase the accuracy of the recommendations, in which two ranking functions were proposed to calculate the time based information. Previously, a type of time information was considered including item delivery time and user's time of purchase, and the time difference between the two, and the results showed that such time information could accurately predict the suggestions in the proposed system based on collaborative refinement for characters Described in the mobile e-commerce environment. In the research, [19] and [20] have introduced a user-centered collaborative refinement algorithm. Because of the initial introduction of this algorithm by [20], in its research, the name of this algorithm is set out in the following charts: Resnick-UCF. In [21] an advisory and proposing system is presented as a comprehensive framework for proposing items, groups, friendships based on the same resources. The proposed method is a hybrid propositional system. Also, in [22], there is an item suggestion system on social networks that has been suggested to him based on user input data. This method is content-driven.

As three main studies, we can refer to articles [23], [24] and [25]. Watsall Mittal [23] analyzes user posts on social networks based on time analysis, video filter analysis, hashtack analysis on images, and analysis of image categorization. In fact, it has used several variables to analyze user posts in order to provide a reseller system based on user posts. In his research in [24], Yu has explored the different structures and features that a social network can provide for the design and development of advisory systems. The introduction of different evaluation criteria to recognize the accuracy and improvement of advocate systems is the most important part of this research. Dong Yan-cho [25] offers an optimal and innovative way to build advocate-based systems based on the number stored in the user's new user list, based on their tastes and their communications in social networks. The modeling of this research is based on the communication between the users. In [26], an artificial immune optimization algorithm was used to improve performance in suggestions in the social networking advisor system, resulting in 85.01%. Also, in [27], the optimal particle swarm algorithm is used to improve the evaluation results, especially the accuracy of previous algorithms, including genetics, which results in 85.00%.

As the newest and related article, in [28], a new model of deep learning combinational method used for recommended systems based on sentiment analysis. This article proposed convolutional-recurrent method as deep learning in hybrid mode, which is CNN-LSTM. This article proposed precise model with suitable parameters determination, but also have computational complexity with many outlier data. Another article proposed multi-recommended system based sentiment analysis in Algeria as cooperative filter which used SVM. This method and recommended system is limited to only Algeria's people interests but with suitable accuracy [29]. In [30], also presented a new kinds of recommended systems with sentiment analysis in hotels reviewing in online websites. This article used cooperative filter based on deep learning. The scalability of this model is too high, but there is no results for evaluation criteria. Also in [31], proposed a collaborative model for recommended systems based on sentiment analysis with acceptable accuracy.

Research Methodology

This research attempts to provide a recommender system to find friends based on user interactions. The proposed system in this research is a knowledge-based propositional system. This system is an approach that is presented to meet the challenges of a large and growing amount of information and helps its user reach their goal within a massive amount of information faster. These systems are asked by the user about the need to Products work and find out which product or person on the social network, according to the user's needs. Considering that the new user login is a challenge in these systems, as well as other challenges, such as the number of mutual interests among individuals, the interest in creating friendships, and so on in the proposer systems, the method should be intelligent to be able to solve all these challenges. Therefore, a clustering with the purpose of separating user information is used. The research method used is that by entering a new user into the social network, the user first enters the specifications as inputs. These specifications include his preferences Favorite music or products required by the user, the system should be able to provide him with the best and most precise suggestion. The system of identifying users and spoolers may be mistaken. Hence, training is involved in this clustering. Since the MVR algorithm is a randomized cuspidation method, it is therefore necessary to use a probabilistic teaching method to improve it and to select and extract the best features. Although the MVR

Algorithm has a slowdown in this section, it is used as a statistical probabilistic method as an optimization method. In order to guarantee the proposed method, a series of evaluation criteria are also used, including accuracy, sensitivity, feature rate, mean square error, and other items used in the base papers. To the overall output of work, the new user suggests a user who has already been in the system and the offer is based on the tastes and interests of the users.

Proposed Approach

The main goal in recommender systems is to provide a suggestion of items to the user. The proposed advisor system decides on the inputs given. In fact, the advisor system, by reading the data given to it, will try to put information in a cluster or in different categories, then it will offer a suggestion based on the inputs of the new users. Given the data input to the program, it is necessary to make a thorough analysis of it, in order to obtain the similarity of the data. In order to find users, there are many methods that the proposed approach in general is based on Deep and Reinforcement Learning algorithm training. Each user in the input data can be seen as his interest as a chromosome, and the difference between these vectors is to calculate and model the cues between the users and finally find the same users and suggest a new user. . In fact, the purpose of this research is to offer previous users a new user based on its tastes, as well as a new user's suggestion to previous users based on their tastes. To find a user similar to a new user, a relationship such as equation (1) is used.

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_{p \in P}(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}} \quad (1)$$

Given this relationship, the likelihood of similarity is obtained based on their average P score. But the reason for the subtraction of the vote is from the average, since it is not possible for each user to announce his vote. In general, the user p is the subset of the mean of the vote and $(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})$ is the vote or the users. Relation (3.1) is a relation called Pearson that is considered for similarity in a system considered in the Deep and Reinforcement Learning algorithm. But the main suggestion to predict the similarity between users in the system based on the reliability and reliability of a statistical data is the use of Cronbach, which is in the form of equation (2).

$$Prediction_{Cronbach}(a,b) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) \times (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)} \quad (2)$$

The research is based on the use of a knowledge-based advisory system. The knowledge-based bidirectional system works by asking the user about the need for products and find out which product or person in the social network is according to the user's needs. These questions have already been asked and put into the data, and it is necessary to make clustering between them using a training-based approach. Before the training of the Deep and Reinforcement Learning algorithm was introduced, the level of similarity of users was done using Pearson's relationship, and then the reliability and reliability of this similarity was measured based on the tones between users using Cronbach's. Then it is necessary to extract the importance of the data in the data set. It is necessary to use a probabilistic statistical method to find out the importance of a data (the taste of a new or former user). The preferred method is TF-IDF. Its relation is in the form of equation (3).

$$tf(t,d) = 0.5 + \frac{0.5 \times f(t,d)}{max\{f(w,d), w \in d\}} \quad (3)$$

Given the relationship (3), tf represents the raw frequency of a word or character in the data, and t is equivalent to words or characters, and in a document set, these words or characters are stored, where d is the number of times, also repeats a word or character. Also w is the weight of these words or characters. The equation (3) for TF is TF-IDF. The second part of the TF-IDF method is the same IDF, which shows how much information about a document, D, is provided by a word or character, which is based on its repetition in the document or N, whose relation is also represented as equation (4).

$$idf(t,D) = log \frac{N}{|d \in D \quad , \quad t \in d|} \tag{4}$$

The multiplication between the TF section in the IDF provides a new relationship that identifies the importance or weight of a word or character in the data, which is in the form of equation (5).

$$td.idf(t,d,D) = tf(t,d) \times idf(t,D) \tag{5}$$

Now it is necessary to teach the system. Based on the fact that there is an uncertainty between the similarity segment in the input data as well as the suggestion of users to each other in the advisor system, the use of evolutionary structures such as Deep and Reinforcement Learning algorithm seems to be appropriate. In the mathematical description, the performance function evaluates the appropriate items in the proposal as $i \in Items$ for each user $u \in Users$, which is defined as the equation (6), which is considered as a chromosome.

$$R: Users \times Items \rightarrow R_0 \tag{6}$$

According to (6), Users will display a set of all users, or chromosomes and items, a set of all possible items that can be offered to users or users, and as genes. R_0, as normal, displays a binary representation of the input data between users. The term "interruption" is the same information that is logged in the system at different time intervals. A more functional function is a subset of the $Users \times Items$ space, because, as noted earlier, a user may have a subset of his tastes arbitrarily owned by him and others have such there is no hierarchy and it has more value or a higher vote. The recommender systems need to cluster and then predict the function function $R(u,i)$ for the user u from item i, and then predict the item with respect to the maximization of the function function. This description will be mathematically motivated as equation (7).

$$\forall u \in Users, i = arg\ max_{i \in Items} R(u,i) \tag{7}$$

In equation (7), $arg\ max$ is the equivalent of the maximal elements in the recommended system. Applying this procedure repeatedly will allow the advisor system to provide further advice. In the multi-criteria approach, which is also knowledge-based, the voting function or value for the purpose of considering votes or values is modeled in equation (8).

$$Users \times Items \rightarrow R_0 \times R_1 \times \ldots \times R_k \tag{8}$$

In equation (8), R_0 shows the total number of votes or values given by the user to the item i and $R_j (j \in \{1, \ldots, k\})$ for each criterion in the field of knowledge-based method. Then, an initial training using the MVR algorithm is presented randomly, which is used to extract the features. The MVR algorithm is performed in six steps:

1- Creating a random population and evaluating it (any answer generated at the moment is calculated by its cost function)
2- Choosing parents and combining them to create a crowd of children (such as combining the ideas of two individuals and obtaining a better idea).
3- Selecting members of the population to make a jump and create a mutated population (such as never having to combine two good cars in an automobile factory of an aircraft and make sure there is a jump in work)

4- Integration of the main population, children and mutations and the creation of a new major population (in fact, we compiled this new population from three sources)
5- If the termination conditions are not met, we repeat step 2.

Operation extraction features are based on MVR optimization algorithm. In fact, the process extraction step involves reducing dimensions, extracting and extracting the attribute. The data obtained from training with the Deep and Reinforcement Learning algorithm are introduced and trained in a MVR optimization algorithm. To find the best features of the data trained by the Deep and Reinforcement Learning algorithm, it is interesting to note that we use artificial agents with the following properties:

1- Deep Learning will be randomly assigned to trained data.
2- Each Techniques of Deep Learning (Specially Convolution Neural Network or CNN) will select the trained data in a possible approach. The probability of choosing the next trained data, the function of the distance to the other trained data is the corresponding value in the trained data.
3- Each convolve operator of Deep Learning must be allowed to move to trained data and select them, they must move to a data point that they did not pass through in the past. This condition is controlled using the Banned List.
4- When a route identification of data is formed as the best feature, the Reinforcement learning is left over and the Reinforcement part transmits some learning methods while passing through the data. Reinforcement Learning moves at a time interval $(t, t+1)$, so for each repetition of the algorithm, m moves will occur.
5- At a time interval $(t, t+1)$, some Reinforcement Learning evaporate. Once each Reinforcement Learning has created its own data and feature, Reinforcement Learning will be identified again for them.

In the first step, the m Reinforcement Learning is initially created with memory. These ants will randomly be placed on n data. On each data there is some initial Learning. Secondly, in order to obtain the initial solutions, the following steps are implemented in parallel. It is now necessary to identify the educational inputs of the Deep and Reinforcement Learning algorithm. Votes are r_{ij}, which is obtained by any U_i user for every I_j item item. If the user U_i does not vote for any advice and does not select it from I_j, then $r_{ij} = N/A$ will be. This is done manually and based on the observations of the article. In the table (1), we can observe the educational variables of the MVR algorithm.

Table (1) variables of Deep and Reinforcement Learning algorithm in training

Data range	Main data for training as input
SI	(4.00, 5.00, 5.00)
MI	(3.00, 4.00, 5.00)
I	(2.00, 3.00, 4.00)
LI	(1.00, 2.00, 3.00)
NI	(1.00, 1.00, 2.00)
N/A	-----

The proposed pseudo code is as follows:

// require n.m matrix
// which n is number of items and m is the number of interest
Require interest, n
//calculate similarity for all pairs of interest in data

```
For interests I from 1 to n, do
        For interest j from 1 to n, do
                Chromo_01 ← number of cells where interest i is 0 and interest j is 1
                Chromo_10 ← number of cells where interest i is 1 and interest j is 0
                Chromo_11 ← number of cells where interest i is 1 and interest j is 1
                If (Chromo_01+ Chromo_10+ Chromo_11) is 0 then
                        Sim (interest i, interest j) ← 0
                        Apply Crossover
                        Apply Mutation
                        Reproduction
                Else
                        Sim (interest I, interest j) ← Chromo_11 | (Chromo_01+ Chromo_10+ Chromo_11)
                End if
        End for
        Find the Best Features with Deep and Reinforcement Learning
        Ants Routes on Data
        Dimension Reduction with operation by Deep and Reinforcement Learning
        Feature Selection
        Feature Extraction
End for
Return Sim for all pairs of (interest i, interest j)
Recommend new data of i and j
```

Simulation and Results

Using a data set of a social network whose data is available in the https://snap.stanford.edu/data/#socnets link is considered. Initially, data is entered into the program. The description of the data and the proposed method in the previous sections is explained. Initially, the normalization of data is done and these data are normalized into the program. Then the data are divided into two categories: educational data and test (test). First, a sampling is done using the Pearson method, then Cronbach and finally the TFIDF. The goal is to use the Deep and Reinforcement Learning algorithm, to initial clustering, and to put each user with tastes in the family to see if they are other than those that are extracted with iodine. In the early clustering with the Deep and Reinforcement Learning algorithm, the identification of all characters, including those listed in Table (2), is considered vital.

Table (2) Characters and signs required for initial training

,	}	6
.	|	7
<	(	8
>	)	9
	-	0
/	_	!
\	=	~
?	+	@
'	-	#
"	*	$
;	1	%
:	2	^
]	3	&
[	4	

{	5	

The letters of the alphabet are also called from a small to a small z with a large to large z. Then initial clustering is performed and features are extracted. It should be noted that the initial population of the MVR algorithm is 100 chromosomes, the amount of the compound 2, the mutation amount is 0.02. Hayman has been randomly assigned 150 rounds in the initial training data routine with Quantum MVR. Then, the operation takes over all three input datasets, and clustering starts with the Reinforcement Learning. The number of training data is 337 files and the number of test data is also 337. In fact, 50% of the data is given as training and 50% of the data is considered as a test and clustering. The number of rounds of Deep and Reinforcement Learning optimization algorithm is also 100 rounds. Its teaching method is also probabilistic-statistical. Also, the threshold value of the colony optimization algorithm is estimated at 0.5, and its initial population includes ants including 100 ants that move with the amount of Deep and Reinforcement Learning 0.9 in trained data. A graph of a multimillion diagram is used to determine the accuracy of the similarity detection with the proposed approach, which indicates that the accuracy of the number of training and test data is increasing in the diagnosis of similarity in the data to provide suggestions. . This item is shown in Figure 1.

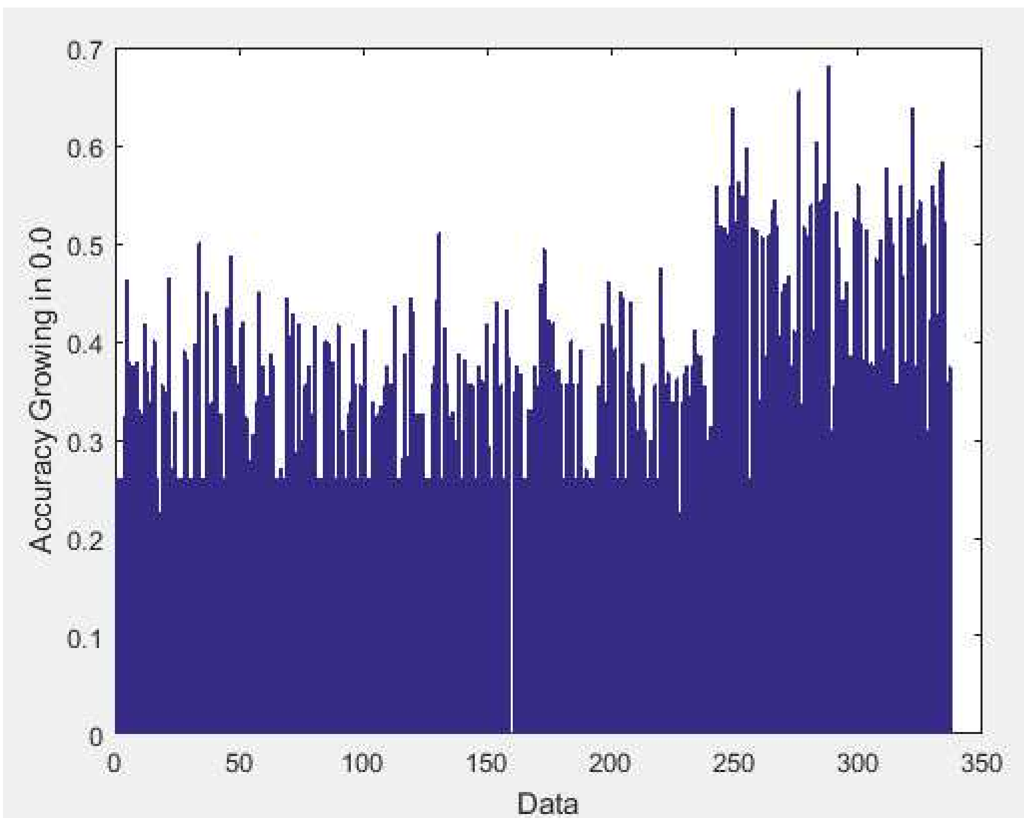

Figure 1. Shows the similarity of users and the degree of accuracy in decimal in the number of training and test data

Then the Deep and Reinforcement Learning algorithm enters action with the number of repetitions 150. It should be noted that this research is a statistical and probabilistic modeling and is trying to provide a method that can be based on it, only improve the evaluation criteria, such as previous research, and is not a system in which proposals should be displayed. In Figure 2, the result of training is shown using the Deep and Reinforcement Learning algorithm based training on the input data.

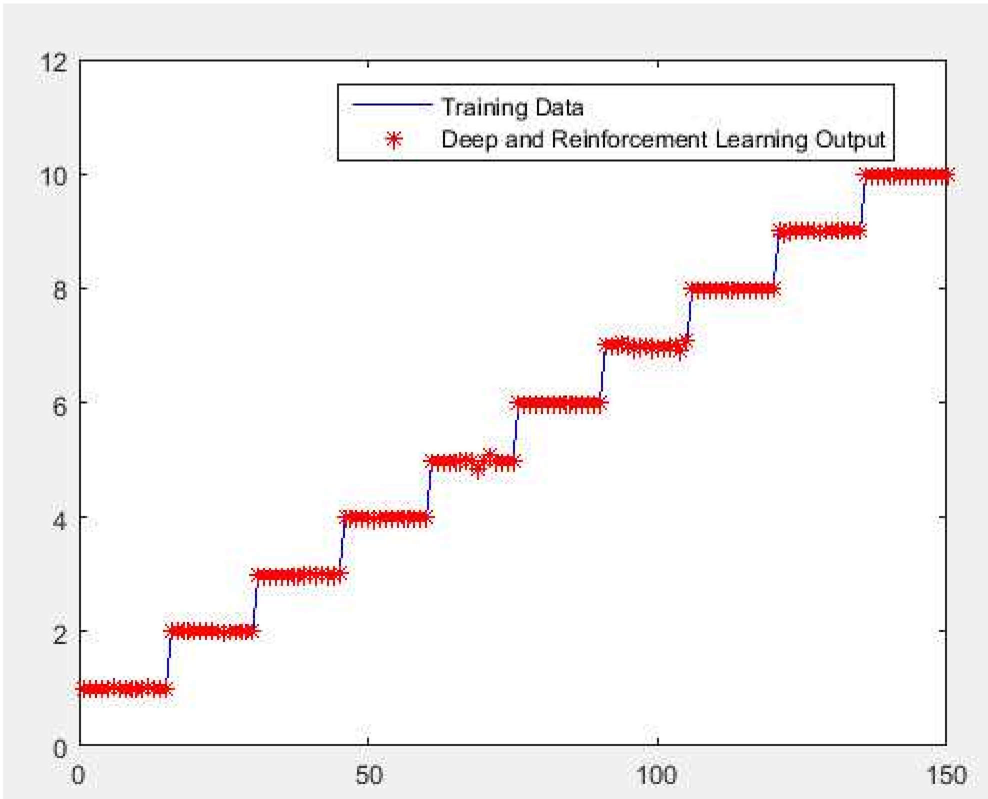

Figure 2. The result of training using Deep and Reinforcement Learning algorithm

In the x-axis, 150 data sets were used for final training with Deep and Reinforcement Learning algorithm, and its 11 features in the y axis based on existing training from previous methods including normalization and then Pearson-Cronbach-based sampling and finally extraction of attribute is derived with the Deep and Reinforcement Learning. It has been shown that the combined output of the Deep and Reinforcement Learning algorithm and the optimization algorithm of the Deep and Reinforcement Learning in different educational stages has been performed. The same operation has also been performed using the Deep and Reinforcement Learning for the data test (test) stage, which shows that the convergence is performed correctly, the results of which are shown in Figure 3.

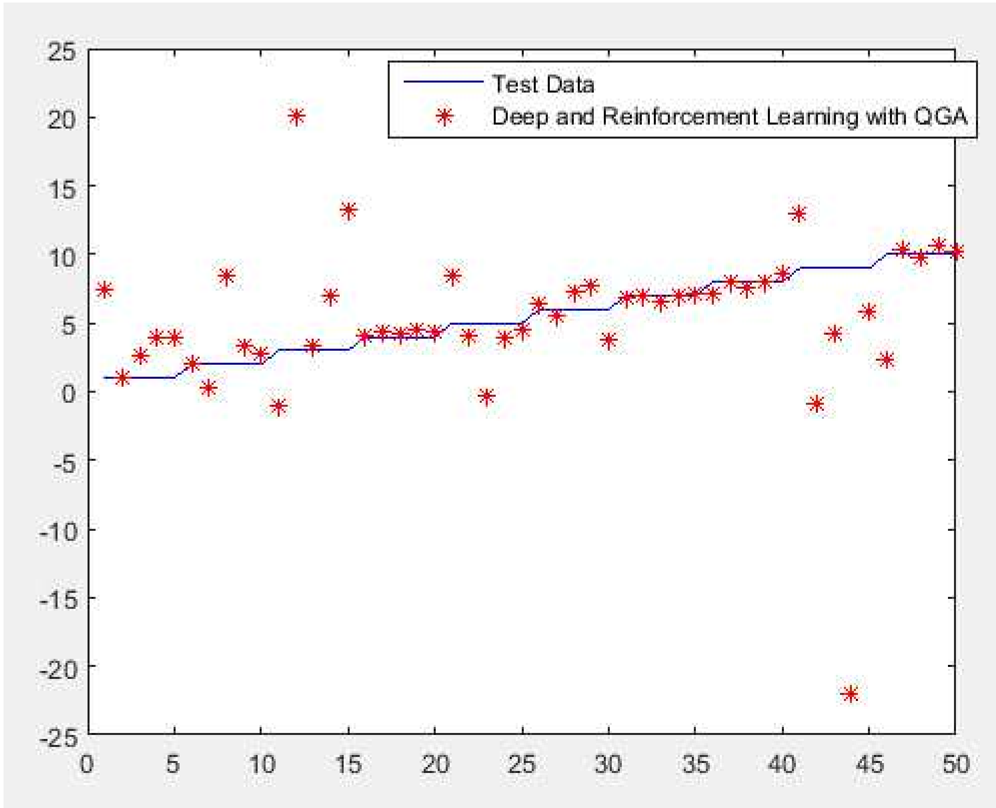

Figure 3. Convergence result at the time of training data testing by an Deep and Reinforcement Learning

The convergence purpose is that the test, which is indicated by the blue graph, is increasing, which can lead to a proper regression that is not considered in the study. Also, a large number of anticipated outputs of the Deep and Reinforcement Learning, which offers users (including offering to a new user or offering to old users), is close to the testing stage and is available around it. Axis x shows the number of test data. The Y axis also displays the features extracted and tested. Data derived from the outsourcing colony optimization algorithm away from the districts are identified as pertinent data, and there are no neighborhoods around the test axis. It is necessary to validate the proposed approach, DRLQGA, in the recommender system in a number of similar ways. For this purpose, Figure 4., shows the comparison of the validation of the proposed approach in the training phase and the test with other similar methods.

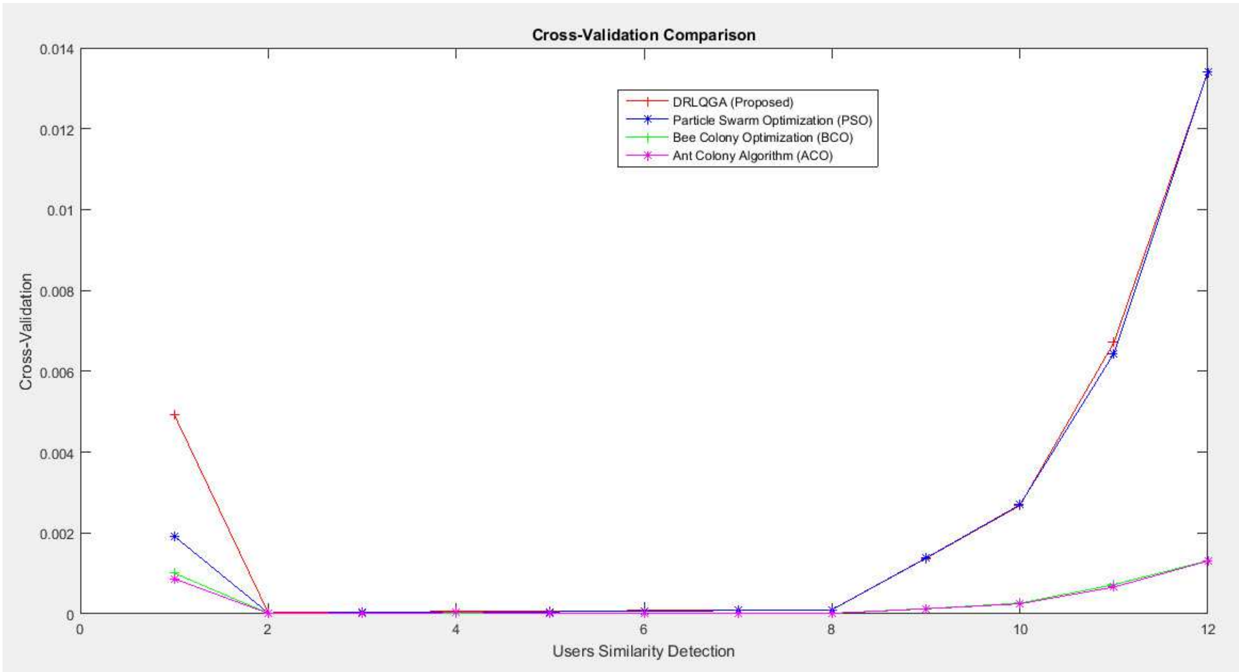

Figure 4. Comparing the validation of the suggested approach in the training phase and the test with other similar methods

According to Figure 4-4, the proposed approach, DRLQGA, has a performance advantage over optimal particle swarm algorithm, artificial bee algorithm and the only used Deep and Reinforcement Learning. Of course, optimum particle swarm algorithm has a fairly convergent diagram to the proposed approach in user validation and matching, but it has also been shown that DRLQGA works better. Finally, the results of the evaluation of the proposed method are derived based on the proposed method of this research. The proposed recommender system has the evaluation results as described in Table (3).

Table (3) The results of the evaluation of the proposed method

Specificity (%)	Sensitivity (%)	Accuracy (%)	SNR (dB)	PSNR (dB)	MSE
85	90	86.053	75.20	48.88	0.8410

In the following, a comparison in terms of the precision criterion in terms of percentage between the three basic articles of this study is [23], [24], [25], [26] and [27] which are shown in Table (4) and Figure 5.

Table (4) Comparison of proposed approach in terms of accuracy with previous methods

References	Accuracy (%)
Mittal, Vatsala, et al., 2017 [23]	84.21 %
Yu, Wei, and Li, Shijun, 2018 [24]	85.12 %
Guo, Dongyan, et al., 2017 [25]	83.94 %
Duma, Mlungisi, and Twala, Bhekisipho, 2018 [26]	85.01 %

Ujjin, Supiya, and Bentley, Peter J., 2003 [27]	85.00 %
Proposed Method	86.053 %

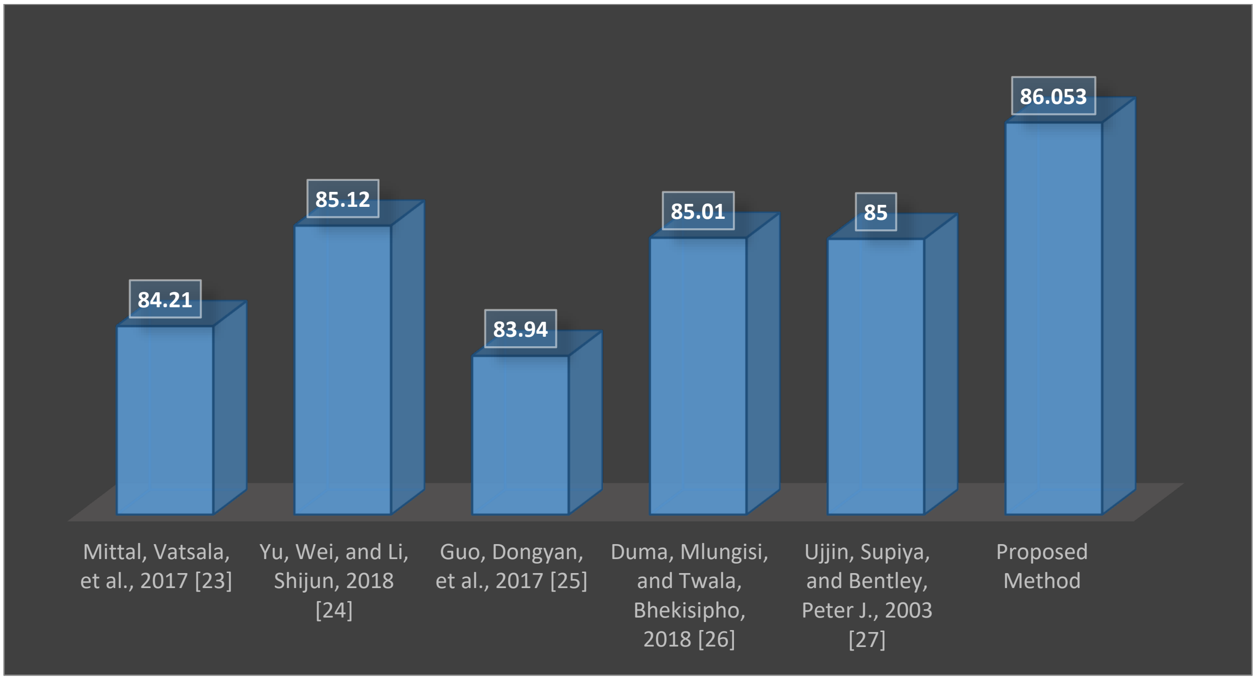

Figure 5. Comparison of proposed approach in terms of precision with previous methods

According to the results, the user recommendation system can be assured by the accuracy, sensitivity and rate of appropriate features in percentages. In the next chapter, the proposed method is discussed.

Conclusion

In this research, we tried to provide a system of recommendations for introducing new users to previous users and vice versa based on the principles of machine learning. The proposed method is that the data is entered into the program and then the keywords are extracted from them. Then a sampling between the data is performed based on the Pearson and Cronbach method. In the process of extraction operations along with diminishing dimensions, selection and finally extraction of the best features is done using the Deep and Reinforcement Learning algorithm and its operators. In the following, due to the lack of probabilistic and statistical training of the Deep and Reinforcement Learning algorithm and random structure that is used to select users and also to offer users with respect to the tastes of the extracted, there is an optimization algorithm for MVR algorithm taking into account the best features with training. , Is used. In the following, a series of evaluation criteria have been used to ensure the proposed approach, indicating the appropriate results of the proposed method.

References

[1] Yardi, Sarita, Romero, Daniel, Schoenebeck, Grant, and Boyd, Danah. 2010. Detecting Spam in Twitter Network. Free Reviewed Journal of Internet, Vol. 15, No. 1.

[2] Bobadilla, J., Serradilla, F., and Bernal, J. 2010. A new collaborative filtering metric that improves the behavior of recommender systems. Knowledge-Based Systems, Vol. 23, No. 8, pp. 520-528.

[3] Isaac, A., Meij, L., Schlobach, S., and Wang, S. 2007. An Empirical Study of Instance-Based Ontology Matching, in The Semantic Web. Vol. 4825, K. Aberer, K.-S. Choi, N. Noy, D. Allemang, K.-I. Lee, L. Nixon, et al., Eds., ed: Springer Berlin Heidelberg, pp. 253-266.

[4] Guo, X., and Lu, J. 2007. Intelligent e-government services with personalized recommendation techniques: Research Articles, Int. J. Intell. Syst., Vol. 22, pp. 401-417.

[5] O'Neill, E. T., Mai Chan, L. 2003. FAST (Faceted Application of Subject Terminology): a simplified vocabulary based on the Library of Congress Subject Headings, IFLA Journal, Vol. 29, pp. 336-342.

[6] Ruiz-Montiel, M., and Aldana-Montes, J. 2009. Semantically Enhanced Recommender Systems, in On the Move to Meaningful Internet Systems: OTM 2009 Workshops. Vol. 5872, R. Meersman, P. Herrero, and T. Dillon, Eds., ed: Springer Berlin Heidelberg, pp. 604-609.

[7] Blanco-Fernández, Y., Pazos-Arias, J. J., Gil-Solla, A., Ramos-Cabrer, M., López-Nores, M., and García-Duque, J. 2008. Exploiting synergies between semantic reasoning and personalization strategies in intelligent recommender systems: A case study, Journal of Systems and Software, Vol. 81, No. 12, pp. 2371-2385.

[8] Porcel, C., Tejeda-Lorente, A., Mart, M. A., Nez, and Herrera-Viedma, E. 2012. A hybrid recommender system for the selective dissemination of research resources in a Technology Transfer Office, Inf. Sci., Vol. 184, pp. 1-19.

[9] Massa, P., and Avesani. 2008. P. Trust-aware Recommender Systems ACM library.

[10] Abasi, M., and Ghaznavi-Ghoushchi, M. B. 2012. Low-Power Themes Classifier (LPTC): A Human-Expert-Based Approach for Classification of Scientific Papers/Theses with Low-Power Theme, Intelligent Information Management, Vol. 4, pp. 364-382.

[11] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. ACM Transactions on Information Systems (TOIS), 23, 103–145.

[12] Gil, A. F. Garc. 2003. E-commerce recommenders: powerful tools for E-business Crossroads. Vol. 10, pp. 6-6.

[13] Buder, J., and Schwind, C. 2012. Learning with personalized recommender systems: A psychological view, Computers in Human Behavior, Vol. 28, pp. 207-216.

[14] Hill, W., Stead, L., Rosenstein, M., and Furnas, G. 1995. Recommending and Evaluating Choices in a Virtual Community of Use," roc. Conf. Human Factors in Computing Systems.

[15] Tso-Sutter, K. H. L., Marinho,L. B., and Schmidt-Thieme, L. 2008. Tag-aware recommender systems by fusion of collaborative filtering algorithms. presented at the Proceedings of the 2008 ACM symposium on Applied computing, Fortaleza, Ceara, Brazil.

[16] Belkin, N. J., and Croft, W. B. 1992. Information filtering and information retrieval: two sides of the same coin?, Commun. ACM, Vol. 35, pp. 29-38.

[17] Bogers, T., and Van Den Bosch. 2008. A. Recommending scientific articles using Citeulike. In: Proceedings of the 2008 ACM conference on Recommender systems, pp. 287–290. New York, NY, USA: ACM.

[18] Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. 2010. Recommender Systems: An Introduction: Cambridge Univ Pr.

[19] Sarwar, B., Karypis, G. Konstan, J., and Riedl, J. 2000. Analysis of recommendation algorithms for e-commerce, presented at the Proceedings of the 2nd ACM conference on Electronic commerce, Minneapolis, Minnesota, United States.

[20] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. 1994. GroupLens: an open architecture for collaborative filtering of netnews. presented at the Proceedings of the 1994 ACM conference on Computer supported cooperative work.

[21] Chen, Li, Zeng, Wei, and Yuan, Quan. 2013. A unified framework for recommending items, groups and friends in social media environment via mutual resource fusion. Expert Systems with Applications, Vol. 40, Issue 18, pp.2889-2903.

[22] Colace, Francesco, De Santo, Massimo, Greco, Luca, Moscato, Vincenzo, and Picariello, Antonio. 2015. A collaborative user-centered framework for recommending items in Online Social Networks. Computers in Human Behavior, Computing for Human Learning, Behaviour and Collaboration in the Social and Mobile Networks Era, Vol. 51, Part B, pp. 694-704.

[23] Mittal, Vatsala, Kaul, Aastha, Sen Gupta, Santoshi, and Arora, Anuja. 2017. Multivariate Features Based Instagram Post Analysis to EnrichUser Experience. Information Technology and Quantitative Management (ITQM2017), Procedia Computer Science, Vol. 122, pp. 138-145.

[24] Yu, Wei, and Li, Shijun. 2018. Recommender systems based on multiple social networks correlation. Future Generation Computer Systems, Accepted Manuscript, 31 pages.

[25] Guo, Dongyan, Xu, Jingsong, Zhang, Jian, Xu, Min, Cui, Ying, and He, Xiangjian. 2017. User relationship strength modeling for friend recommendation on Instagram. Neurocomputing, ARTICLE IN PRESS, pp. 1-10.

[26] Duma, Mlungisi, and Twala, Bhekisipho. 2018. Optimising latent features using artificial immune system in collaborative filtering for recommender systems. Applied Soft Computing, Vol. 71, pp. 183-198.

[27] Ujjin, Supiya, and Bentley, Peter J. 2003. Particle swarm optimization recommender system. Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706), Indianapolis, IN, USA, USA.

[28] Thanh Hung, Bui. 2020. Integrating Sentiment Analysis in Recommender Systems. Reliability and Statistical Computing, pp. 127-137.

[29] Ziani, Amel, Azizi, Nabiha, Schwab, Didier, Aldwairi, Monther, Chekkai, Nassira, Zenakhra, Djamel, and Cheriguene, Soraya. 2019. Recommender System through Sentiment Analysis. HAL.

[30] Abbasi, Fatemeh, Khadivar, Ameneh, and Yazdinejad, Mohsen. 2019. A Grouping Hotel Recommender System Based on Deep Learning and Sentiment Analysis. Journal of Information Technology Management, Vol. 11, No. 2, pp. 59-78.

[31] Aida Osman, Nurul, Azman Mohd Noah, Shahrul, Darwich, Mohammad, and Mohd, Masnizah. Integrating contextual sentiment analysis in collaborative recommender systems. Plos One, Published: March 22, 2021 https://doi.org/10.1371/journal.pone.0248695.