A novel recommendation method based on general matrix factorization and artificial neural networks

Stelios Kapetanakis[1*], Nikolaos Polatidis[1], Gharbi Alshammari[1] and Miltos Petridis[2]

[1]School of Computing, Engineering and Mathematics University of Brighton, Moulsecoomb
Campus, Lewes Road, Brighton BN2 4GJ, UK
(s.kapetanakis, n.polatidis, g.alshammari) @brighton.ac.uk
[2]Department of Computer Science
Middlesex University London,
The Burroughs, London NW4 4BT, UK
m.petridis@mdx.ac.uk

Abstract. Collaborative filtering is a successful approach in relevant item or service recommendation provision to users in rich, online domains. This approach has been widely applied in commercial environments with success, especially in online marketing, similar product suggestion and selection and tailor-made consumer suggestions. However, regardless its market penetration, there are still considerable limitations in terms of the accuracy in proposed recommendations stemming from the high-frequency low-relevance user-item bias, data specificities and individual user patterns and needs that may be hidden in data. We propose a novel recommendation approach that improves accuracy and requires significantly less maintenance compared to traditional collaborative filtering. For the experimental evaluation, we use two real datasets and well-known metrics with the results validating our method. Our proposed method outperforms all the alternative recommendation methods for each of the two datasets and metrics and seems holistically effective against alternatives since it requires less settings to be considered without affecting the output.

Keywords: Recommender Systems, Collaborative Filtering, Matrix Factorization, Neural Networks

* Corresponding author: s.kapetanakis@brighton.ac.uk, Tel: +44 1273642563, Fax: +44 1273642405

1 Introduction

Recommender systems help users to easily find relevant items that meet their interests, thus addressing the information overload problem. Users can be confused while coming against large volumes of information and can have challenges with cognitive and data sparsity when attempting to find the most appropriate information to them to the shortest possible time span [4]. Recommender systems have an important role in the growth of online information navigation by filtering and recommending relevant items to users. The knowledge discovery approach can help making personalised recommendations by collecting user interests. Among others Collaborative filtering (CF) is one of the most successful techniques for recommender systems [17]. Several CF techniques have been proposed in different domains, such as e-commerce applications. Typically, elaborate approaches outperform the commonly-used k-nearest neighbour (kNN) recommendation method in terms of accuracy, particularly for sparse datasets or in terms of scalability as they rely on offline pre-processing or model-building phases [6].

Most CF approaches analyse user ratings to determine the similarity between users and items. The similarity measure is important for finding accurate results when recommender systems are utilized. However, it is challenging to determine distance measures in these systems in order to find similarities among users. CF is most frequently implemented via a kNN approach [11]. Similarity is calculated among users with common rating history whereas a neighbourhood of k users for each user is created recommendations are provided based on the neighbourhood. kNN relies significantly on the rating aspect, which allows users to assign a high or low rating to a certain item based on their preference or dislike for it [13]. Several similarity measures have been adopted in recommender systems such as Pearson's Correlation Coefficient (PCC) [17] and Cosine [20] to provide recommendations based on the absolute ratings between users. Thus, modified similarity measures is one of the most important research challenges in recommender systems, because the prediction accuracy in recommender systems can be improved substantially (or not) according to the applied method. In this paper we extend previous work [1] by introducing a recommendation method that is based on general matrix factorization and artificial neural networks. The main contributions of this paper are as follows:

1. We propose a new recommendation method that is based on general matrix factorization and artificial neural networks

2. We ran extensive experiments to show its effectiveness based on two real datasets, showing that our proposed method is both practical and effective

The rest of this paper is organized as follows: Section 2 presents the related work, section 3 explains the proposed method, section 4 contains the evaluation, while sections 5 and 6 contain the discussion and conclusions sections respectively.

2 Related Work

CF is the most popular technique for recommender systems. It has been widely implemented in different domains such as movies [15] and music [24] to generate recommendations. It is a method of information filtering that seeks to predict the rating that a user will give to a particular item based on a similarity matrix. CF provided a foundation for the first recommenders systems, that were used to help people make choices based on the opinions of other people with similar backgroun [7]. The task is to make an automatic prediction by considering other similar user ratings for an item. Therefore, the basic idea of CF is to find a user whose past rating behaviour is similar to that of the user the algorithm is currently trying to predict. This approach uses a kNN algorithm to calculate recommendations, and the main required data are the rating matrix and a function that computes similarity between users. An Item-based collaborative filtering (IBCF) technique looks into a set of items that the target users have rated and computes how similar they are to the target item; it then selects the k most similar items. At the same time, the corresponding similarity are also computed [18]. Once the most similar items are found, the prediction is computed using a weighted average of the target users' ratings. Hence, there are two main aspects to be considered: similarity computation and prediction generation [2]. Basically, to compute the similarity between items, the first step is to determine users who have rated both items and who have the most similar items with similar ratings. Many different measures can be used to compute the similarity between items, such as Pearson's correlation, cosine, Jaccard and Triangle similarity. Of these methods, Pearson's has had the most successful applications, which is defined in Equation 1 where Sim a,b is the similarity of users a and b, ra,p is the rating of user a for product p, r b,p is the rating of user b for product p and ra, rb represent user 's average ratings. P is the set of all products. However, in [21] the authors have proven that the triangle similarity outperforms Pearson correlation and shows improved results in IBCF.

$$\mathrm{Sim}_{a,b}^{PCC} = \frac{\sum_{P\in P}(r_{a,p} - \bar{r_a})(r_{b,p} - \bar{r_b})}{\sqrt{\sum_{p\in p}(r_{a,p} - \bar{r_a})^2}\sqrt{\sum_{p\in p}(r_{b,p} - \bar{r_b})^2}} \tag{1}$$

More recently, a combination of one or more methods called a hybrid recommender system has been applied to overcome the limitations of using one approach and obtain better results [5]. For instance, in [3], a hybrid case-based reasoning approach was proposed to solve a long tail problem, which is items that have a few ratings by switching between collaborative filtering and content-based filtering. In addition, the authors in [12] implemented a hybrid recommender system that applied clustering technique and an artificial algae algorithm with a multi-level CF approach. However, co-rated items have been used for a problem solving in recommender systems to improve their predictive accuracy. Authors in [23] also introduced a hybrid approach for solving the problem of finding the rating of unrated items in a user-item matrix through a weighted combination of user-based and item-based collaborative filtering. These methods addressed the two major challenges of recommender systems, the accuracy of recommendations and sparsity of data, by simultaneously incorporating the correlation of users and items. In [22] the authors address a cold-start problem in user-based CF by considering both the distance between users and the co-rating of items using Jaccard factors. In [21], the authors proposed a new measure that integrates the triangle similarity approach with Jaccard similarity, which consider non-co-rating users. The authors in [16] propose a multi-level constraint that improves the quality of a recommendation using PCC. Equation 2 considers the similarity between users relying on PCC and co-rated items in different levels. In the equation, sim denotes the similarity between two users. T stands for the total number of co-rated items. tl, t2, t3 and t4 are the predefined threshold of co-rated items for user similarity Sim and y is a threshold for the similarity value returned by the PCC function. In the equation t1, t2, t3 and t4 are set as natural numbers when PCC is applied. In case the number of co-rated items falls within the specified range, while the similarity value y is of a certain value (between -1 and 1) or more, a float positive value x1, x2, x3, or x4 is added to the similarity value. Thus, the PCC similarity values between users are manipulated and user nearest neighbourhoods are rearranged according to the numerical values set to t1, t2, t3, t4, x1, x2, x3, x4 and y.

$$Sim_{i_j i_q}^{PCC} = \left\{ \begin{array}{c} Sim_{i_j i_q}^{PCC} + x1, if \ \left|\frac{Ia \cap Ib}{T}\right| \geq t1 \ and \ Sim_{i_j i_q}^{PCC} \geq y \\ Sim_{i_j i_q}^{PCC} + x2, if \ \left|\frac{Ia \cap Ib}{T}\right| < t1 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t2 \ and \ Sim_{i_j i_q}^{PCC} y \\ Sim_{i_j i_q}^{PCC} + x3, if \ \left|\frac{Ia \cap Ib}{T}\right| < t2 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t3 \ and \ Sim_{i_j i_q}^{PCC} \geq y \\ Sim_{i_j i_q}^{PCC} + x4, if \ \left|\frac{Ia \cap Ib}{T}\right| < t3 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t4 \ and \ Sim_{i_j i_q}^{PCC} \geq y \\ 0, if \left|\frac{Ia \cap Ib}{T}\right| < t4 \ \ otherwise \end{array} \right\} \qquad (2)$$

3 The Proposed Method

Previous work [1] has proposed a multi-level triangle recommendation method, which is explained in section 3.1 below, that improved the recommendation accuracy and effectiveness outer performing all baselines using MAE. To automate further this approach, in this paper we propose the use of a method based on general matrix factorization (GMF) and artificial neural networks (ANN) which is explained in section 3.2. Our motivation for this enhancement is the nature of factorisation or decomposition which reduces the feature space by applying more effective linear matrix representation. At its most simple form matrix factorisation is reducing the original feature matrix into two (for example lower and upper triangular matrices respectively, requiring less operations to solve. Further to the performance improvements of GMF, its combination with ANN proposes substantial improvements offers similar or better results in certain cases compared to the method in 3.1 dynamically, whereas the triangle multi-level recommendation is a static method that needs to be observed and change settings manually when required.

3.1 Triangle multi-level recommendation

The number of co-rated items reflects the degree of connection between users. For instance, a high number of co-rated items might indicate a high level of similarity. Traditional similarity metrics do not consider the number of co-rated items[19]. To solve this problem, a triangle similarity has been proposed by [21], which results in a significance improvement in accuracy when it is combined with co-rating. The triangle similarity is integrated with some constraints that apply a number of co-rated items. In our proposed method, we apply a hybrid method that also adopts a multi-level CF approach, which enhances the similarity value of users that belong to certain categories and ignores the rest [16]. It enhances the process of kNN by finding a large margin within an application. The triangle similarity measure is defined in equation 3, where Sim a,b is the similarity of users a and b, ra,p is the rating of user a for product p, r b,p is the rating of user b for product p and ra, rb represent user 's average ratings. P is the set of all products.

$$Sim_{a,b}^{Tri} = 1 - \frac{\sqrt{\sum_{u \in c_{a,b}} (r_a - r_b)^2}}{\sqrt{\sum_{u \in c_{a,b}} {r_a}^2} + \sqrt{\sum_{u \in c_{a,b}} {r_b}^2}} \tag{3}$$

The value range is [0,1], where closer a value is to 1, the more similar they are. The triangle approach considers both the length of the vectors and the angle between them, so it is more reasonable than the angle-based cosine similarity. For example, if the two vectors A = 5,5,5 and B = 1,1,1 are given, then cosine similarity is 1. By contrast, the triangle similarity between them is 0.33. Furthermore, the triangle similarity is shown in figure 1 and the proposed multi-level triangle-based method is defined in equation 4.

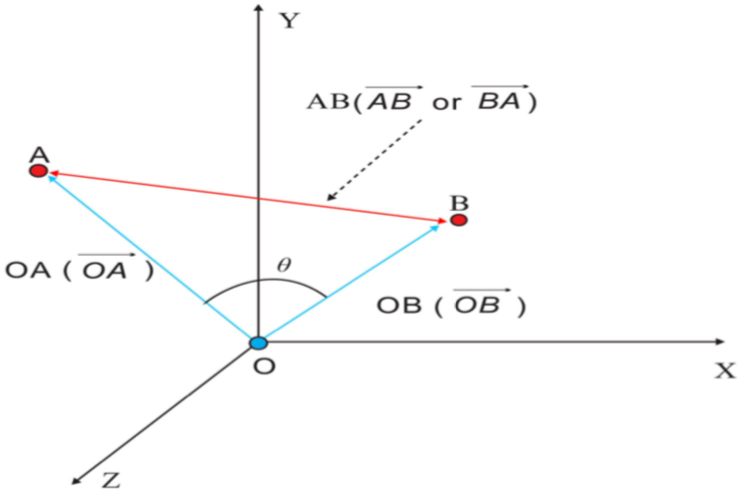
Y
AB($\overrightarrow{AB}$ or $\overrightarrow{BA}$)
A
B
OA ($\overrightarrow{OA}$)
θ
OB ($\overrightarrow{OB}$)
O
X
Z

Fig. 1. A triangle in three-dimensional (3D) space [21].

$$Sim_{\mathrm{a,b}}^{Proposed} = \left\{ \begin{array}{c} Sim_{a,b}^{Tri} + x1, if \ \left|\frac{Ia \cap Ib}{T}\right| \geq t1 \ and \ Sim_{a,b}^{Tri} \geq y \\ Sim_{a,b}^{Tri} + x2, if \ \left|\frac{Ia \cap Ib}{T}\right| < t1 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t2 \ and \ Sim_{a,b}^{Tri} \geq y \\ Sim_{a,b}^{Tri} + x3, if \ \left|\frac{Ia \cap Ib}{T}\right| < t2 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t3 \ and Sim_{a,b}^{Tri} \geq y \\ Sim_{a,b}^{Tri} + x4, if \ \left|\frac{Ia \cap Ib}{T}\right| < t3 \ and \left|\frac{Ia \cap Ib}{T}\right| \geq t4 \ and \ Sim_{a,b}^{Tri} \geq y \\ 0 \qquad , if \ \left|\frac{Ia \cap Ib}{T}\right| < t4 \ otherwise \\ \end{array} \right\} \tag{4}$$

In the above equation, sim a,b denotes the similarity between user a and user b. T stands for the total number of co-rated items. tl, t2, t3 and t4 are the predefined threshold of co-rated items for user similarity Sim and y is a threshold for the similarity value returned by the PCC function. The method compares the similarity given by Sim using y and the number of co-rated items. If it is less than a specified level t, then it goes to the next level and continues to the next level until the right level is found. If all four levels are not found, then the similarity is equal to 0.

3.2 Recommendations based on general matrix factorization and artificial neural networks

CF recommendations rely on k-Nearest Neighbours and similarity measuring involving algorithms like Euclidean distance, variations of Euclidean distance and others like Pearson correlation or Triangle distance (as presented above). Traditional kNN seems effective in capturing linear relationships across data. However, several cases may contain non-linear elements, and, in such cases, an Artificial Neural Network may be able to capture them. To investigate this hypothesis, we propose an alternative similarity finding method by applying Collaborative Filtering with a used Multi-Layer Perceptron (MLP) as the similarity function. Approaches to improve CF treat users (Us) and ratings (Rs) in a form of matrix factorisation where an inner product of Us x Rs is estimating a degree of similarity. Our proposed method replaces this calculation by applying an ANN instead of the inner product to calculate similarity. This can be performed using a range of ANN approaches from MLP, Convolutional, Recurrent networks and others. For this work, we propose an MLP approach to estimate its efficiency against traditional CF approaches. For the needs of the training one-hot embeddings are used for each user (u) and movie (m) respectively. Pairs of user-movie embeddings are combined using Generalised Matrix Factorisation (GMF) using a dot product. Then an MLP is applied to generated embeddings for users and movies respectively. Advanced Neural networks in the form of Convolutional or long short term memory (LSTM) networks seem effective in areas like natural language processing or image processing and can be used in recommender systems as well. We chose a plain MLP approach to investigate its efficiently in combination with GM and one hot-encodings, whereas applications of LSTMs and other advanced forms is planned for the near future. A feature vector is being created and reused in further MLP layers. Finally, a neural Matrix Factorisation combines both predictions from GMF and MLP to obtain the final prediction. Matrix Factorisation (MF) assumes that a user vs movies preference matrix $M \in R^{nxm}$ is equivalent into a product of lower rank matrices $K \in R^{d\,x\,n}$, $L \in R^{d\,x\,m}$ and can identify similarities by ranking through K and L. MF can be successful since it relies on identifying more granular relationships from small dataset shards and by applying less calculations. In equation 5, the matrix equation is defined, whereas MLP is defined in equation 6 and 7. For the equation 5 the notation reads: M is the user preference matrix whereas K and L are the lower rank matrices as presented above. Since in most applied cases besides the user preferences there is feature bias [25] (such as time and space) equation 5 presents this bias as feature x.

$$\widehat{M}\ (i,j,x) = K^T(i,x)L\ (j,x) \tag{5}$$

The hyperbolic tangent is selected as the transfer function of the MLP classifier, where φ is the output of the ith node (neuron) and υ is the weighted sum of the input connections. Tanh is selected as activation function due to its simplicity and efficiency in calculating derivatives from functions.

$$\varphi_i\ (v_i) = \tanh\ (v_i) \tag{6}$$

$$X = \begin{pmatrix} X_0 \\ . \\ . \\ . \\ X_n \end{pmatrix} W = \begin{pmatrix} \omega_0 \\ . \\ . \\ . \\ \omega_n \end{pmatrix} v = \sum_{j=1_{b=\,\omega x_0\,,x_0=1}}^{n} x_{i\omega_i+b=\,\omega^T\,x} \tag{7}$$

A classifier structure with vector inputs represented as $x_1, x_2, ..., x_n$, weights as $\omega_1,\omega_2, ..., \omega_n$; b is the bias and φ is the transfer function; υ refers to the weighted sum of the input.

4 Experimental Evaluation

The experimental evaluation took place on a Windows 10 based computer using the java programming language for the method in section 3.1 and the python programming language for the method in section 3.2. Furthermore, 2 real datasets have been used along with comparisons to a baseline and a state-of-the-art alternative. The evaluation metrics include the well-known rating prediction error measures Mean Absolute Error (MAE) and Precision at 10.

4.1 Real datasets

We have run our experiment with two real datasets in order to compare the results with different parameters, such as the number of items and users. Both datasets have been evaluated using cross-validation with 5 folds and K is the number of neighbours.

MovieLens 100.000: This is a real dataset that is publicly available. It contains 943 users and 1,682 movies. Each user has rated at least 20 movies. It contains 100,000 ratings, all of which are in a range between 1 and 5. The three main features are [UserID], [MovieID] and [Rating] [8].

MovieLens 1.000.000: This dataset contains 1,000,209 anonymous ratings of 3,900 movies made by 6040 users. All ratings are in a scale between 1 and 5. This dataset is also publicly available for running offline experiments and is widely used for collaborative filtering recommender systems [8].

4.2 Comparison

The following recommendation methods have been used for the comparisons.

PCC: In this method, the statistical correlation between the similar ratings of two users is calculated to find users that are the closest to a particular user. The output will be a value between -1 and 1; 1 is a totally positive correlation, 0 indicates that there is no correlation and -1 is a totally negative correlation.

Multi-level CF: This algorithm relies on PCC similarity measures. It is based on multiple levels, from the top to the bottom, with each of these levels having a number of constraints that are defined in the equation 2, where sim (a,b) denotes the similarity between user a and user b. T stands for the total number of co-rated items; t1,t2,t3 and t4 are the threshold of co-rated items for user similarity. It is considered that t1=50, t2=20, t3=10 and t4=5. In addition, x is defined as x1=0.5, x2=0.375, x3=0.25, x4= 0.125 and y=0.33.

The above two methods are compared in our proposed method. Our method considers both the length and angle of the rating vectors between users. Multi-level approach also considers the right level for each user after calculating the triangle similarity and compare it with a specified threshold. In addition, in each level, a certain constraint is conducted to modify the similarity between certain users who share similar items.

4.3 Evaluation metrics

Recommender system researchers have applied different measures to evaluate the quality of proposed recommendation algorithms [10]. Since 1994 [17], most of the empirical studies examining recommender systems have focused on appraising the accuracy of these systems using different methods [9]. Appraisals of accuracy are useful for evaluating the quality of a system and its ability to forecast the rating for a particular item. Predictive accuracy measurement metrics are widely used by the research community in CF, which measures the similarity between true user ratings and recommender system predicted ratings. Hence, we apply both the mean absolute error (MAE) and Precision to measure the performance of the proposed methods and evaluate their prediction accuracy compared with other recommendation techniques. MAE is defined in Equation 8 and Precision is defined in Equation 9 as the ration of True positives divide by the sum of the identified true positives plus false positives. In equation 8 pi is the predicted rating, and ri is the actual rating. Furthermore, lower values mean a lower error and thus better rating predictions. In Equation 9, higher values are better.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - r_i| \tag{8}$$

$$\text{Precision} = \frac{True\ positives}{True\ positives + False\ positives} \tag{9}$$

4.4 Experimental results

The results in section 4.4.1present a comparison between the baseline, the state-of-the-art alternative the proposed triangle multi-level method defined in section 3.1, while the results in section 4.4.2 present the results obtained with the use of the proposed method defined in section 3.2.

4.4.1 Triangle multi-level recommendation method evaluation results

Figure 2 presents the MAE results respectively for the MovieLens 100.000 dataset and figure 3 presents the MAE results for the MovieLens 1.000.000 dataset. In all figures K is the number of nearest neighbours, Pearson is the PCC baseline method explained in section 4.2 and multi-level CF is the state-of-the-art alternative explained in section 4.2 as well.

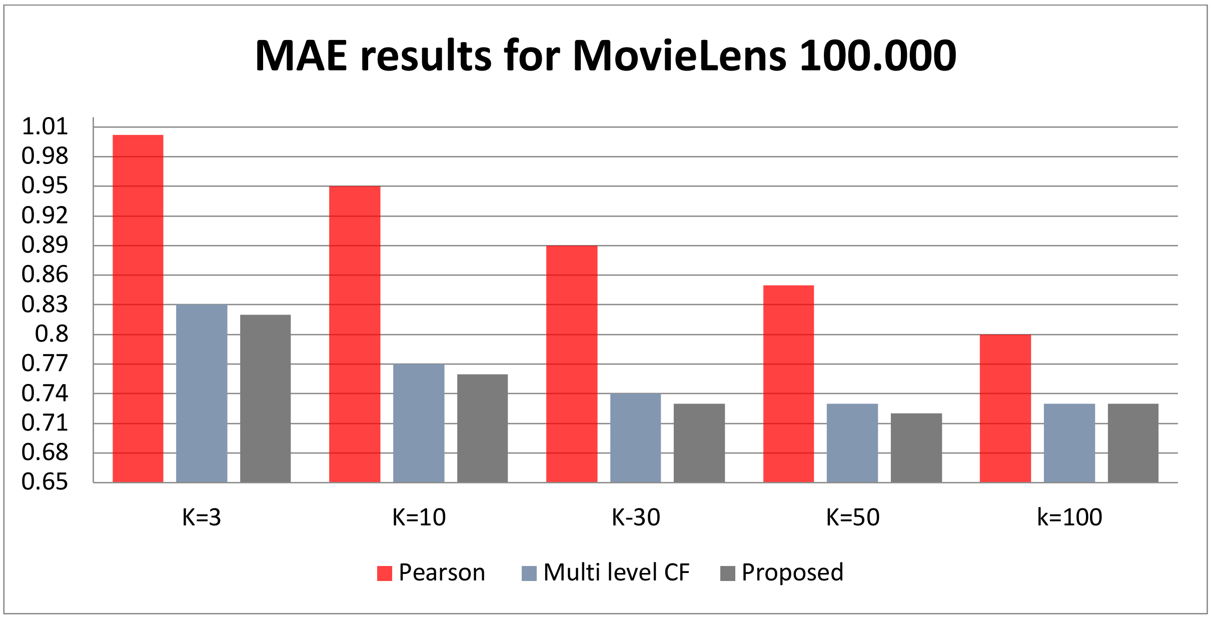

Figure 2. MAE results based on the MovieLens 100.000 dataset

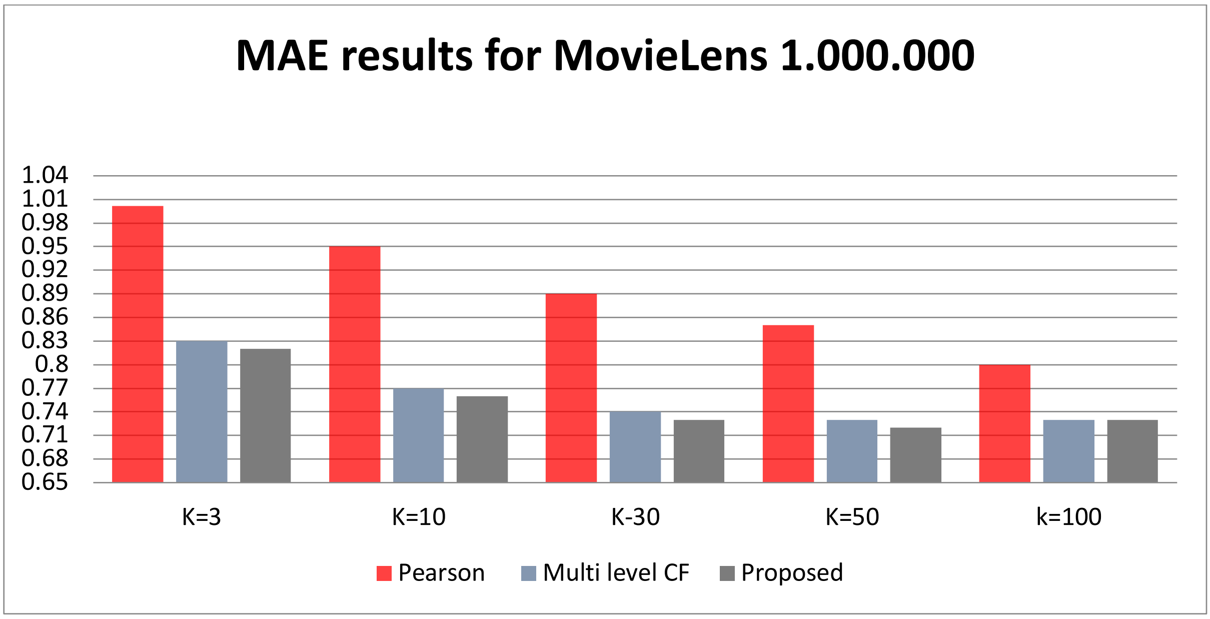

Figure 3. MAE results based on the MovieLens 1.000.000 dataset

4.4.2 General matrix factorization and artificial neural networks recommendation method evaluation results

For the evaluation, we used dataset splitting it in an 80-10-10 training, validation and testing mode based on a 5-fold validation method per MLP. Table 1 below presents the MAE results based on the MovieLens 100.000 and 1.000.000 datasets. Furthermore, tables 2 presents the Precision comparison for both the MovieLens 100.000 and 1.000.000 datasets.

Dataset	PCC	Multi-level CF	Proposed	Proposed ANN
MovieLens 100.000	0.898	0.760	0.752	**0.738**
MovieLens 1.000.000	0.906	0.732	0.716	**0.714**

Table 1. MAE results

Dataset	Precision @ 10 for PCC	Precision @ 10 for multi-level CF	Precision @ 10 for the proposed ANN
MovieLens 100.000	0.193	0.189	0.174
MovieLens 1.000.000	0.159	0.1574	0.141

Table 2. Precision results

The evaluation results show great efficiency for the proposed method, however compared with previous work [1], Multi-level CF and PCC still show to be superior by a small percentage. Experimentation with GMF and ANNs shows a competent and fast approach to converge in recommender datasets, however more advanced neural network similarity functions should be experimented to extend the experimental coverage of possible efficient advanced recommender system techniques.

5 Discussion

Recommender systems are decision support systems that rely mostly in CF approaches to provide recommendations of items or services to users that might be of interest to them, thus trying to solve the information overload problem to an extent. Initially we extended the classical CF approach by proposing a multi-level triangle-based CF method found in section 3.1. The initial proposed method performs well enough and outperforms the alternative recommendation methods used for the comparisons in most cases. As shown in figures 2 and 3 the proposed method outperforms the alternative when the user neighbourhood is up to 50 users for the MovieLens 100.000 dataset. The results in figures 4 and 5 for the MovieLens 1.000.000 dataset are similar to the first dataset with the proposed method performing better in most cases. In all cases and methods as the user neighbourhood grows the performance is improved, with the more advanced methods performing better as well. However, more advanced means more complicated with more settings being available, thus making it harder to decide which method to use in online environments where users constantly submit ratings and the data change. In offline evaluations researchers have the ability to run several tests until the optimal settings are identified and such methods might have to be checked again after certain periods of time for performance once deployed.

ANNs however can be applied to solve this problem and perform better compared to the alternative CF based methods. The proposed ANN recommendation method performs better when compared to the average MAE for both datasets and all alternative recommendations methods as shown in tables 2 and 3. In online commercial environments an ANN method such as the one proposed in section 3.2, will be more valuable in terms of performance and maintenance. The proposed ANN method outperforms all the alternative methods in all metrics and datasets, thus providing better recommendation to users with the minimal intervention required. CF based methods need to be maintained in order to validate that their output is the best possible according to the settings and the data.

6 Conclusions

In this paper we proposed a recommendation method based on general matrix factorization and ANNs, which improves the overall quality of the recommendations provided to users and minimizing human intervention and offline evaluations. Initially we proposed a triangle multi-level CF recommendation method, followed by the ANN method and showing that although similar results can be obtained for both methods, the ANN method is more flexible in terms of settings and maintenance. Furthermore, an extensive experimental evaluation took place based on 2 real datasets and well-known rating prediction error metrics with the results validating the proposed method. In the future we aim to investigate the applicability of our method in different recommendation domains and evaluating using different evaluation metrics according to the domain. Based on the presented results in the future work we will extend the current approach using sequence to sequence similarity metrics that can lead to richer recommender models and item coverage. This approach is novel in the area of recommender systems and will be extended to other domains such as online marketing and e-commerce recommendations to test further and evaluate it. So far, the results have been substantially encouraging and will be extended and applied further.

Conflict of Interest: The authors declare that they have no conflict of interest.

References

1. Alshammari G, Kapetanakis S, Polatidis N, Petridis M. A Triangle Multi-Level Item-Based Collaborative Filtering Method that Improves Recommendations. In International Conference on Engineering Applications of Neural Networks 2018 Sep 3 (pp. 145-157). Springer, Cham.
2. Aggarwal, C.C., et al.: Recommender systems. Springer (2016)

3. Alshammari, G., Jorro-Aragoneses, J.L., Kapetanakis, S., Petridis, M., Recio- Garcfa, J.A., Diaz-Agudo, B.: A hybrid cbr approach for the long tail problem in recommender systems. In: International Conference on Case-Based Reasoning. pp. 35-45. Springer (2017)
4. Bobadilla, J., Ortega, F., Hernando, A., Gutierrez, A.: Knowledge-Based Systems Recommender systems survey 46, 109-132 (2013)
5. Burke, R.: Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction 12(4), 331-370 (2002)
6. Gedikli, F., Jannach, D.: Recommending based on rating frequencies: Accurate enough? In: Proceedings of the 8th Workshop on Intelligent Techniques for Web Personalization & Recommender Systems at UMAP lO (ITWPlO). pp. 65-70 (2010)
7. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Communications of the ACM 35(12), 61-70 (1992)
8. Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5(4), 19 (2016)
9. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. pp. 230-237. ACM (1999)
10. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) 22(1), 5-53 (2004)
11. Jeong, B., Lee, J., Cho, H.: Improving memory-based collaborative filtering via similarity updating and prediction modulation. Information Sciences 180(5), 602- 612 (2010)
12. Katarya, R., Verma, O.P.: Effectual recommendations using artificial algae algorithm and fuzzy c-mean. Swarm and Evolutionary Computation 36, 52-61 (2017)
13. Konstan, J.A., Riedl, J.: Recommender systems: from algorithms to user experience. User modeling and user-adapted interaction 22(1-2), 101-123 (2012)
14. Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., Khudanpur, S.: Recurrent neural network based language model. In: Eleventh Annual Conference of the International Speech Communication Association (2010)
15. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J.: Movielens unplugged: experiences with an occasionally connected recommender system. In: Proceedings of the 8th international conference on Intelligent user interfaces. pp. 263-266. ACM (2003)
16. Polatidis, N., Georgiadis, C.K.: A multi-level collaborative filtering method that improves recommendations. Expert Systems with Applications 48, 100-110 (2016)
17. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM conference on Computer supported cooperative work. pp. 175-186. ACM (1994)
18. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web. pp. 285-295. ACM (2001)
19. Shen, K., Liu, Y., Zhang, Z.: Modified similarity algorithm for collaborative filtering. In: International Conference on Knowledge Management in Organizations. pp. 378-385. Springer (2017)
20. Shi, Y., Larson, M., Hanjalic, A.: Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. ACM Computing Surveys (CSUR) 47(1),3 (2014)
21. Sun, S.B., Zhang, Z.H., Dong, X.L., Zhang, H.R., Li, T.J., Zhang, L., Min, F.: Integrating triangle and Jaccard similarities for recommendation. PloS one 12(8), e0183570 (2017)
22. Tan, Z., He, L.: An efficient similarity measure for user-based collaborative filtering recommender systems inspired by the physical resonance principle. IEEE Access 5, 27211-27228 (2017)
23. Wei, S., Zheng, X., Chen, D., Chen, C.: Electronic Commerce Research and Applications A hybrid approach for movie recommendation via tags and ratings q 18, 83-94 (2016)
24. Yoshii, K., Goto, M., Komatani, K., Ogata, T., Okuno, H.G.: An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Transactions on Audio, Speech, and Language Processing 16(2), 435-447 (2008)
25. Weston, J., Wang, C., Weiss, R., and Berenzweig, A. Latent collaborative retrieval. In International Conference on Machine Learning (ICML), Edinburgh, Scotland, June 2012.