A Multi-view Clustering Approach for the Recommendation of Items in Social Networking Context

Lamia Berkani[1,2](✉), Lylia Betit[1,2], and Louiza Belarif[2]

[1] Laboratory for Research in Artificial Intelligence (LRIA), Algiers, Algeria
[2] Department of Computer Science, USTHB University, Bab Ezzouar, 16111 Algiers, Algeria
{lberkani,lbetit,lbelarif}@usthb.dz

Abstract. The review of the literature shows that the clustering-based approaches suffer from relatively low accuracy and coverage. We propose in this article a multi-views clustering approach for the recommendation of items in social networks. First, users are iteratively clustered from the views of both rating patterns and social information which includes different features, namely: friendship, trust and influence. Different clustering algorithms have been used (Kmedoids and CLARANS algorithms). Then, based on the multi-view clustering, recommendations are generated according to different hybridization. In order to evaluate our approach, experiments have been conducted using the well-known FilmTrust dataset. The results we have obtained show that our approach outperforms the existing related work approaches and baselines.

Keywords: Multi-view clustering · Social recommendation · Collaborative filtering · User's profile · Friendship · Trust · Influence

1 Introduction

Clustering-based approaches are being demonstrated to be efficient and scalable to large-scale data sets. As a dimension-reduction method, they are capable of alleviating the sparsity of rating data [1]. However, although clustering techniques have been demonstrated to be efficient and scalable to large-scale data sets, the clustering-based approaches gain less attention in recommender systems. Even if some works reported that by applying more advanced clustering method, the accuracy can be further improved and even outperform the other collaborative filtering (CF) approaches [2, 3]. To address this problem, Guo et al. [4] developed a multi-view clustering method through which users are iteratively clustered from the views of both rating patterns (user similarity) and social trust relationships (social similarity).

We focus in this paper by this same issue for the social recommendation of items. The proposed items' recommendation algorithm considered a multi-views clustering approach (MV), where users are clustered from the views of both user similarity (using users' assessments on items) and social relationships. In this paper we have formalized

M. R. Senouci et al. (Eds.): CSA 2020, LNNS 199, pp. 241–251, 2021.
https://doi.org/10.1007/978-3-030-69418-0_22

the social information that includes the list of friends, trust and influence. Different clustering algorithms have been used, including: Kmedoids and CLARANS algorithms. Furthermore, in order to determine the influencing medoids of the community, the Degree of Centrality heuristic have been used.

The remainder of this paper is organized as follows: Sect. 2 presents some related work on clustering-based recommender systems. Section 3 proposes a novel social recommender system using a multi-views clustering approach. Then in Sect. 4, experiments are conducted and analyzed. Finally, Sect. 5 highlights the most important results and draws some future perspectives.

2 Literature Review

Few works have tried to integrate social relationships into clustering-based methods with the aim of better performance of CF. Sun et al. [5] proposed a social regularization approach that incorporates social network information, namely the users' friendships and rating records (tags) for the prediction of the recommendations. They used a bi-clustering algorithm to identify the most suitable group of friends for generating different final recommendations. DuBois et al. [6] combined a correlation clustering algorithm and trust models together to derive trust from the connection distance in a trust network. However, only limited improvement is observed. On the other hand, Selvi and Sivasankar [7] considered a model-based CF based on a fuzzy c-means clustering approach.

According to Guo et al. [4] previous clustering-based approaches suffer from relatively low accuracy and, especially, coverage. To alleviate these issues, they developed a multi-view clustering method through which users are iteratively clustered on the basis of rating patterns in one view and social trust relationships in the other [4]. Sheugh and Alizadeh [8] proposed a multi-view clustering based on Euclidean distance, merging similarity and trust relationships including explicit and implicit trusts. He et al. [9] proposed a web items recommendation system based on a multi-content clustering CF model. According to the authors, a multi-view clustering can be applied to the mining of the similarity and relevance of items to improve the classic CF. Different views such as user ratings and user comments have been considered and users' preferences were analyzed by their historical interaction features and additional behavior features for an appropriate recommendation.

3 MV-Based Recommendation Approach

We present in this section our recommendation approach which proposes a novel multi-views clustering technique (MV). Figure 1 illustrates the general structure of our recommendation system, including the multi-views clustering-based recommendation as well as other hybrid algorithms. First, we start with a representation of the user's profile and then we expose the different algorithms of our recommendation system: (1) the CF (i.e. standard and clustering-based CF algorithms); (2) the Social filtering (SocF) (i.e. the standard and clustering-based SocF algorithms); and (3) the hybrid algorithm (HybF), including the weighted HybF (i.e. standard and clustering-based HybF) and the multi-view clustering-based HybF algorithm.

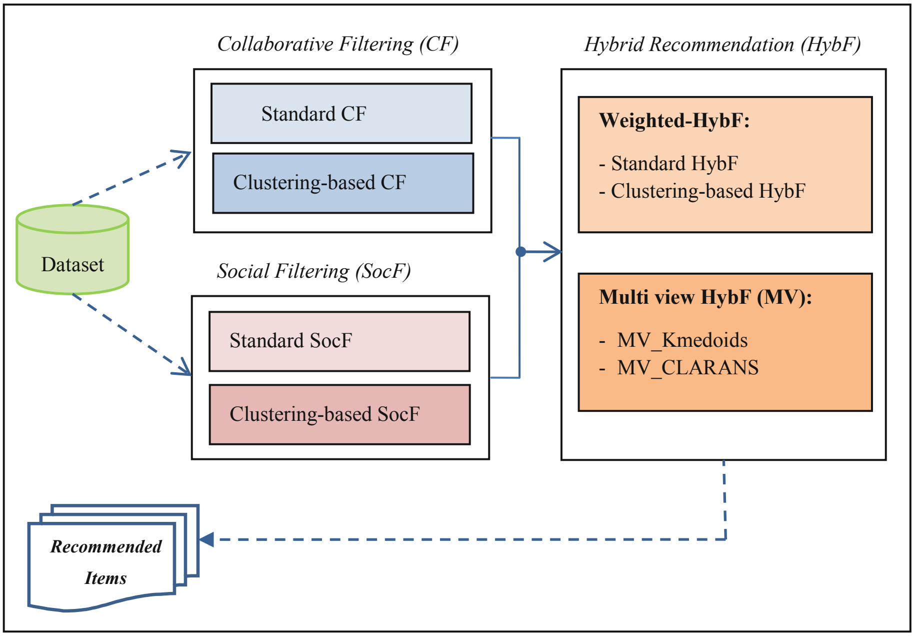

Fig. 1. General structure of our recommendation system

3.1 User's Profile Modeling

We consider that each user is characterized by a set of information including the following: assessments made on social network resources (reviews), the list of friends, trust and influence. Our recommendation algorithm is based on a hybridization of the two algorithms: collaborative filtering (CF) and social filtering (SocF). The similarity between two users is based on collaborative and social distance calculation.

Collaborative Distance. The similarity calculation between two users, u and v, is based on the evaluation history, and uses the Pearson correlation function [10]. The distance between two users, u and v, denoted $D_{Sim}(u, v)$ is calculated as follows:

$$D_{Sim}(u, v) = 1 - Sim_{Pearson}(u, v) \tag{1}$$

Social Distance. In order to determine the social relationship between users, we use three features, namely: trust, influence, and friendship:

1. Extraction of the degree of Trust (u, v): There are several algorithms for the calculation of trust. We have chosen the six-level method presented by Guo [4], which calculates how much two users, u and v trust each other, considering a distance equal to six. The distance $D_{Trust}(u, v)$ is calculated as follows:

$$D_{Trust}(u, v) = 1 - Trust(u, v) \tag{2}$$

2. Extraction of the degree of influence $Inf(u, v)$: The degree of influence between two users u and v is calculated taking into account the number of items that are liked/not liked by u and v according to the items that are liked/not liked by u, as presented by this formula:

$$Inf(u, v) = \frac{1}{2} * \left(\frac{Nb\,Items\,liked\,Comm(u, v)}{Nb\,items\,liked\,(u)} + \frac{Nb\,Items\,not\,liked\,Comm\,(u, v)}{Nb\,items\,not\,liked\,(u)} \right) \tag{3}$$

The distance $D_{Inf}(u, v)$ is calculated as follows:

$$D_{Inf}(u, v) = 1 - Inf(u, v) \tag{4}$$

3. Extraction of the degree of friendship: The extraction of the degree of friendship between two users is calculated with the Jaccard formula:

$$Friendship(u, v) = \frac{|F_u \cap F_v|}{|F_u \cup F_v|} \tag{5}$$

where: F_u is the set of friends of u and F_v is the set of friends of v.
The distance $D_{Friendship}(u, v)$ is calculated as follows:

$$D_{Friendship}(u, v) = 1 - Friendship(u, v) \tag{6}$$

3.2 Collaborative Filtering

We considered two variants of the CF: (1) the standard CF; and (2) The CF based on clustering technique. These algorithms will be presented as follows:

Standard CF. The standard CF allows detecting the community of the active user u based on the distances that separate him from the other users (D_{Sim}). The prediction will be based on the neighbors who are most similar to the active user. We chose the user-based filtering. This type of filtering starts by fully loading the usage matrix into memory (including user ratings on items), and then two important steps are executed:

- Selection of the K closest neighbors according to a given similarity threshold (if D_{Sim} $> =$ threshold), where the value of the threshold is fixed according to the density of the database and varies in the interval $[-1, 1]$.
- Prediction based on the ratings of the K closest neighbors.

Clustering-Based CF. This algorithm detects the community of the active user u using one of the clustering techniques. The prediction will be based on users' evaluations that belong to the same cluster as the active user. The CF algorithm based on the Kmedoids classification, for example, has been implemented according to the following two variants:

– The first variant of the algorithm consists in applying Kmedoids clustering.

– The second variant of the algorithm consists in applying the Kmedoids clustering, however when choosing the initial medoids we used a “Degree of centrality” heuristic to determine the influencing medoids of the community.

The CF algorithm based on Kmedoids is as follows:

```
Algorithm 1: CF-based Kmedoids

Input: Similarity Distance D_Sim, Active user U_a, Item j,
Number of medoids K, Heuristic: Boolean.

Output: Prediction for the active user on the item j.

BEGIN

  // Select Init_Medoids set according to the Heuristic

  1: If Heuristic = false then Init_Medoids = Random se-
     lection of K initial medoids.

     Else Init_Medoids = Selection of initial K medoids ac-
     cording to the "Degree of centrality" heuristic.

     End if.

  2: Apply the Kmedoids algorithm using Init_Medoids set
     and D_Sim for the computation of the distances;

  3: Identify the cluster of U_a.

  4: Predict (U_a, j).

END.
```

3.3 Social Filtering

As for the CF, the SocF algorithm consists of two variants: (1) the standard SocF; and (2) the Clustering-based SocF. The SocF considers the three features: trust, influence and friendship to calculate the social distance. A weighted formula was used to calculate the social distance, as follows:

$$D_{Soc} = \beta_1 D_{Trust} + \beta_2 D_{Inf} + \beta_3 D_{Friendship} \tag{7}$$

where: $\beta_1, \beta_2, \beta_3$: represent the importance weights related, respectively, to trust, influence and friendship, with: $\beta_1 + \beta_2 + \beta_3 = 1$.

Standard SocF. The pseudo algorithm below illustrates the standard SocF steps:

```
Algorithm 2: SocF ()

Input: Active user U_a, the distances: D_Trust, D_Inf, D_Friendship,
the item j, the number of k nearest neighbors to consider
for prediction.

Output: Prediction (U_a, j).

BEGIN

  1: Construct D_Soc table using the weighted D_Soc formula
     (Equation 7);

  2: Generate the clustering configuration based on D_Soc ;

  3: Select the K closest neighbors in terms of D_Soc;

  4: Apply the prediction (U_a, j) based on the assessments
     of the K nearest neighbors;

END.
```

Clustering-Based SocF. This algorithm is similar to the clustering-based CF. The only difference consists to replace the collaborative distance D_{Sim} by the social distance D_{Soc}.

3.4 CF and SocF Hybridization

The Hybrid filtering algorithm (HybF) combines the two filtering algorithms (CF and SocF) to consider both views (similarity view and social view) at the same time. We implemented the hybrid algorithm using two different methods, namely: (1) the weighted hybrid method (W-HybF); and (2) the multi-view clustering hybrid method (MV-HybF).

The Weighted HybF (W-HybF). This algorithm combines the CF and SocF algorithms according to the following formula:

$$D_{Hyb} = \alpha * D_{Sim} + (1 - \alpha) * D_{Soc} \tag{8}$$

where: α is the importance degree of the CF; and intuitively, (1 − α) is the importance degree of the SocF.

The detail of this formula is represented as follows:

$$D_{Hyb} = \alpha * D_{Sim} + (1 - \alpha) * \left(\beta_1 D_{Trust} + \beta_2 D_{Inf} + \beta_3 D_{Friendship}\right) \tag{9}$$

Therefore, we will have different hybrid algorithms: (1) the standard weighted hybrid algorithm; and (2) the weighted clustering-based hybrid algorithm.

1. *The standard weighted hybrid algorithm:* This algorithm starts by calculating the distances between the active user U_a and all the other users by applying the previous formula (Formula 9), in order to identify the k nearest neighbors to U_a. Then the prediction function will be applied.

2. *The weighted clustering-based hybrid algorithm:* The weighted hybrid algorithm with clustering proceeds according to the following steps:

- Calculate the distances between U_a and the other users using D_{Hyb};
- Generate a final configuration of clustering;
- Identify the cluster associated with the active user;
- Calculate the predictions using the harmonic average weighted prediction formula considering the users of the cluster where U_a belongs.

The MV-HybF. We adapted the multi-view algorithm presented by Guo et al. [4], with improvements and extensions. Our proposal consists to combine the social and similarity information. This hybridization method is, however, different from that mentioned in the weighted hybridization algorithm (W-HybF) because with this method we will not have to compute once the distance by considering a weighted combination between the two pieces of information. However we will calculate the distance D_{Sim} to make a classification according to the similarity and separately we calculate the distance D_{Soc} and we will make a classification according to a weighting of the social information. These two classification processes run in an alternative way and end with an integration phase to generate a final multi-views classification. The clustering-based MV is executed according to the following steps:

- *Step 1:* This step consists in enriching the social information of the MV algorithm considered in Guo et. al [4] (where only trust information has been taken into account), adding influence and friendship.
- *Step 2:* This step allows the selection of medoids randomly or using the "Degree of centrality" heuristic. The algorithms generated by this method are named as follows: Adapted_MV_Kmedoids and Adapted_MV_KMedoids_H; (with or without heuristic). In this step, the classification will be done once according to the social view, and another time according to the similarity view while passing the resulting medoids from each classification step to the next classification that will follow. Thus two clustering configurations of different views C_{Soc} and C_{Sim} will be built.
- *Step 3:* Apply an integration algorithm.
- *Step 4*: With this hybridization technique, a user can belong to at most two clusters, due to the integration step (step 3). Once the final configuration is obtained, we will only have to identify the community of the user U_a to be able to calculate its prediction on the item j to recommend. The latter can belong either to a single cluster *"C"* and thus the generation of prediction $P^{C}_{U_a,j}$ will be done with the formula of the harmonic mean, or it will belong to the intersection of two clusters, in this case the prediction generation will proceed in two ways:

1. *The Harmonic average method:* let us start with the most trivial method that will be noted *AVG*, which consists in calculating the average of the two predictions: $P^{C_1}_{U_a,j}$ and $P^{C_2}_{U_a,j}$ applying the formula of the harmonic mean and considering the two clusters C_1 and C_2 separately where U_a appears, as follows:

$$AVG = \frac{1}{2}\left(P^{C_1}_{U_a,j} + P^{C_2}_{U_a,j}\right) \tag{10}$$

2. *The SVR regression-based method:* in order to improve our results, we applied a second method for the generation of the prediction of the active user, which belongs to the intersection of two clusters. This method involves using supervised classification for the prediction, using the SVR regression technique.

4 Experiments

In order to evaluate our approach, experiments have been conducted using the well-known FilmTrust[1] dataset.

4.1 Dataset Description

FilmTrust is a database collected from the website that allows users to share movie ratings. It explicitly specifies other users as trusted neighbors. We used the dataset provided by Guo et al. [4] where the ratings are assigned in a range of 0.5 to 4 with a step of 0.5. This database contains 2853 trusted links linking 1508 users, making 35,497 reviews on 2071 listings. The density of this database is equal to 1.14.

4.2 Evaluation Metrics

We used the following two metrics: the MAE metric (Mean Absolute Error) and the RMSE metric (Root Mean Square Error):

$$MAE = \frac{\sum_{u,i\in\omega}\left|r_{u,i} - p_{u,i}\right|}{|\omega|} \tag{11}$$

$$RMSE = \sqrt{\frac{\sum_{u,i\in\omega}\left(r_{u,i} - p_{u,i}\right)^2}{|\omega|}} \tag{12}$$

where:

- ω: is the set of test evaluations and | ω | is the size of the test assessments;
- $r_{u,i}$: represents the evaluations of the user *u* on the item *i;*
- $p_{u,i}$: represents the prediction of the evaluations of the user *u* on the item *i*.

4.3 Results

Evaluation of the CF. We have evaluated the CF with a variation of the similarity threshold (CF-Threshold) from 0.1 to 1. The best value obtained is for the CF-threshold equal to 0.6. This value will be considered in the rest of our evaluations.

Contribution of Social Information. To better explore the contribution of social information to the recommendation and to set our parameters, we evaluated all the social information, namely: trust, influence and friendship with different weights and varying the number of clusters *K*. The results are presented in Table 1, where we have highlighted in bold the best obtained values with the number of clusters $K = 95$. These values will be considered in the rest of our evaluations.

1 www.FilmTrust.com.

Table 1. Evaluation of the social information

Weights	K	MAE	RMSE
$\beta_1 = 0.33$, $\beta_2 = 0.33$, $\beta_3 = 0.33$	95	0.685208	0.8812
$\boldsymbol{\beta_1 = 0.25}$, $\boldsymbol{\beta_2 = 0.5}$, $\boldsymbol{\beta_3 = 0.25}$	**95**	**0.68460**	**0.8825**
$\beta_1 = 0.5$, $\beta_2 = 0.25$, $\beta_3 = 0.25$	95	0.684779	0.8810
$\beta_1 = 0.25$, $\beta_2 = 0.25$, $\beta_3 = 0.5$	95	0.684782	0.8811

Evaluation of the Standard HybF. We tested different weights that allowed us to determine the best values between the CF and the SocF. Table 2 shows these results:

Table 2. Evaluation of the standard HybF

Weights	K	MAE	RMSE
$\alpha = 0.5$	100	0.64174	0.8648
$\alpha = 0.3$	100	0.64173	0.8647
$\boldsymbol{\alpha} = 0.7$	**100**	**0.63491**	**0.8575**

Contribution of Clustering on the Recommendation. We have evaluated the clustering-based CF and the clustering-based SocF. Table 3 summarizes the results we have obtained using the clustering Kmedoids algorithm with the different recommendation algorithms (CF, SocF and HybF):

Table 3. Comparison between CF, SocF and HybF algorithms

Algorithm	Without heuristic	With heuristic
CF using D_{Sim}	MAE: 0.7004 RMSE: 0.8946	MAE: 0.6913 RMSE:0.8712
SocF using D_{Soc}	MAE: 0.7378 RMSE:0.9371	MAE: 0.701 RMSE:0.88
HybF ($\alpha * D_{Sim} + (1 - \alpha) * D_{Soc}$)	**MAE: 0.6887** **RMSE: 0.8767**	**MAE: 0.691** **RMSE: 0.8752**

We can see that hybridization has significantly improved the CF algorithm in terms of MAE and RMSE (for both cases, using the heuristic or without using it).

Clustering-based recommendation algorithms (see Table 4):

Table 4. Evaluation of the MV-based recommendation algorithm

Multi-Views approaches	K	MAE	RMSE
MV_Kmedoids_H	50	0.6287	0.7091
MV_CLARANS_H	30	**0.5675**	0.7413
MV_CLARANS_Trust	10	0.6115	0.8051
MV_Kmedoids-Trust [4]	10	0.67271	0.8604

– The adapted MV with Kmedoids using the heuristic "Degree of centrality" and considering all the social information, i.e. trust, influence and friendship. This algorithm is noted by: MV_Kmedoids_H.
– The MV with Kmedoids and considering only Trust information, which is equivalent to the approach proposed by Guo et al. [4]. This algorithm will allow us to perform a comparison of our algorithms with the existing related work algorithms.
– The adapted MV with CLARANS clustering algorithm using the heuristic "Degree of centrality" and considering all the social information. Furthermore we have also considered the MV with CLARANS where the social information based on the trust information.

We can see the contribution of the MV algorithm compared to the existing related work. The use of friendship and influence in addition to trust to model social information has contributed to this improvement. Furthermore, the use of the degree of centrality heuristic has also enhanced the results, especially with the CLARANS algorithm which has given the best results compared to the Kmedoids algorithm.

5 Conclusion

This paper proposes a recommendation approach which applies a multi-views clustering method for the recommendation of items. Different clustering algorithms have been used. The multi-views approach (combining the similarity view and the social view) proved to be the best in terms of MAE and RMSE, compared to the hybrid and the clustering-based hybrid algorithms. The results we have obtained are promising especially when using CLARANS algorithm and the degree of centrality heuristic. In our future work, we plan to test our approach with other databases and compare it with other existing approaches [11]. Moreover, we envisage to further improve the recommendation accuracy: (1) using optimization techniques to have the best partitioning for a given clustering such as in [12]; (2) applying deep learning techniques [13]; and (3) integrating other features like users' interaction behavior and items popularity such as in [14].

References

1. Pham, M., Cao, Y., Klamma, R., Jarke, M.: A clustering approach for collaborative filtering recommendation using social network analysis. J.UCS J. **17**, 583–604 (2011)
2. Bellogín, A., Parapar, J.: Using graph partitioning techniques for neighbour selection in user-based collaborative filtering. In: Proceedings of the 6th ACM Conference on Recommender Systems (RecSys), pp. 213–216 (2012)
3. Salah, A., Rogovschi, N., Nadif, M.: A dynamic collaborative filtering system via a weighted clustering approach. Neuro Comput. **175**, 206–215 (2016)
4. Guo, G., Zhang, J., Yorke-Smith, N.: Leveraging multi-views of trust and similarity to enhance clustering-based recommender systems. KBS J. **74**, 14–27 (2015)
5. Sun, Z., Han, L., Huang, W.G., Wang, X., Zeng, X., Wang, M., Yan, H.: Recommender systems based on social networks. J. Syst. Softw. **99**, 109–119 (2015)
6. DuBois, T., Golbeck, J., Kleint, J., Srinivasan, A.: Improving recommendation accuracy by clustering social networks with trust. Recommender Syst. Social Web **532**, 1–8 (2009)
7. Selvi, C., Sivasankar, E.: A novel optimization algorithm for recommender system using modified fuzzy c-means clustering approach. Soft Comput. **23**, 1–16 (2017)
8. Sheugh, L., Alizadeh, S.H.: Merging similarity and trust based social networks to enhance the accuracy of trust-aware recommender systems. J. Comput. Rob. **8**(2), 43–51 (2015)
9. He, X., Kan, M.Y., Xie, P., Chen, X.: Comment-based multi-view clustering of web 2.0 items. In: International World Wide Web Conference, Seoul, Korea, pp. 771–781. ACM (2014)
10. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. TKDE **17**, 734–749 (2005)
11. Li, W., Ye, Z., Xin, M., Jin, Q.: Social recommendation based on trust and influence in SNS environments. Multimedia Tools Appl. **76**(9), 11585–11602 (2017)
12. Berkani, L.: Social-based collaborative recommendation: bees swarm optimization based clustering approach. In: The 9th International Conference on Model & Data Engineering (2019)
13. Deng, S., Huang, L., Xu, G., Wu, X., Wu, Z.: On deep learning for trust-aware recommendations in social networks. IEEE Trans. Neural Netw. Learn. Syst. **28**(5), 1164–1177 (2016)
14. Lai, C.H., Lee, S.J., Huang, H.L.: A social recommendation method based on the integration of social relationship and product popularity. Int. J. Hum Comput Stud. **121**, 42–57 (2019)