Chaos, Solitons and Fractals 103 (2017) 194–204

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Community clustering based on trust modeling weighted by user interests in online social networks

Farman Ullah[a,b,1], Sungchang Lee[a,*]

[a] School of Electronics and Information Engineering, Korea Aerospace University, Republic of Korea
[b] Department of Electrical Engineering, COMSATS Institute of Information Technology, Attock, Pakistan

ARTICLE INFO

Article history:
Received 10 July 2016
Revised 22 May 2017
Accepted 29 May 2017

Keywords:
Community clustering
Social trust
User interests
Probabilistic modeling

ABSTRACT

Online social networking websites provide platforms through which users can express opinions and preferences on a multitude of items and topics, and follow users and information, and flood it by retweeting. User-user interests vary, and based on the users' interests, they can be grouped to multiple implicit interest communities. However, every interaction and user may not be trustworthy. Capturing the user's interaction with others, and predicting user interest and trust from the interactions are important parts of social media analytics. In this paper, we propose community clustering for implicit community detection based on trust and interest modeling. The trust modeling is weighted by the user's interests to group the users in multiple clusters having higher interest and trust similarity within a cluster. The proposed community clustering algorithm begins by ranking the nodes by the weighted degree and then selecting the initial community centers that are not in the neighbors of each other's. We then assign the user to the community with whom the user has the higher interest and trust similarity and higher common connections topology. We provide a probabilistic trust model to predict the unknown reliable trust between users considering their friends. We model user interests based on preferences and opinions, as well as the content experienced in social media. Furthermore, we evaluate the proposed algorithm comparing publicly available datasets with well-known algorithms for clustering quality.

1. Introduction

The rapid development of web technologies such as web2.0/3.0, online social networks (OSNs), multi-device platforms, and communications technology has enabled people to express their preferences on products and share opinions on various topics en masse. Social media platforms are transforming the ways people live and interact, and as well as how they do business and market products. OSNs provide platforms for information sharing but also cause an information overload problem, uncertainty and risk of reliability of information from various users. Recommender systems (RSs) overcome the information overload problem and provide personalized services and content to enhance users' continuity and belonging on social websites. RSs are the applications and web-based tools that employ user preferences, opinions, and experienced items and products to predict the users' interests and suggest personalized items for them going forward [1–4]. Users' interests are broad and vary, but are also sometimes heterogeneous interactions depending on various factors such as the topic of interest, time and spatial location that lead to the sparsity issue for finding user-user interest similarity. Communities' detection algorithms group users with similar interests and similar nature of interactions to improve the sparsity issues and efficiency of recommendation algorithms [5–6].

OSNs are complex networks and can be modeled by a graph $\boldsymbol{G}(\boldsymbol{U}, \boldsymbol{V})$, where $\boldsymbol{U}$ is the list of users or items, and $\boldsymbol{V}$ represents the set of edges showing the relationship between the users or items. For example, the relationship might be the friendship on Facebook. Communities are the subgraphs within an OSN (in a social graph) having higher nodes and edge density within the subgraph and low among the subgraphs [7]. Community detection in network analysis is one of the most fundamental and important tasks in many fields and research areas such as sub-markets identification [8] for products and brand awareness, sexual exploitation of children [9], and bacterial communities in water [10]. The terms 'community' and 'cluster' [11] appear interchangeably, but the main difference is that clustering partitions the graph. Indeed, every node belongs to exactly one cluster wherein the communities' – but especially overlapping communities' – nodes may belong to more than one cluster. Communities extract the useful in-

* Corresponding author.
E-mail addresses: sclee@kau.ac.kr, sclee712@gmail.com (S. Lee).

[1] Farman Ullah was Graduate student at Korea Aerospace University during which he carried out this research and submit this paper. Now he joined COMSATS Pakistan as a Faculty member.

http://dx.doi.org/10.1016/j.chaos.2017.05.041
0960-0779/

formation from the network and explore the network structure. Communities' detection focuses mostly on edge density to group the users, but there is nevertheless a risk of uncertainty and reliability concerning the users in the group [12]. In this paper, we propose community detection to group those users with high trust and similar interests between them in one community in OSNs.

Trust is the subjective probability of belief about a person or the objective degree of users' previous knowledge and experience [13]. People regularly require the opinion of family, friends and acquaintances on whom one trusts more to decide even very small things like where to eat, which movie to watch, and so on [14]. Social connectedness of users in OSNs has changed interactions, sharing and collaborating but also increased the risk of uncertainty and reliability to accept the information increases. Trust is getting attention in OSNs because it improves the cooperation and interactions among SN members, as well as reducing uncertainties and risk from unreliable users and therefore mitigating the information overload problem. Abdul-Rahman et al. [13], defined three types of trusts: (i). *Interpersonal trust*: where an agent has direct trust of another agent (agent and context specific) (ii). *System/Impersonal Trust*: this type of trust is not based on any property or state of the trustee but rather on perceived properties or reliance on the system wherein that trust exists (iii). *Dispositional/Basic Trust*: This type of trust refers to the general attitude of trusting. The user can provide trust value by providing a true explicit value but most of the time it is difficult and the user does not provide the value [15]. In this paper, we suggest the probabilistic trust model to predict the unknown trust values between two unfamiliar users considering the trustworthy paths that connects these users.

The rest of the paper is organized as follows: Section 2 briefly introduces the background and related works. The proposed social trust and interests based dynamic communities' detection for personalized recommendation appears in Section 3. Section 4 exhibits the simulation results and compares them with other schemes and algorithms. Finally, Section 5 concludes the paper.

2. Background and related works

The scope of this paper is closely related to the users' interest identification and personalization in the heterogeneous social media, as well as trust finding and propagation in the OSNs between users, and detections of communities.

2.1. Predicting user interest, and personalization of products/content for a user

The surge of the OSNs has enabled users to generate content and share it at any time and from any location. Rapoport [58] first revealed the importance of degree distribution for information dissemination and propagation in the OSNs. Understanding and capturing the user's interest from heterogeneous interactions in the OSNs are critical tasks [16] due to tackling and analyzing structured and unstructured content. OSNs provide the medium for sharing en masse but also increase the information overload problem. RSs are the application tools and web-based services that use user opinions, preferences, interactions and products they have experienced to personalize the user services such as movies, games, and advertisements [1–4]. The core of the RSs is the recommendation algorithm, classified into three main types: collaborative filtering recommendations, content based filtering, and hybrid recommendation approaches.

Collaborative filtering (CF) RSs acquire user ratings on products or product features, and learn the user interests from ratings so as to personalize the product [1–4,17–18]. CF uses the ratings of untold numbers of users to find similarity user-user or item-item to personalize the items for a given user [19]. Based on ratings information usage the CF algorithms are divided into main categories as follows: (i) Memory-based CF [20] uses the entire ratings to predict users' interest and personalize the recommendation such as K-nearest neighbors (KNN) approach [21]. Memory-based CF algorithms are easy to implement but as the size of the rating data increases, so does the likelihood of sparsity and scalability issues. (ii) Model-based CF [24] uses the rating information to create a training model for generating recommendations such as matrix factorization [22–23], a Bayesian classifier [25], neural networks [26–27], and clustering [3–4]. The model-based CF improves the sparsity issues and computational efficiency by using reduced features instead of the whole dataset.

The content-based filtering (CBF) RSs use products/items features and attributes such as movies genres, product description and keywords to make recommendations to the user [4,28–29]. Ferman et al. [30] provided a structured description of multimedia content and user profile to filter personalized user content. Deldjoo et al. [31] proposed techniques to analyze the video contents and automatically extract the video stylistic features such lighting, color, and motion for a CBF-based recommendation system. CBF-based RSs do not consider the user's rating information, and they assume that items with similar features and attributes are rated similarly by users. CBF algorithms have the limitation of requirements of the structural information and useful structured data, such as movie-genre-based recommendations [32]; however, this is ineffective for unstructured data. CBF restricts the users to products that have similar features and attributes previously experienced by users, though they cannot recommend different features.

Hybrid RSs use CF or CBF (or a combination) with any other context information such as demographics or location [4,33–34]. Hybrid RSs can be [35]: W*eighted*, which combines recommendations together to produce a single recommendation; S*witching,* which depends on the situation switch between various recommendation schemes; *Mixed,* which presents recommendations at the same time from several recommendation techniques; *Feature Combination*, which employs a single recommendation method combining features from various recommender systems; *Feature augmentation* is the output of one recommendation used as an input feature to another; *Cascade* involves one recommendation that refines recommendations given by another; and finally, *Multi-Level* is a model learned by one and input to the other. Hybrid RSs improve the sparsity and cold start issues of the CFs methods.

2.2. Trust in online social networks

Trust plays a major role in twenty-first century online social life by facilitating coordination and cooperation for mutual benefits and is the product of past experiences and perceived trustworthiness [36]. People constantly modify and upgrade their trust in others based on their feelings in response to changing circumstances. OSNs provide the platform to generate user-centric content and share it with a large number of individuals, but those users are uncertain about the reliability of information those sharing it. Reliable information can be provided by acquiring the trust value in another user/product explicitly or implicitly from the user [37]. The online products reviewing and sharing community (http://epinion.com/) allows users to express their opinions regarding trust and maintains a web-trusted network in which edge shows "1" (trust), "−1" (distrust) and "0″ (neutral). However most of the times, the user does not provide trust explicitly in another user; many trust inference models have been developed to infer trust in OSNs [38]. Kim and Song [39] provided the reinforcement learning-based trust inference model to find a reliable trust path from the source node to the unknown target node. Bedi et al. [14] proposed a trust-based RS that stores knowledge in the form of ontologies and generates recommendations based on trust between users.

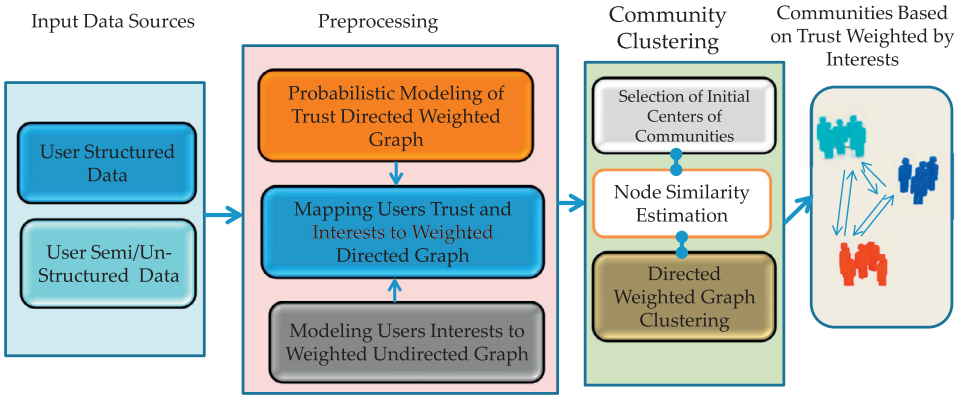


Fig. 1. Architecture of community clustering based on social trust modeling weighted by the users' interests in online social networks.

2.3. Communities detection in online social networks

Understanding network structure and dynamics, and the detection of communities has been a fundamental challenge in networks to provide user personalized services. Newman [40,41] proposed Modularity function that adopted a fast community detection algorithm to increase the detection speed and quality of the community. Community detection based on divisive hierarchical clustering finds the community structure, (i) calculate the betweenness for all the edges (ii) remove the highest betweenness edge (iii) recalculate the betweenness of all edges affected by the removal (iv) steps (ii) and (iii) are repeated until no edges remain [42]. Kim and Kim [43] proposed detecting optimal overlapping and hierarchical communities in the complex networks using interaction-based edge clustering. They considered complex network topology but also interactions-based edge weights to identify overlapping and hierarchical communities. Node popularities-based overlapping communities' detection is proposed in [44–45]. The authors considered node popularity to decide whether to include the node in the community and adopted the hierarchical Bayesian scheme to find communities. This helped in adaptively shrinking or removing irrelevant communities. Clique Percolation Method [46] was the first method based on overlapping community detection in the complex networks. Eustace et al. [47] proposed using local neighborhood-based community detection in the networks. Liang et al. [56] proposed an iterative searching algorithm for community detection in a graph. A community description model is provided that considers simultaneously a node "local importance" in a community and a node "important concentration" in all communities. If a node has more neighbors in a community then it has more local importance and importance concentration is if a node has high "local importance" in a community rather than other communities, the node has high responsibility to it. A node is assigned to a community based on a similarity measure and at each iteration the description model is updated. A density-based community clustering algorithm using local expansion method is proposed [57]. The algorithm is based on the structural centrality, which incorporates local density of nodes and relative distance between clusters. Ginestra et al. [59] proposed triadic closure based basic generating mechanism of communities in complex networks. High community clustering coefficients imply high number of triads (triangles) in the network, and more triads are formed between nodes of the same cluster than the nodes of different clusters. In Ref. [60], the authors studied collaboration network of science and movies actors as a multiplex network where the node have different relationship. The relationships are represented by the graphs or layers. A model is proposed to grow multiplex networks based on mechanisms of intra and inter triadic closure which mimic the real collaboration process.

3. The proposed community clustering based on social trust modeling weighted by the user interests in OSNs

In this section, we present the community clustering using social trust relationship and user's interests. The objective is to find communities having higher internal edge density with social trust and interest within the community, as well as minimize the influence of external edges between the communities. Fig. 1 shows the architecture and scheme of detection of communities using social trust and interests of users in OSNs. First, we map user interests and social trust into a directed weighted graph and then use the modeled social graph to cluster the nodes.

3.1. Probabilistic modeling of the social trust between users in OSNs

Trust is the subjective degree of belief that a user has in another user, while reputation is the accumulation of objective views of belief in a certain user's expertise from the whole online community members [48]. In OSNs, the user can provide the trust value explicitly but most of the time the user at first fails to provide trust and is also unfamiliar with most of the users when they initially join the SNS. A Facebook study [49] showed that the user can reach another user on an average of 4.7 hops (edges). Considering this fact and also that users have more trust in direct friends and friends-of-friends, we propose the probabilities trust modeling that considers two hops between the source node to the target node as a means of predicting unknown trust value between them. Fig. 2 shows the pictorial example to overview the proposed probabilistic trust modeling between the users. To explain the concept, we assumed simple networks graphs where the edge weight represents the trust of one user on another. In Fig. 2(a) the edge weights are based on the trust analysis of a community of social network site, Advogato [61] where a user can provide a trust value {0.6, 0.8, 1} on other user in the network. Fig. 2(b) is based on (http://epinion.com/) allows users to express their opinions regarding trust and maintains a web-trusted network in which edge shows "1″ (trust), "−1″ (distrust) and "0″ (neutral). However most of the times, the user does not provide trust explicitly in another user.

We predict the trust from the source to the unknown target considering the two hops to reach the target node. The trust from source A to the target C in Fig. 2(a) is predicted by Eq. (1).

$$\tau(A,C) = 1 - \frac{1}{|S|}\sum_{j\in S}\left(1 - p_{AB_j}p_{B_jC}\right)^{\alpha};\ \alpha > 0 \qquad (1)$$

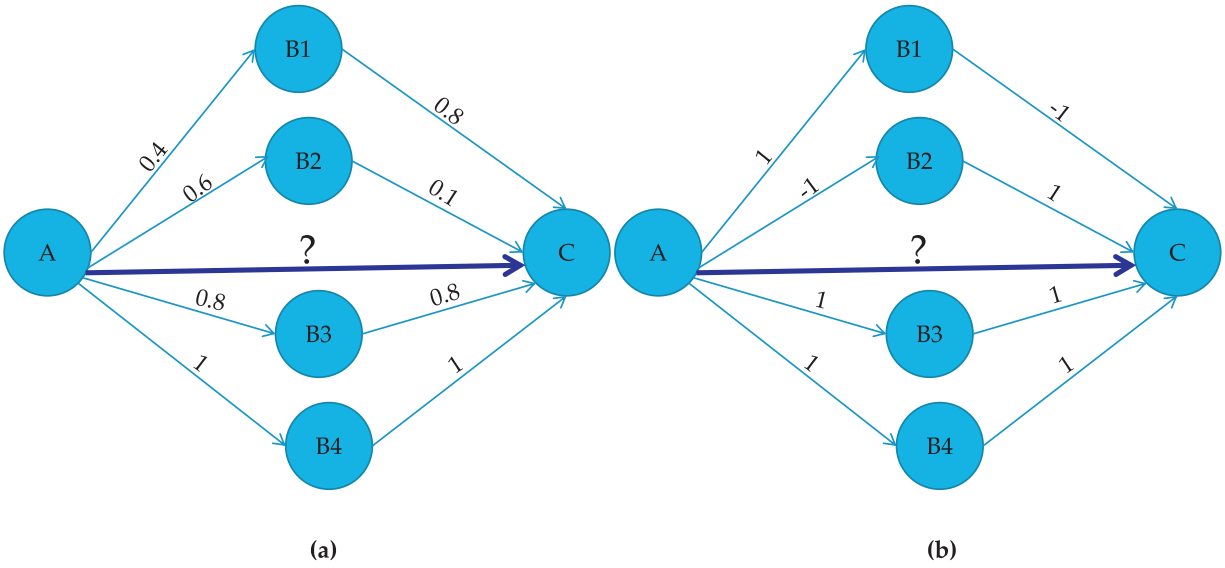


Fig. 2. Probabilistic trust modeling overview from source node to the target unknown node (a) Probability trust values between users based on the concept of trust analysis in social network website, Advogato (b) Categorical trust values between users based on (http://epinion.com/).

where $\boldsymbol{S}$ is the set of nodes that join the node $\boldsymbol{A}$ to node $\boldsymbol{C}$, and $\boldsymbol{p}$ shows the trust belief one user has in other. $\boldsymbol{\alpha}$ is the tunable parameter to predict the trust value, and for simplicity in this paper we estimated using ($\boldsymbol{\alpha} \cong \frac{\log(0.5)}{\log(\langle\tau\rangle)}$) and $\langle\tau\rangle$ is the average trust value in the network. Considering that Fig. 2(a) is part of the graph in which $\langle\tau\rangle = 0.8$, so $\boldsymbol{\alpha}\cong 3.1$ and $\tau(\boldsymbol{A}, \boldsymbol{C})\cong 0.71$.

To predict the trust value in the web trusted network where users provide trust in others user such as "1" trustworthy, "−1" untrustworthy and "0" neutral, we used Eq. (2) considering Fig. 2(b).

$$\tau(\boldsymbol{A}, \boldsymbol{C}) = \begin{cases} 1; & \dfrac{\boldsymbol{P_{tAC}}}{\boldsymbol{P_{tAC}} + \boldsymbol{N_{tAC}}} > .55 \\ -1; & \dfrac{\boldsymbol{P_{tAC}}}{\boldsymbol{P_{tAC}} + \boldsymbol{N_{tAC}}} < .45 \\ 0; & .45 \leq \dfrac{\boldsymbol{P_{tAC}}}{\boldsymbol{P_{tAC}} + \boldsymbol{N_{tAC}}} \leq .55 \end{cases} \tag{2}$$

where;

$$\boldsymbol{P_{AC}} = \sum_{(\boldsymbol{j} \in \boldsymbol{P}) \cap (\boldsymbol{p_{AB}} \cap \boldsymbol{p_{BC}} = 1)} 1$$
$$\boldsymbol{N_{AC}} = 1 - \boldsymbol{P_{AC}}$$

$\boldsymbol{P}$ is the number of paths from $\boldsymbol{A}$ to $\boldsymbol{C}$, $\boldsymbol{P_{tAC}}$ are the positive trust paths and $\boldsymbol{N_{tAC}}$ are the negative trust paths.

3.2. Modeling user-user interests similarity graph based on users' interests

The development of web2.0/3.0 and communication technologies gather user interests explicitly in terms of rating values or stars, or implicitly from the user interactions with online websites. In this paper, we consider that users provide their interests in terms of rating on the experienced item/product value from 1–5. We model the user-user similarity weighted graph from the users-items graph using the cosine similarity Eq. (3).

$$\boldsymbol{sim}(\boldsymbol{u}, \boldsymbol{v}) = \frac{\boldsymbol{r_u}.\boldsymbol{r_v}}{||\boldsymbol{r_u}||\ ||\boldsymbol{r_v}||} \tag{3}$$

where $\boldsymbol{r_u}$ and $\boldsymbol{r_v}$ are the co-rated items by the users $\boldsymbol{u}$ and $\boldsymbol{v}$ respectively. Fig. 3 shows the pictorial overview of the user-user similarity weighted graph modeling from the users-items bipartite graph. Table 1 shows the matrix representation of the graph.

3.3. Modeling the social graph based on user-user interest similarity and social trust modeled graphs

We model the social graph considering the user-user interests similarity graph and social trust graph by assigning different weights to the users' interests' similarity and trust. The edge weight in the modeled social graph can be found using Eq. (4).

$$\boldsymbol{w}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{\beta}\ \boldsymbol{sim}(\boldsymbol{u}, \boldsymbol{v}) + \left(1 - \boldsymbol{\beta}\right)\ \boldsymbol{\tau}(\boldsymbol{u}, \boldsymbol{v}) \tag{4}$$

where $\boldsymbol{\beta} \in [\boldsymbol{0}, \boldsymbol{1}]$ is a tunable parameter to adjust the weights of user-user interest similarity and the trust between users. The modeled social graph is a weighted directed network because the trust is directed although the user-user similarity graph is weighted and undirected. Fig. 4 shows the overview of the proposed social graph using the user-user interest similarity and social trust with $\boldsymbol{\beta} = 0.6$.

3.4. Community clustering using the modeled edge weights of user-user interest similarity and social trust

The general concept of the community regards a subgraph of a graph that has highly connected topological nodes that maximize the internal edge density within a community and maximize the external edge density with other communities. We cluster user interests and trust-based modeled weighted directed graph into communities with higher trust and interest weighted edge density. We propose a dynamic community detection algorithm consisting of three steps to cluster the modeled weighted directed graph. First, we select the set of nodes as the initial community centers with the highest weighted and topological connections with their neighbors and each center node should be at least 2-hops from the other center nodes. Second, we assign the node to a community with which the node has higher weighted similarity and topological connections. Third, we use the hierarchical clustering to merge the communities if needed and improve the community structure using the quality function.

3.4.1. Selection of initial centers' of communities in OSNs

Identification of Initial centers of communities/clusters is a key challenge in finding the communities with minimum processing time and maximizes the internal edge density, simultaneously

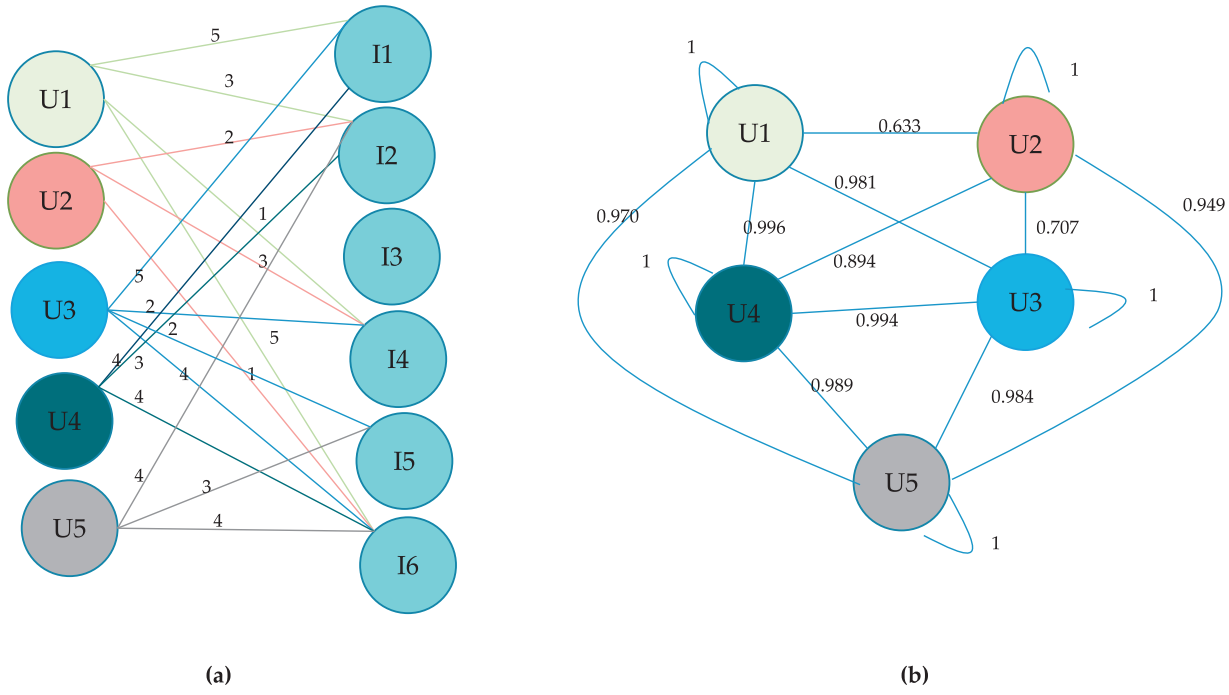


Fig. 3. Modeling the user-user interest similarity graph based on users' interests (a) users-items bipartite graph (b) user-user similarity graph.

Table 1.
Matrix representation of the graph (a) matrix of users-items bipartite graph (b) users-user interests similarity based adjacency matrix.

Users	**Items**						Users	Users				
	I_1	I_2	I_3	I_4	I_5	I_6		U_1	U_2	U_3	U_4	U_5
U_1	5	3	–	1	–	5	U_1	1	0.633	0.981	0.996	0.970
U_2	–	2	–	3	–	1	U_2	0.633	1	0.707	0.894	0.949
U_3	5	–	–	2	2	4	U_3	0.981	0.707	1	0.994	0.984
U_4	4	3	–	–	–	4	U_4	1	0.894	0.994	1	0.989
U_5	–	4	–	–	3	4	U_5	0.970	0.949	0.984	0.989	1

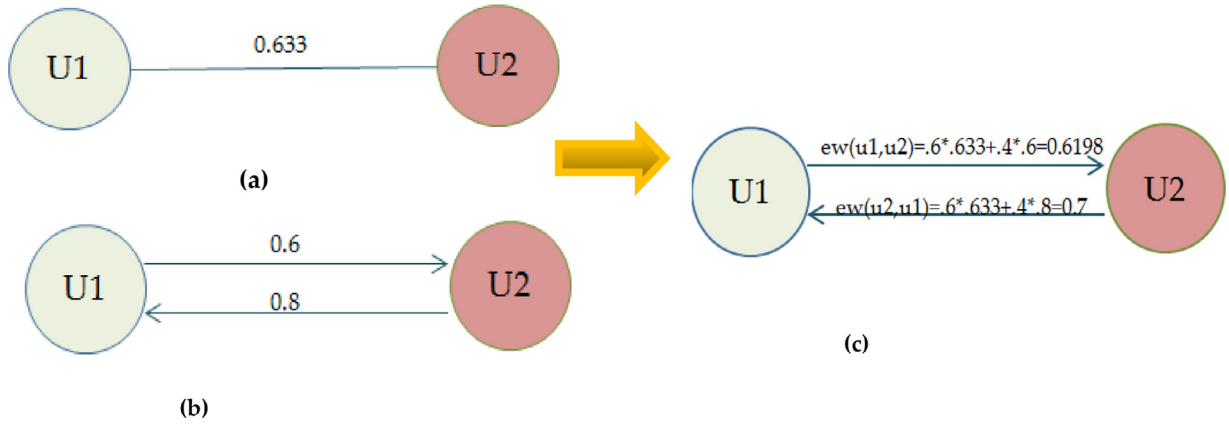


Fig. 4. Modeling the social graph using user-user interests similarity and social trust (a). user-user interest similarity (b). social trust graph (c). proposed social graph.

minimizing the external edge. Mostly, the current algorithms either select the initial centers randomly or use a higher degree, but mostly the high degree nodes have many common nodes. We proposed the initial nodes selection algorithm that have high weighted degree and each center node should be at lease 2-hops away from each other considering the nodes selected by high degree centrality have a large number of common nodes [50] and the Facebook anatomy [49] (that user can reach to another user an average 4.7 hops). Going forward [50], we first rank the nodes by weighted degree and then we select the initial community centers from the highly ranked nodes that are at least 2-hops from each other. Eq. (5) finds the node ranking centrality:

$$NC_k = \sum_{l \in K} (w_{kl} + w_{lk}) \tag{5}$$

where NC_k is the centrality ranking value of node k, and K are the first neighbors of k, while w_{kl} is the weighted out-degree and w_{lk} is the weighted in-degree. After ranking the nodes, we select the initial nodes as communities among the high ranked nodes that should be at least 2-hops from each other. Algorithm 1 provides

the initial centers selection mechanism in the weighted modeled directed graph.

3.4.2. Community clustering using the modeled weighted directed graph

To cluster the user interests and trust based modeled edges into communities, we used similarity function for assigning a node to the community cluster using Eq. (6).

$$S_{kc} = \gamma \frac{\sum_{j \in co_{nodes}} (w_{jk} + w_{kj})}{\sum_{l \in K} (w_{kl} + w_{lk})} + (1 - \gamma) \frac{|co_{nodes}|}{|K|} \quad (6)$$

S_{kc} is the similarity value of the node k to the community c and has value from [0, 1]. co_{nodes} are the nodes from the community that share edges with the node k, and γ is the tunable parameter needed for incorporating the edge weights and topology. Similarity function introduced two main parts. $\frac{\sum_{j \in co_{nodes}} (w_{jk}+w_{kj})}{\sum_{l \in K} (w_{kl}+w_{lk})}$ is the ratio of weighted edges a node shared with a community. A node with highest ratio with a community will have higher probability to be assigned to that community. $\frac{|co_{nodes}|}{|K|}$ is the ratio of topology the node k shared with a community.

We introduced T_{sim}(Similarity Threshold) for not assigning a node directly to a community. Existing algorithms directly assign the node to a community although their similarity may be 0.001. The similarity threshold $S_{kc} \geq T_{sim}$ selects the candidate communities if applies and among them a community that node has higher similarity with it. We use Eq. (7) as cluster quality function (Q) of maximizing the internal weighted edge density within a community and minimizing the edges among the communities. Q is the ratio of communities' internal weighted edges densities to the total weighted edges density.

$$Q = \sum_{c=1}^{C} \frac{\sum_{i}^{K_c} \sum_{j}^{K_c} w_{ij}}{W} \quad (7)$$

To find better community clusters, the Q value should be maximized until Q converges (no further changes). C represents the total number of communities, K_c is the number of nodes in a community, and w_{ij} is the weight of an internal edge within a community connecting nodes i and j – i and j should be within a single community. Fig. 5 shows the pictorial overview of the proposed community clustering algorithm. Fig. 5a shows some intermediate steps toward community clustering, and C = 2 represents two communities. $K_1 = \{1, 2, 3, 4, 5\}$ has 5 nodes and $K_2 = \{6, 7, 8\}$ has 3 nodes, and $Q = 0.72$. Fig. 5b shows the assignment of $node = \{9\}$ to one of the communities (either $c1$ or $c2$) based on the node similarity threshold and we assume $T_{sim} \geq .7$ in this figure and the node similarity Eq. (6). $S_{91} = .9 * (\frac{1.1}{1.1+1.5+1.5}) + .1 * (\frac{1}{3}) = 0.2748$ and $S_{92} = .9 * (\frac{3.0}{1.1+1.5+1.5}) + .1 * (\frac{2}{3}) = 0.7252$, $S_{91} < T_{sim}$ and $S_{91} > T_{sim}$. As a result, we assigned the $node = \{9\}$ to $c2$ and the value of Q is updated and $Q = 0.8828$. Algorithm 2 shows the procedure of the proposed community clustering based on the modeled weighted directed graph, which is in turn based on the users' interests weighted by the social trust. The algorithm continues to assign nodes to different communities until one of these three conditions are achieved (i) No node remains to assign it to a community ii) The maximum iterations specified are achieved iii) The cluster quality function Q converges (means its value become constant and does not change further.)

3.4.3. Merging of detected communities

We extended the community merging of [5]. First, we selected the initial communities approximately 2 to 3 time more than the required number of communities to detect. Initial communities are obtained using the proposed community clustering Algorithm 2. After detecting the initial communities first we check, is there any community that has only one node and we remove that community and then using the Step 2-Step 4 assign that nodes to the community with whom it has higher similarity. After that, we merge the communities to obtain the required number of communities, and we perform the merging recursively. We perform the community merging by first finding the merging cost using Eq. (8).

$$\rho_{pq} = \frac{\sum_{w \in (p \cap q)} w_{pq}^{out}}{min(\sum_{w \in p} w_{p}^{in}, \sum_{w \in q} w_{q}^{in})} \quad (8)$$

where in the Eq. (8), the numerator shows the sum of the weighted edges that connect the two communities' p and q means the external edges. The denominator shows the internal edges weighted sum of the communities and take the minimum of them. We sort the merging communities by the merging cost in descending order and select the first one and merge these two communities. The communities merging procedure continues until getting the required number of communities.

4. Results and discussion

We evaluated the proposed algorithm using a publicly available dataset, and compared our results with state of the art algorithms such as KM clustering [51], Modularity-based communities [52], and Girvan Newman Clustering [42]. For modeling and testing of the algorithms, we also used two publicly available datasets and details of the datasets are given in the below subsection.

4.1. Dataset description and modeling/preprocessing of the dataset

The details of the dataset we used in the simulation appear in Table 2. We did not consider the trust and interests loops in this paper. In the dataset, every user did not rate every movie and also did not provide the trust on every other user.

On FilmTrust-Movies website, a user provides trust rating when he adds someone as a friend. When users are providing trust, they are advised to rate their friends trust about movies. To model the users' interests-based social weighted undirected graph, we consider that at least 3-movies should be rated in common between two users in the FilmTrust Dataset and 10-movies in the CiaoDVD dataset. The directed weighted graph is modeled considering $\beta = 0.6$.

4.2. Result of community clustering based on trust weighted by the interests modeled weighted directed graph

The community clustering algorithm mostly selects the initial centers either randomly or the nodes of the highest degree as described in [5]. But most of the time, the nodes selected as initial centers overlap each other without helping to converge the clusters' quality function in less processing time, and also to produce higher quality clusters. Tables 3 and 4 shows the Top-10 ranked nodes in the FilmTrust Movies and CiaoDVD dataset respectively in the Trust network dataset. In the proposed initial communities' center selection, when the highest weighted degree is selected as initial community center its direct (first) neighbors are not considered for center selection. Tables 3 and 4 clearly depicts that the Top-10 initial communities' centers selected removed the direct neighbors and each of the initial center is 2-hops from each other. Fig. 6 shows the first neighbors of the node id-509 in the FilmTrust-Movies. It clearly depicts that the node-509 has the highest weighted degree and topology. Since the node id-188 and node-628 are the direct neighbors of it and not 2-hops away so they are not taken as initial communities' centers and similarly the other nodes should fulfil the Algorithm 1. It clearly shows that most Top-10 ranked nodes in the modeled weighted directed graph

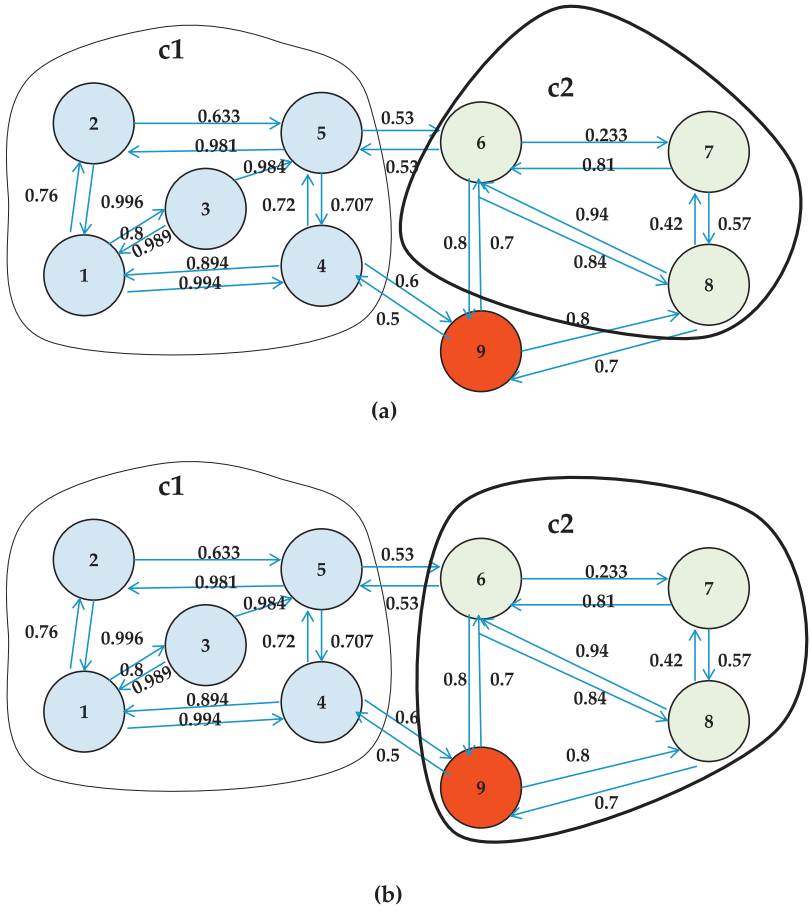


Fig. 5. Proposed community clustering based on modeling user interests weighted by trust (a) depicts two communities with internal and external edges (b) depicts node assignment to a community (c1 or c2).

Table 2.
Datsets statistics and parameter description.

Datsets	User interests ratings information			User trust information		
	Users	Items	Rating scales	Users	Links	Trust scale
FilmTrust-Movies [53]	1508	2071	[0.5, 4.0]	1642	1853	{0,1}
CiaoDVD-Movies [54]	17,615	16,121	[1,5]	4658	40,133	{0,1}

Table 3.
(FilmTrust Dataset): Top-10 ranked nodes (nodes ID) in the Trust Dataset and the modeled directed weighted graph of trust weighted by interest and the initial communities centers nodes ID.

Rank	Top-10 nodes in Trust Dataset	Top-10 nodes in proposed modeled graph	Top-10 initial centers to initialize community clustering (2-hops away)
1	509	509	509
2	188	188	938
3	628	628	79
4	29	1398	433
5	1398	546	272
6	433	1147	452
7	546	1187	1212
8	436	436	918
9	1147	716	1159
10	716	965	319

Table 4.
(CiaoDVD Dataset): Top-10 ranked nodes (Nodes ID) in the Trust Dataset and the modeled directed weighted graph of trust weighted by interest and the initial communities centers nodes ID.

Rank	Top-10 nodes in Trust Dataset	Top-10 nodes in proposed modeled graph	Top-10 initial centers to initialize community clustering (2-hops away)
1	17,750	17,750	17,750
2	8076	8076	5228
3	5228	5228	356
4	12,239	1906	1422
5	1906	12,239	1973
6	3377	3377	17,764
7	4613	4613	3568
8	13,204	1607	897
9	3484	13,204	20,235
10	17,923	3484	1257

are distinct when considering the direct trust. The initial centers nodes selected by the proposed algorithm can minimize the overlapped nodes in the initial nodes assignment, expediting cluster convergences. Fig. 7 visualizes the Top-10 nodes selected as initial community centers to cluster the users in the datasets of FilmTrust and CiaoDVD, respectively.

Fig. 8 shows the performance comparison of the proposed community clustering with the KM Clustering. KM clustering method

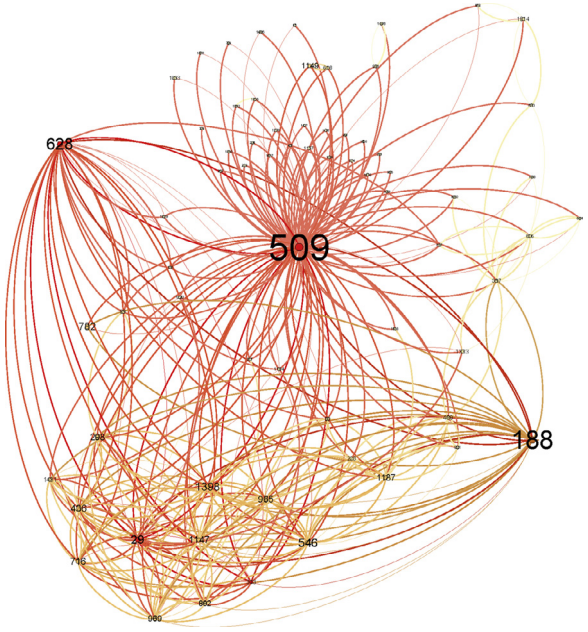


Fig. 6. Overview of the direct (first) neighbors of the Node Id-509 in the FilmTrust-Movies modeled weighted directed graph.

is based on node weighting by density of local neighborhood and outward traversal from a locally dense seed to isolate the dense regions according to given parameters. It works in three stages i) node weighting ii) complex prediction and iii) optionally post-processing for filtering or adding node in the resulting complexes through certain connectivity criteria. The node weighting mechanism, weights all nodes based on their local network density using the highest k-core of the vertex neighborhood. The complex prediction takes the vertex weighted graph as an input, seeds a complex with the highest weighted node and recursively moves outward from the seed node, including nodes in the complex whose

Algorithm 1. initial centers selection for communities detection in modeled users' interest weighted by Trust.

Input: ***G(U,V,W)***-modeled directed weighted graph of users' interest weighted by trust, ***n***- number of initial centers of communities,
Output: **C**-list of nodes ids as initial communities centers

Procedure:
Step 1: Ranking nodes in **G**
-**Compute** the weighted degree of each node in **G** using Eq. (5)
-**SL-** Sorted list of the nodes by weighted degree in descending order
-**C** $=\varphi$- initialize the list of initial communities centers
Step 2: Finding the list of initial communities centers
-**While** (length(C) < n) **do**
-$l = \max(SL)$
-$C = l \cup C$
-$l\_neighbors$ = find the first neighbors of l
-$SL = SL - l - l\_neighbor$
- **End While**

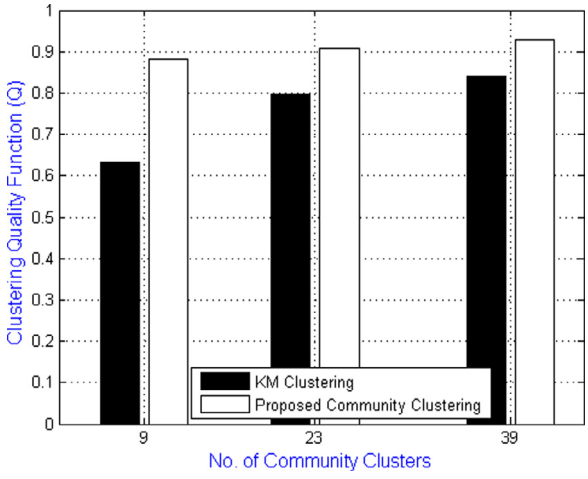
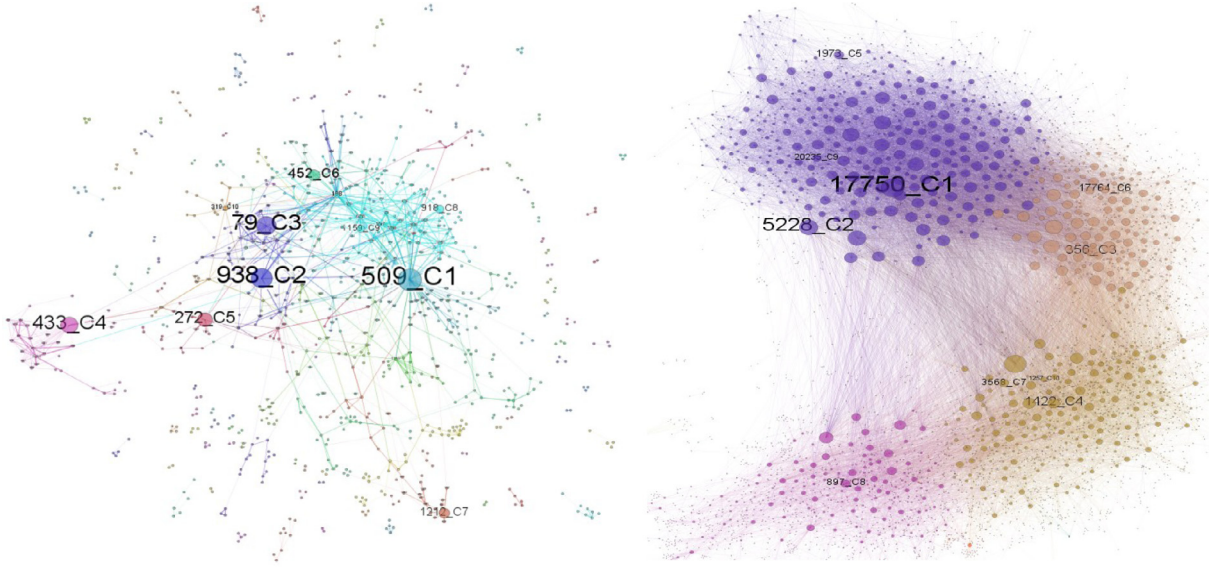


Fig. 8. Clustering quality performance comparison of the proposed community clustering with the KM clustering for the modeled weighted directed network of FilmTrust Dataset.



(a) **(b)**

Fig. 7. Pictorial overview of the initial community centers' selection by the suggested algorithm (a). Top-10 initial communities centers in FilmTrust Dataset (b). Top-10 initial communities centers in CiaoDVD Dataset.

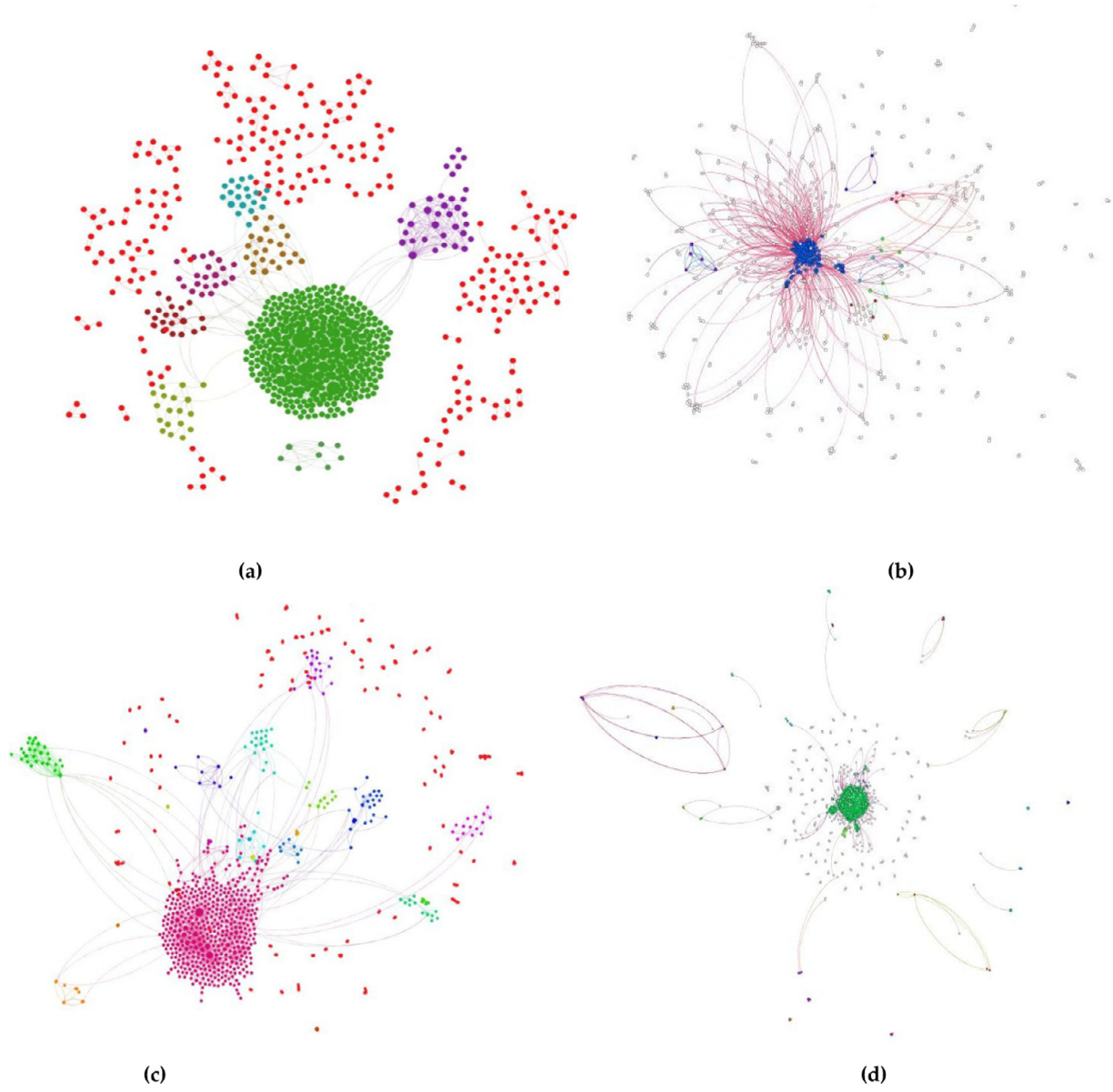


Fig. 9. Illustration of communities identified in the FilmTrust Dataset (a,c). 9 and 23 communities (clusters) identified by the proposed community clustering and red color nodes are not assigned to any cluster. (b,d) 9 and 23 communities (clusters) identified by the KM clustering and white color nodes with black outer are not assigned to any cluster. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

weight is above a given threshold, which is a given percentage away from the weight of the seed node. The post-processing filtered out the complexes if at least 2-core (graph of minimum degree 2) are not contained in the complexes. The KM clustering considers mainly the *k*-core value and other parameters to find the clusters. We used the K-core value {3, 2 and 1} and detect the {9, 23 and 39} clusters respectively. We used Eq. (7) as acluster quality function that considers the ratio of the communities' internal edges weights to all the edges weights in the data set. In the proposed community clustering, we first find the numbers of initial communities centers approximately three times of the communities' required to detect and then merge the communities to get the required number of communities. To detect the number of clusters {9, 23 and 39}, the initial number of clusters are {27, 69 and 117} respectively, and $\gamma = 0.8$. Fig. 8 clearly depicts the proposed community clustering have better performance for all settings of the cluster detection. Fig. 9 illustrates to visualize the clusters in the FilmTrust Dataset. It clearly shows that he proposed community clustering have denser and better clusters. We used Gephi [55] for visualization and finding the KM and modularity based clustering.

Fig. 10 shows the performance comparison of the proposed community clustering with the modularity based community clustering in the FilmTrust and CiaoDVD datasets. In the FilmTrust Dataset, for the modularity based community clustering, we used the randomized, edge weights and resolution {1, 0.8, 0.5 and 0.4} that produces the number of clusters {110, 113, 120 and 130} respectively. For the proposed community clustering we used 300 initial clusters centers to detect the communities and then using merging of communities we detect the required communities {110, 113, 120 and 130}. In the CiaoDVD dataset, for the modularity based clustering, we used resolution {1, 0.8, 0.6 and 0.5} that produces the number of clusters {52, 65, 68 and 70} respectively. For the proposed community clustering, we used the 175 initial communities and then using the merging we find the communities {52, 65, 68 and 70} respectively. Fig. 10(a) and (b) clearly depicts that

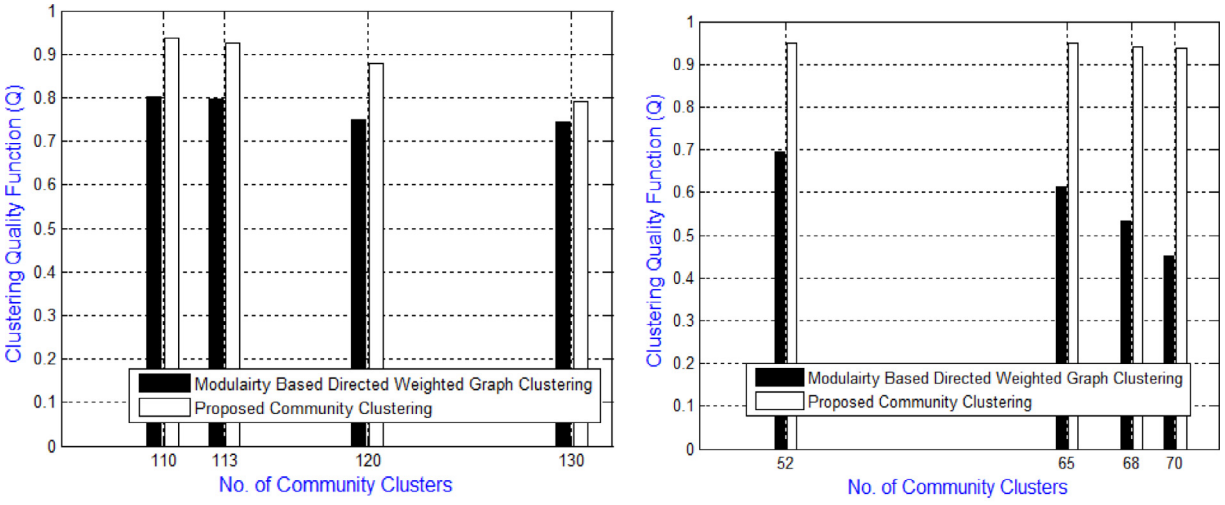


Fig. 10. Clustering quality performance comparison of the proposed community clustering with the modularity based clustering (A). FilmTrust Dataset (b). CiaoDVD Dataset.

Algorithm 2. Community clustering based on modeled users' interest weighted by Trust.

Input: ***G(U,V,W)***-modeled directed weighted graph, *C*- number of communities, T_{sim}- similarity threshold to assign node to a community, **max_iter**- maximum number of iteration
Output: c1,c2,——-,cn- n communities label for the graph **G**

Procedure:
Step 1: Initial communities centers with natural community selection
-**Select** the initial community centers with natural communities using Algorithm 1
Step 2: Node assignment to the Community
-**Find** node sets ***S*** that share edges with communities but not in any community until now.
-**for** every node ***k*** in ***S*** **do**
- **for** ***c*** in ***C*** **do**
-Find the node ***k*** similarity (S_{kc}) with the community ***c*** using Eq. (6)
-**if** ($S_{kc} \geq T_{sim}$) **do**
-**Select** ***c*** as a candidate community and add it to ***candid_communities***
-**Add** S_{kc} to a ***temporary_ similarity_variable***
-**end if**
-**end for**
-**Sort** the ***candid_communities*** in ascending order w.r.t S_{kc} of ***temporary_similarity_variable.***
-**Select** the first community ***c*** in the sorted list and assign ***k*** to the ***c***.
-**end for**
Step 3: Check the community cluster quality
-**Find** the community cluster quality (**Q**) using Eq. (7)
Step 4: ***Repeat*** *Steps 2–3* ***Until***
-***Q*** *is converged* || **max_*iter*** *is achieved* || ***no further node*** *to assign it to a community*

the proposed community clustering have better performance and detect better communities' clusters.

5. Conclusion

This paper proposed a method of community clustering for implicit communities detection based on the trust modeling weighted by the user's interest in a given OSN. First, we modeled the directed weighted network by incorporating the user's interest similarity and modeled trust. We proposed the probabilistic trust model to predict the unknown trust between users because most of the time, the user did not provide trust value explicitly. We predicted the user-user interest similarity based on their preferences and the content experienced in the social network. Second, we clustered the users in the trust weighted by user interest modeled network; this grouped the users with higher trust and greater interest similarity. The proposed clustering algorithm first ranked the nodes by the weighted degree and selected the community centers that are not direct neighbors, thereby minimizing the clustering convergence time and providing a way to find efficient community clusters. We assigned the user to the community having higher trust and interest weighted similarity and higher common nodes of topological connections. We then evaluate the clustering quality of the proposed community clustering with other well-known approaches for two publicly available datasets, and the results showed that the proposed community clustering algorithm efficiently analyzes the community structure by exhibiting a higher weighted community subgraph within a community and lower one for other communities.

Acknowledgment

This research was supported by Basic Science Research Programs through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2014R1A1A2056357 and NRF-2017R1A2B1010817).

References

[1] Goldberg D, et al. Using collaborative filtering to weave an information tapestry. Commun ACM 1992;35(12):61–70.
[2] Schafer JB, et al. E-Commerce recommendation applications. Data Min Knowl Discov 2001;5(1-2):115–53.
[3] Ullah F, Sarwar G, Lee S. N-Screen aware multicriteria hybrid recommender system using weight based subspace clustering. Sci World J 2014;2014:11 Article ID 679849pages. doi:10.1155/2014/679849.
[4] Bobadilla J, et al. Recommender systems survey. Knowl Based Syst 2013;46(0):109–32.
[5] Feng H, et al. Personalized recommendations based on time-weighted overlapping community detection. Inf Manage 2015;52(7):789–800.
[6] Fatemi M, Tokarchuk L. A community based social recommender system for individuals & groups. In: Social computing (SocialCom), 2013 international conference on. IEEE; 2013. p. 351–6.
[7] Fortunato S. Community detection in graphs. Phys Rep 2010;486(3):75–174.
[8] Wang Y, et al. The impact of sellers' social influence on the co-creation of innovation with customers and brand awareness in online communities. Ind Marketing Manage 2016;54:56–70.
[9] Westlake BG, Bouchard M. Liking and hyperlinking: Community detection in online child sexual exploitation networks. Soc Sci Res 2016;59:23–36.
[10] Guo J, et al. Bacterial communities in water and sediment shaped by paper mill pollution and indicated bacterial taxa in sediment in Daling river. Ecol Indic 2016;60:766–73.

[11] Xu X, He P. Improving clustering with constrained communities. Neurocomputing 2016;188:239–52.
[12] Chen S, et al. Cluster-group based trusted computing for mobile social networks using implicit social behavioral graph. Future Gener. Comput. Syst. 2016;55:391–400.
[13] Abdul-Rahman A, Hailes S. Supporting trust in virtual communities. system sciences, 2000. In: Proceedings of the 33rd annual Hawaii international conference on. IEEE; 2000.
[14] Bedi P, et al. Trust based recommender system for semantic web. IJCAI. 2007.
[15] Kim YA, Phalak R. A trust prediction framework in rating-based experience sharing social networks without a web of trust. Inf Sci 2012;191(0):128–45.
[16] Abrahams AS, et al. An integrated text analytic framework for product defect discovery. Prod Oper Manage 2015;24(6):975–90.
[17] Liao C-L, Lee S-J. A clustering based approach to improving the efficiency of collaborative filtering recommendation. Electron Commerce Res Appl 2016.
[18] Ortega F, et al. Recommending items to group of users using matrix factorization based collaborative filtering. Inf Sci 2016;345:313–24.
[19] Sarwar B, et al. Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web, ACM; 2001.
[20] Rashid AM, Lam SK, et al. ClustKNN: a highly scalable hybrid model-& memory-based CF algorithm. Proceeding of WebKDD 2006.
[21] Lathia N, et al. Trust-based collaborative filtering. In: Trust management II. Springer; 2008. p. 119–34.
[22] Koren Y, et al. Matrix factorization techniques for recommender systems. Computer 2009;8:30–7.
[23] Shi Y, et al. List-wise learning to rank with matrix factorization for collaborative filtering. In: Proceedings of the fourth ACM conference on recommender systems. ACM; 2010.
[24] Bobadilla J, et al. Recommender systems survey. Knowl Based Syst 2013;46(0):109–32.
[25] Miyahara K, Pazzani MJ. Collaborative filtering with the simple bayesian classifier. PRICAI 2000 topics in artificial intelligence: 6th Pacific Rim international conference on artificial intelligence; 2000.
[26] Lee M, Choi P, Woo Y. A hybrid recommender system combining collaborative filtering with neural network. In: Adaptive hypermedia and adaptive web-based systems. Berlin Heidelberg: Springer; 2002. p. 531–4.
[27] Roh TH, et al. The collaborative filtering recommendation based on SOM cluster-indexing CBR. Expert Syst Appl 2003;25(3):413–23.
[28] Balabanović M, Shoham Y. Fab: content-based, collaborative recommendation. Commun ACM 1997;40(3):66–72.
[29] Su J. Content based recommendation system. U.S. Patent 9,230,212, issued January 5, 2016.
[30] Ferman AM, et al. Content-based filtering and personalization using structured metadata. In: Proceedings of the 2nd ACM/IEEE-CS joint conference on digital libraries. Portland: Oregon, USA, ACM; 2002. p. 393.
[31] Deldjoo Y, et al. Content-based video recommendation system based on stylistic visual features. J Data Semant 2016:1–15.
[32] Kim K-R, Moon N. Recommender system design using movie genre similarity and preferred genres in SmartPhone. Multimedia Tools Appl 2012;61(1):87–104.
[33] Vozalis MG, Margaritis KG. Using SVD and demographic data for the enhancement of generalized collaborative filtering. Inf Sci 2007;177(15):3017–37.
[34] Ullah F, Sarwar G, Lee SC, Park YK, Moon KD, Kim JT. Hybrid recommender system with temporal information. In: Information networking (ICOIN), 2012 international conference on. IEEE; February, 2012. p. 421–5.
[35] Burke R. Hybrid recommender systems: survey and experiments. User Model User Adapted Interact 2002;12(4):331–70.
[36] Yan Z, Holtmanns S. Trust modeling and management: from social trust to digital trust. IGI Global 2008:290–323.
[37] Burns C, Mearns K, McGeorge P. Explicit and implicit trust within safety culture. Risk Anal 2006;26(5):1139–50.
[38] Artz D, Gil Y. A survey of trust in computer science and the semantic web. Web Semantics 2007;5(2):58–71.
[39] Kim YA, Song HS. Strategies for predicting local trust based on trust propagation in social networks. Knowl Based Syst 2011;24(8):1360–71.
[40] Newman ME. Fast algorithm for detecting community structure in networks. Phys Rev E 2004;69(6):066133.
[41] Newman ME. Modularity and community structure in networks. Proc Natl Acad Sci 2006;103(23):8577–82.
[42] Girvan M, Newman MEJ. Community structure in social and biological networks. Proc Natl Acad Sci 2002;99(12):7821–6.
[43] Kim P, Kim S. Detecting overlapping and hierarchical communities in complex network using interaction-based edge clustering. Physica A 2015;417:46–56.
[44] Jin X, et al. Coupling effect of nodes popularity and similarity on social network persistence. Scientific Reports 2017;7.
[45] Jin D, et al. Detect overlapping communities via ranking node popularities Thirtieth AAAI conference on artificial intelligence; 2016.
[46] Palla G, Derényi I, Farkas I, Vicsek T. Uncovering the overlapping community structure of complex networks in nature and society. Nature 2005;435(7043):814–18.
[47] Eustace J, et al. Community detection using local neighborhood in complex networks. Physica A 2015;436:665–77.
[48] Zhang J, Cohen R. A comprehensive approach for sharing semantic web trust ratings. Comput Intell 2007;23(3):302–19.
[49] Ugander J, Karrer B, Backstrom L, Marlow C. The anatomy of the facebook social graph. arXiv preprint arXiv:1111.4503 (2011).
[50] Sheikhahmadi A, et al. Improving detection of influential nodes in complex networks. Physica A 2015;436:833–45.
[51] Bader GD, Hogue CWV. An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics 2003;4(1):1.
[52] Blondel VD, et al. Fast unfolding of communities in large networks. J Stat Mech 2008;10(2008):P10008.
[53] Guo G, Zhang J, Yorke-Smith N. A novel Bayesian similarity measure for recommender systems. IJCAI; 2013.
[54] Guo G, Zhang J, Thalmann D, Yorke-Smith N. Etaf: an extended trust antecedents framework for trust prediction. In: Advances in social networks analysis and mining (ASONAM), 2014 IEEE/ACM international conference on. IEEE; 2014. p. 540–7.
[55] Bastian M, Heymann S, Jacomy M. Gephi: an open source software for exploring and manipulating networks. ICWSM 8 (2009).
[56] Bai L, Cheng X, Liang J, Guo Y. Fast graph clustering with a new description model for community detection. Inf Sci 2017.
[57] Wang X, Liu G, Li J, Nees JP. Locating structural centers: a density-based clustering method for community detection. PLoS One 2017;12(1):e0169355.
[58] Rapoport A. Spread of information through a population with socio-structural bias: I. Assumption of transitivity. Bull Math Biol 1953;15(4):523–33.
[59] Bianconi G, Darst RK, Iacovacci J, Fortunato S. Triadic closure as a basic generating mechanism of communities in complex networks. Phys Rev E 2014;90(4):042806.
[60] Battiston F, Iacovacci J, Nicosia V, Bianconi G, Latora V. Emergence of multiplex communities in collaboration networks. PLoS One 2016;11(1):e0147451.
[61] Massa P, Salvetti M, Tomasoni D. Bowling alone and trust decline in social network sites. In: Dependable, autonomic and secure computing, 2009. DASC'09. eighth IEEE international conference on. IEEE; 2009. p. 658–63.