Open access • Journal Article • DOI:10.1145/2501977

Cluster-Based Collaborative Filtering for Sign Prediction in Social Networks with Positive and Negative Links — Source link

Amin Javari, Mahdi Jalili

Institutions: Sharif University of Technology

Published on: 30 Apr 2014 - ACM Transactions on Intelligent Systems and Technology (ACM)

Topics: Collaborative filtering, Social network analysis, Sign (mathematics) and Cluster analysis

Cluster-based Collaborative Filtering for Sign Prediction in Social Networks with Positive and Negative Links

AMIN JAVARI, Sharif University of Technology
MAHDI JALILI, Sharif University of Technology

Social network analysis and mining get ever-increasing importance in recent years, which is mainly due to availability of large datasets and advances in computing systems. A class of social networks is those with positive and negative links. In such networks, a positive link indicates friendship (or trust), whereas links with negative sign correspond to enmity (or distrust). Predicting sign of the links in these networks is an important issue and has applications such as friendship recommendation and identifying malicious nodes in the network.

In this manuscript, we proposed a new method for sign prediction in networks with positive and negative links. Our algorithm is based on, first, clustering the network into a number of clusters, and then, applying a collaborative filtering algorithm. The clusters are such that the number of inner-cluster negative links and inter-cluster positive links are minimal, i.e., the clusters are socially balanced as much as possible (a signed graph is socially balanced if it can be divided into clusters with all positive links inside the clusters and all negative links between them). We then used similarity between the clusters (based on the links between them) in a collaborative filtering algorithm. Our experiments on a number of real datasets showed that the proposed method outperformed previous methods including those based on social balance and status theories and the one based on machine learning framework (logistic regression in this work).

Categories and Subject Descriptors: **G.2.2 [Graph Theory]: Graph algorithms; H.3.3 [Information Storage and Retrieval]**: Information Filtering, Information Search and Retrieval

General Terms: Algorithms; Experimentation; Performance

Additional Key Words and Phrases: Social networks, signed networks, collaborative filtering, cluster identification, social balance theory, social status theory

ACM Reference Format:

Amin Javari and Mahdi Jalili, 2013. Cluster-based collaborative filtering for sign prediction in social networks with positive and negative links. *ACM Transaction on Intelligent Systems and Technology.*

1. INTRODUCTION

Due to tremendous development of the Internet, World Wide Web and social networks, network-centric mining and analysis has become an emerging field in science and engineering. These analysis are based on constructing graphs of social systems in which the nodes represent the individuals (or organizations) and the links represent the relations between them [Boccaletti et al. 2006; Strogatz 2001]. It was shown that many real networks show a number of common structural properties such as small-worldness [Watts and Strogatz 1998], scale-free degree distribution

Author's addresses: A. Javari (javari@ce.sharif.edu) and M. Jalili (mjalili@sharif.edu), Department of Computer Engineering, Sharif University of Technology, Tehran, Iran.

DOI:http://dx.doi.org/10.1145/0000000.0000000

[Barabasi and Albert 1999], densification and shrinking diameter [Leskovec et al. 2007]. Furthermore, many social networks show properties such as community structure and modularity [Girvan and Newman 2002; Newman and Park 2003].

Most of social network models consider only the presence/absence of a connection between the nodes in constructing the network structure, and ignore the possible sign of the connection. In many real social networks, however, the relations between individuals might be positive or negative [Cartwright and Harary 1956; Leskovec et al. 2010]. Positive relations mean friendship (trust or voting in favor), whereas negative links correspond to enmity (distrust or voting against). One can name Wikipedia, Epinions and Slashdot as examples of networks with positive and negative links [Brzozowski et al. 2008; Burke and Kraut ; Kunegis et al. 2009; Massa and Avesani 2005]. Social networks are highly dynamic and their structure goes under heavy changes through appearance/disappearance of new nodes and edges [Holme and Saramäki 2012; Kossinets and Watts 2006; Kumar et al. 2006]. Temporal networks, i.e., networks whose topology and link formation change over time, has been shown to have significantly different dynamic behavior as compared to static networks [Holme and Saramäki 2012]. A possible motivator to change links in a social network would be to improve a utility function such as cooperativity among the nodes [Perc 2009; Perc and Szolnoki 2010].

In network science, this issue is often studied under a topic under “link prediction”, which refers to reliably predicting the future links that are likely to appear in the network [Liben-Nowelly and Kleinberg 2003; Lü and Zhou 2011]. The aim of the link prediction problem is to use the information available from the structure of the network, i.e., the links that the nodes already have in the network, and extract a number of features. The features can be linked to both local and global properties of networks. These features are then used in a classification/prediction model to predict the future links to come.

In signed networks, not only the link prediction problem is important, but also correctly predicting the sign of the links is an important issue. In sign prediction problem, local and global structural properties of the network are used to determine an unknown sign of an existing link between two nodes [Guha et al. 2004; Leskovec et al. 2010; Shahriari et al. 2012]. Sign prediction has many potential applications in social networks. for example, prediction of trust between two users can be used as a similarity measure in recommender systems [O'Donovan and Smyth 2005]. In some cases, the signs might be determined by malicious users; sign prediction problem can identify such activities and purify the network. There are, in general, two approaches for the sign prediction problem; those using machine learning techniques and those trying to do the prediction without learning. In methods based on a machine learning framework, first, a number of meaningful structural features are extracted, and then, a classifier is used to solve the prediction problem [Chiang et al. 2011; DuBois et al. 2011; Leskovec, Huttenlocher and Kleinberg 2010]. Another class of algorithms tries to use available theories in social science to do the prediction task without performing a learning process. For example, one may use social balance and status theory [Guha, Kumar, Raghavan and Tomkins 2004; Heider 1946] in order to predict the signs. The logic behind such techniques is that the networks are evolved such that they become more balanced or better support status theory [Antal et al. 2005; Leskovec, Huttenlocher and Kleinberg 2010]. The results of these two approaches cannot be directly compared, since applying machine learning based techniques is always costly, whereas computations based on social balance and status theories are much simpler. While methods based on machine learning framework cannot be applied to very large graphs, those based on social science theories can be simply applied.

In this manuscript, we use collaborative filtering approach [Manos Papagelis 2005; Sarwar et al. 2001] for sign prediction. Collaborative filtering algorithm and its variants are the most frequently used methods in recommender systems. In a users-items network, these algorithms recommend a list of items for each user such that he/she is likely to use (or positively rate) such items. We consider the sign prediction problem as the problem of predicting votes among the nodes, where each node can be simultaneously user (the links are pointed out from users) and item (the links are pointed to the items).

A common drawback of collaborative filtering algorithms is their rather poor performance for networks with high levels of sparsity, which is the case for many real social networks [Huang et al. 2004; Xue et al. 2005; Yildirim and Krishnamoorthy 2008]. In order to overcome the sparsity problem, we use clustering techniques, that is, first the signed network is divided into a number of clusters such that the highest level of balanced-ness is obtained. This can be achieved by trying to maximize the number of positive links inside the clusters and those with negative sign inter them [Bo et al. 2007; Bogdanov et al. 2010; Doreian et al. 2005; Doreian and Mrvar 1996]. Then, each cluster is considered as an individual and the collaborative filtering is applied on them by finding similarities between the clusters. We applied the proposed method on a number of real signed social networks. Our method being much more scalable and less costly than those based on machine learning framework, outperformed them in terms of prediction accuracy. The proposed method also showed better performance than those based on social theories; however, these methods are less complex than ours. Our key contributions in this manuscript are as follows:

- Modeling the signed network as a bipartite users-items network that is commonly used in recommendation systems
- Applying user-based collaborative filtering for sign prediction and modifying it to make it appropriate for sign prediction problem
- Solving the sparsity problem of signed network by clustering the network
- Introducing a method to extract conditional similarities between clusters in signed networks

2. SIGN PREDICTION IN NETWORKS WITH POSITIVE AND NEGATIVE LINKS

2.1 Sign prediction problem

In order to formally define the sign prediction problem, let us consider a signed directed graph $G(V,E)$ where V indicates the set of nodes and E the set of signed relations between them (for binary signed graphs, the entries of E would be +1 or –1). In the sign prediction problem, we assume that all structural information about graph G is given, except the sign of the edge (u,v) from node u, denoted as trustor, to node v, denoted as trustee. The problem is to infer the disappeared sign of the edge (u,v) based on the information extracted from the rest of the graph. Most of previous works in sign prediction are based on various machine learning based techniques [Chiang, Natarajan, Tewari and Dhillon 2011; DuBois, Golbeck and Srinivasan 2011; Leskovec, Huttenlocher and Kleinberg 2010; Shahriari, Askari, Gharibshah and Jalili 2012]. There have also been some efforts to use two psychological theories, namely, social balance and status theories, for predicting the sign of the links. In the sequel, we give some brief description of these methods.

2.2 Social balance theory and sign prediction problem

The *social balance* theory is a theory from social psychology to show that how people develop their relations in a singed network [Heider 1946]. This theory is based on principles from psychology such as "*my friend's friend is my friend", "my friend's enemy is my enemy", "my enemy's friend is my enemy" and "my enemy's enemy is my friend*". Social balance theory is often studied in all-to-all connected signed triads consisting of nodes u, v and w. The network will be a balanced structure if we have odd number of occurrences for positive edges (for example, when all three connections are positive). For general networks, it is argued that the network is balanced if all its triads are balanced. It is simple to show that balanced networks have cycles with only odd number of positive connections. However, real networks are not completely balanced and they are indeed balanced to some extent [Leskovec, Huttenlocher and Kleinberg 2010]. It can also be shown that if a network is structurally balanced, it is possible to cluster it into a number of sub-networks such that all the links inside the clusters are positive, whereas the inter-cluster links are all negative.

Social balance theory may be used for predicting the unknown sign of the connection links [Leskovec, Huttenlocher and Kleinberg 2010]. Indeed, real networks may evolve in a way that they get more structurally balanced, that is, the sign of a link is determined such that the triads in which the link is participating, are structurally balanced. However, the link (u,v) is likely to participate in multiple triads for which some of them will be balanced, whereas others are unbalanced, for any sign for the link. In such situations, we choose the sign resulting in more balanced triads. More precisely, considering

$$F_{balance} = Sign\left(\sum_{i=1}^{n}\left(Sign\left(u,w_i\right)\times Sign\left(w_i,v\right)\right)\right), \quad (1)$$

if $F_{balance} > 0$, we choose "+" sign and if $F_{balance} < 0$, we choose "–" sign for the edge (u,v). $Sign(u,v)$ indicates the sign of the link from node u to node v. This theory does not predict anything for the case where $F_{balance} = 0$. In the above formula, w_i represents the node that form triad with u and v, and n indicates the number of such nodes.

2.3 Social status theory and sign prediction problem

Social balance theory does not consider directions in the connections, and thus, is less likely to work for directed networks. Another theory from social psychology that specifically deals with directed signed graphs is *social status* theory. This theory defines an status value for each node and a positive edge from u to v means that v has higher status compared to u, whereas a negative edge in the same direction means a higher status for u than v. Obviously, based on social status theory, when direction of a link changes, its sign also changes. In other words, a positive link from u to v means negative link from v to u. One can also use this theory for predicting the signs in real networks, which in most cases, are directed [Leskovec, Huttenlocher and Kleinberg 2010]. To this end, an appropriate measure for the status of the nodes should be first adopted. Then, having the status values for the nodes u and v, for which the sign is unknown for the link from u to v, it will be "+" if the status of v is higher than u and "–" if u is of higher status than v. In practical applications, in order to predict the sign of the link from u to v, we consider all nodes w_i that are connected to both u and v. The direction of the links (u,w_i) and (w_i,v) are changed in a way such that it is from u to w_i and from w_i to u, respectively (if this makes the change of direction of a link in the original graph, the sign of the link also changes). Then, F_{status} is computed as

$$F_{status} = Sign\big(Sign(u, w_i) + Sign(w_i, v)\big). \quad (2)$$

When $F_{status} > 0$, the sign is predicted to be "+", whereas $F_{status} < 0$ means "–" as the predicted sign of the link (u,v). Similar to balance theory, status theory also works on single triads. In the case the link (u,v) participates in multiple triads, the following term can be calculated instead of the above formula.

$$F_{status} = Sign\left(\sum_{i=1}^{n}\big(Sign(u,w_i)+Sign(w_i,v)\big)\right). \quad (3)$$

As it is seen from equations (1) and (3), in order to be able to use social balance and status theories for predicting the sign of a specific link, this link should participate in some triads, which is far from being the case in many real signed networks. Thus, applying social balance and status theory – in practice – does not result in high prediction accuracy in many real networks, which are sparse and with many of the links not appearing in any triad. We will show that our proposed method does not have this problem and outperforms these methods in terms of prediction accuracy.

2.4 Sign prediction based on machine learning methods

Most of the research carried out for sign prediction problem are based on machine learning framework [Chiang, Natarajan, Tewari and Dhillon 2011; DuBois, Golbeck and Srinivasan 2011; Leskovec, Huttenlocher and Kleinberg 2010; Shahriari, Askari, Gharibshah and Jalili 2012]. In methods based on a machine learning framework, first, a set of features are constructed for the links (or for the head and tail nodes). Then, a classifier is used to perform the prediction (or classification) task. Recently, Leskovec et al (2010) proposed a set of features for the trustor and trustee to use in sign prediction through logistic regression as a machine learning framework [Leskovec, Huttenlocher and Kleinberg 2010]. These features are listed in Table 1. In the same work, they further extended the feature list by considering some features based on status and balance theories. However, their results showed that this extension of feature list could not significantly improve the performance of the classification, although making the computations more complex [Leskovec, Huttenlocher and Kleinberg 2010].

Predictions obtained through machine learning are usually more precise than those obtained through social balance and status theories. However, they are often computationally costly, which limits their application in large-scale datasets Furthermore, when the network changes (which is often the case for many real social networks), the learning process should accordingly be updated accordingly, which means the computations should be repeated. The problem of "cold start" is another common problem of most techniques based on machine learning framework, which means the algorithms do not have satisfactory performance unless a fair amount of data (e.g., known sign of the links) are already available.

Table 1: features description for machine learning based approach

Description	Feature
# of positive links produced by trustor	d_{out}^{+} (trustor)
# of negative links produced by trustor	d_{out}^{-} (trustor)
# positive incoming links to trustee	d_{in}^{+} (trustee)
# positive incoming links to trustee	d_{in}^{-} (trustee)
Total # link produced by trustor	d_{out}^{+} (trustor) + d_{out}^{-} (trustor)
Total # link incoming to trustee	d_{in}^{+} (trustee) + d_{in}^{-} (trustee)
# common neighbor of trustee and trustor	CN (trustee,trustor)

3. COLLBORATIVE FILTERING FOR SIGN PREDICTION

As mentioned, methods based on machine learning frameworks cannot be efficiently applied for large networks and they always suffer from scalability problem. In this work, we proposed a method based on collaborative filtering, that is more computationally efficient that machine learning methods, while resulting in better prediction performance than them. Collaborative filtering algorithms are well-known in recommender systems, where proper items are recommended to users such that the users are likely to use (or positively rate) them in the future. In order to use collaborative filtering methodology in sign prediction, we suppose that each link (u,v) is a vote from (user) u towards (item) v, where the votes can be either +1 or –1. With this in mind that collaborative filtering methods largely depend on reliable similarities between users in order to result in precise prediction, the main problem of applying collaborative filtering for the sign prediction problem is often high level of sparsity in signed networks. This makes it difficult, if not impossible, to extract similarity between the users' voting patterns in such networks. Therefore, direct application of collaborative filtering to sign prediction problem does not result in satisfactory performance.

Here we give a solution to overcome this problem through, first, clustering the signed network, and then, applying collaborative filtering to the clusters. In this work, we used collaborative filtering in a different way than it is often used in recommender systems. In our method, in order to predict the sign of the link from user a to user b, we take the weighted aggregation of the votes pointed by all users to user b, where the weights are based on the similarity between the voter user and user b. Whereas, in standard collaborative filtering used in recommender systems, the vote from user a to user b is based on the votes pointed to user b by those that are similar to user a. The network clustering is carried out such that it maximizes the number of positive edges inside clusters and those with negative sign between clusters. As the network is clustered and collaborative filtering algorithm is applied, an output is obtained, which is a value in the range [–1,1]. We then use a thresholding approach to decide the sign [Zolfaghar and Aghaie 2010].

3.1 Collaborative filtering

Although the problem of sign prediction is similar to predicting users' voting, they are different in some aspects. The votes (signs) in the problem of sign prediction are limited to +1 and –1, whereas they can have much wider range in recommendation systems. Also, the graph structure in recommender systems has a bipartite type, where the links are from one group of nodes (users) to another group (items). But, in the sign prediction problem, each node can have a role as user and item, simultaneously.

In a recommender system, collaborative filtering algorithms try to extract utility of arbitrary item for target user based on previously recorded votes in the systems. As the items are valued for each user, a number of them with the highest value are recommended to the user. collaborative filtering algorithms can be performed, in general, in two forms: item-based and user-based [Herlocker et al. 2004; Konstan et al. 1997]. In user-based collaborative filtering, prediction is performed based on the recorded votes from other users on the target item, whereas item-based collaborative filtering estimates the utility based on votings from target user on other items. In this work, we used user-based collaborative filtering in the model. In the following, we give brief description of the user-based collaborative filtering algorithm. The main idea for user-based collaborative filtering is that the users with similar preferences and tastes in the past times are likely to have similar preferences in the future. User-based collaborative filtering recommends items to a particular user according to the preferences of its similar users. To this end, proper statistical methods are used to find a set of users that have a history of agreeing with the target user by rating or selecting similar set of items. One of the most frequently used techniques for measuring similarities between users is cosine-based similarity that is defined as [Sarwar, Karypis, Konstan and Riedl 2001]

$$Sim_{i,j} = \frac{\sum_{s \in S_{x,y}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{x,y}} r_{x,s}^{2}} \sqrt{\sum_{s \in S_{x,y}} r_{y,s}^{2}}}, \tag{4}$$

where $S_{x,y}$ represents the list of items, rated by both users u and v and $r_{x,s}$ indicates the vote of user x on item s. $Sim_{i,j}$ is the cosine similarity between users i and j.

In order to apply user-based collaborative filtering for the sign prediction problem, the similarity between the nodes should be first extracted. However, due to the high levels of sparsity – as defined in the following formula – of many real signed networks, this might not result in precise prediction. Let us define the sparsity of a network of users and items (in signed social networks all nodes are users and items) as follows

$$\text{Sparsity=1-}\frac{\text{Number of links in the network}}{\text{Possible links between users}}, \tag{5}$$

If a network is sparse, the similarity of preferences between the nodes will not be precise. This can be explained as follows. According to cosine-based similarity, as expressed in equation (4), in order to measure the similarity between any pair of nodes, first, the nodes received links from these two nodes are extracted, and then, the cosine similarity is estimated based on them. However, due to sparsity of real signed networks, there are not many such nodes receiving links from two adjacent nodes, and thus, the estimated similarity is not reliable. The reason for higher degree of sparsity in signed networks than bipartite networks of recommender systems is due to the fact that in signed networks all nodes are considered as users and items. This leads to having large number of nodes which may potentially receive and give votes. In order to compare the sparsity level of networks of recommender systems and those used for signed prediction, let us consider two sample networks. The networks have N nodes (users) and P links. Furthermore, the network of recommender system has M items where $M << N$. It can be simply seen that the sparsity value of the signed network, ($1 - P/N^2$), is much higher than that of the one used in the recommender system, ($1 - P/NM$).

We try to tackle the problem caused by sparsity through clustering the network. We try to have nodes with similar preferences of voting in the same clusters. We then apply user-based collaborative filtering to this clustered network by computing the similarities between the clusters. It is likely that there are enough links between the clusters, which makes it possible to have enough neighboring clusters for any cluster. Therefore, the inter-cluster similarity scores can be computed reliably in clustered networks.

3.2 Clustering signed networks

As mentioned in the previous section, a structurally balanced signed network can be clustered into clusters such that the inner-cluster links have positive sign, while all the negative links are inter the cluster. However, real networks do not often have complete structural balance and they are balanced to some extent. This makes it impossible to find such perfect clusters for real sign networks. Such networks can be divided into clusters such that the positive links become within the clusters and links with negative sign are placed inter the cluster, as much as possible. This means that in such clustering we will have some positive inter-cluster and negative inner-cluster links. An approach, called generalized block-modeling, has been proposed for clustering signed networks [Doreian and Mrvar 1996]. This algorithm, first, randomly assigns the nodes to a number of clusters, and then, tries to optimize an objective function by reallocating the nodes to predefined clusters. Criterion function implies inconsistent edges with an ideal k balanced functions, where k indicates the number of clusters that is an input for clustering algorithm. The algorithm uses an objective function to optimize, which is as follows

$$E_c = \alpha N + (1-\alpha)P , \quad (6)$$

where P is the number of positive edges between clusters, N is the total number of negative edges inside clusters and E_c is the error of the algorithm. α is a control parameter in the range [0,1], where $\alpha > 0.5$ means inconstancies of the negative links are more important, whereas $\alpha < 0.5$ gives more weight for inter-cluster positive links. Ideally, N and P should be small numbers to have a good clustering algorithm; the optimal algorithm is the one that minimizes E_c. Having the above objective function, the clustering can be considered as determining the clusters C' such that

$$E_{c'} = \min \; E_c \; ; \; c \in \Phi , \quad (7)$$

where Φ is the possible cluster sets for a given signed network. Since Φ has a large space, an iterative local optimization procedure has been introduced to do the optimization task [Doreian and Mrvar 1996]. The following pseudo-code summarizes the algorithm:

- Initialize nodes with random clusters
- While E(new clustering) > E(previous cluster); E indicates error, as expressed by equation (6),
    - For each node i (i = 1, ..., N; there are N nodes in the network):
        - Extract local optimal cluster
        - Change nodes' cluster to their local optimum found above and generate new clustering

Indeed, each node selects a local optimal cluster, by minimizing E_c according to the clusters to which its neighboring nodes belong. We used the algorithm proposed in [Doreian, Batagelj and Ferligoj 2005; Doreian and Mrvar 1996] to do the local optimization task. This greedy algorithm computes the local clustering error for the nodes. The local clustering error for node i ($LE_{c,i}$) is computed as

$$LE_{c,i} = \alpha N_i + (1-\alpha) P_i \,, \tag{8}$$

where N_i is the number of negative links between node i and those in its cluster and P_i is the number of positive links between node i and those located in other clusters. The algorithm, first, randomly assigns the nodes to one of the clusters (having fixed the number of clusters a priori). Then, in each iteration, the nodes, one-by-one, are located in the cluster that minimizes its local clustering error as expressed by equation (8). After each iteration, the clustering error, as expressed by equation (6), is computed and if it is more than that of the previous iteration, the algorithm stops; otherwise, a new iteration is performed.

3.3 Datasets

We applied the sign prediction algorithms on three real signed networks: Epinions, Slashdot, and WikiElection. These datasets have been frequently used as benchmarks in sign prediction [Chiang, Natarajan, Tewari and Dhillon 2011; DuBois, Golbeck and Srinivasan 2011; Leskovec, Huttenlocher and Kleinberg 2010; Shahriari, Askari, Gharibshah and Jalili 2012].

Table 2: Datasets statistics

	Node	Edges	+ Edges	– Edges
Epinions	119217	841200	85.0%	15.0%
Slashdot	82144	549202	77.4%	22.6%
Wikipedia	7118	103747	78.7%	21.2%

Epinions: Epinions is an online product review website where users give positive or negative votes on each other based on their review ratings on different topics presented in this website. Indeed, this is a directed signed network with both positive and negative relations among nodes. Epinions dataset contains 131828 nodes and 841372 edges.

Slashdot: Slashdot dataset has a similar structure with Epinions dataset. Slashdot is a technology news website where each user is allowed to give foe or friend values to other users. These labels on relation between users are considered as positive and negative edges. The network obtained from this website, comprises 82144 nodes and 549202 edges.

WikiElection: Wikipedia is a well-known encyclopedia which is managed by some promoting user. These users are elected as administers by the vote of other users, where the users' vote have positive or negative values.

The statistics of the datasets including the number of nodes and positive/negative links is summarized in table 2.

3.4 Extracting conditional similarities between clusters

As the network is divided into clusters, similarities between these clusters are obtained. We used cosine similarity (4) to obtain these similarities. In order to obtain reliable similarities between the clusters, we used a technique proposed in [Truong Khanh et al. 2006], which calculates similarity between two clusters based on their votes on the third cluster. In classic collaborative filtering algorithms, the similarities are calculated on the whole set of items. However, users with similar taste on a category of items (e.g., those items that are in the same cluster) may have completely different preferences on the items of other categories. In order to explain our strategy for calculating the similarity values, let us consider a sample network with four clusters A, B, C and D (Fig. 1). In such a network, nodes of clusters A and B might have similar votes (or links) with those in cluster C, while having dissimilar votes with the nodes in cluster D. We denote this measure as conditional similarity metric. Conditional similarity between two clusters A and B with respect to (or conditioned to) a third cluster C ($Sim_{A,B|C}$) is calculated as

$$Sim_{A,B|C} = \frac{\sum_{s \in S_{A,B|C}} m_{A,s} m_{B,s}}{\sqrt{\sum_{s \in S_{A,B|C}} m_{A,s}^2} \sqrt{\sum_{s \in S_{A,B|C}} m_{B,s}^2}}, \quad (9)$$

where $S_{A,B|C}$ indicates the list of nodes in cluster C receiving links from the nodes located in clusters A and B and $m_{A,s}$ represents average of the signs of the links from nodes located in cluster A to node s which can be calculated as

$$m_{A,v} = \frac{\sum_{s \in S_{A,v}} r_{s,v}}{|S_{A,v}|}, \quad (10)$$

where $S_{A,v}$ represents the list of nodes of cluster A linking to node v and $r_{s,v}$ is the sign of the edge from node s to v. $|S_{A,v}|$ indicates the number of ratings from nodes of cluster A to node v.

As discussed, the clustered networks will have majority of links with positive sign inside the cluster and those with negative sign inter the clusters. Considering two clusters A and B, the links from nodes of A to those in B will have mostly negative sign, while those inside B will be positive. Therefore, it is expected that similarity of clusters A and B conditioned to B to be negative. Similarity, the similarity between clusters A and B conditioned to any other third cluster gets a positive value.

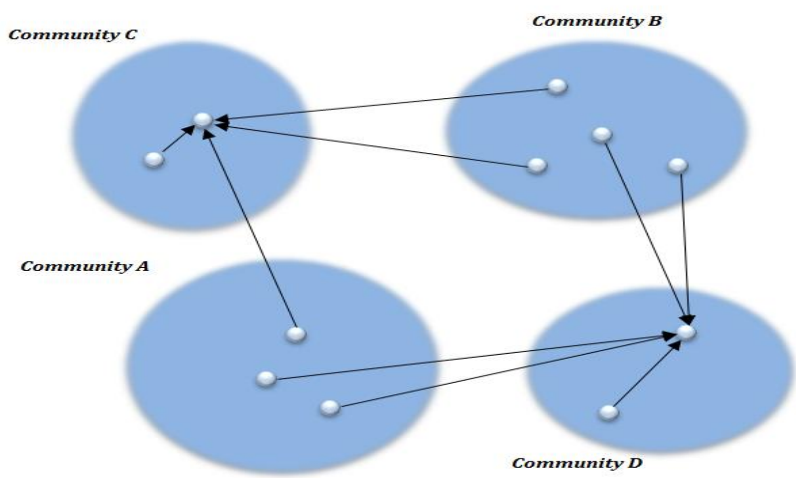

Figure 1: A sample network with four clusters A, B, C and D.

We applied the above procedure on one of the datasets – Epinions dataset. First, the network was divided into four clusters using the clustering algorithm resulting in clusters 1, 2, 3, and 4. Then, the inter-community similarity values, conditioned to cluster 4, were obtained. Table 3 shows such similarity scores. As it is seen, the similarity values between cluster 4 and other clusters are negative, whereas those between clusters 1, 2 and 3 are all positive. These similarity matrices can be used not only in sign prediction, but also in studying the inter-group relationship in social networks such as determining similar or opposite preferences between different groups.

Table 3: Similarity values between different clusters conditioned to cluster 4. The dataset is Epinions dataset, which has been divided into 4 clusters.

	Community 1	Community 2	Community 3	Community 4
Community 1	1.0000	0.3245	0.4225	-0.2357
Community 2	0.3245	1.0000	0.3394	-0.2883
Community 3	0.4225	0.3394	1.0000	-0.3648
Community 4	-0.2357	-0.2883	-0.3648	1.0000

3.5 Sign prediction based on inter-cluster similarities

In this work, we performed user-based collaborative filtering; i.e., the similarities were calculated based on outgoing links from the nodes, which mean that the nodes were treated as users. Alternatively, one can also compute the similarities based on incoming links, i.e., treating the nodes as items, leading to item-based collaborative filtering. Therefore, in our approach, in order to predict the sign of the link (u,v) from node u to node v, weighted average of the sign of links pointing to node v is considered. The values that are used for weighting the link between nodes i and v (for predicting the sign of (u,v)) is the similarity between the clusters to which these two nodes belong. The predicted sign of the link from node u to node v ($PS_{u,v}$) is obtained as

$$PS_{u,v} = \frac{1}{\sum_{i=1}^{N_v} Sim_{U,I|V}} \sum_{i=1}^{N_v} Sim_{U,I|V} \times r(i,v)\,, \qquad (11)$$

where N_v represents the number of links pointing to node v and $Sim_{U,I|V}$ indicates the similarity between the cluster to which node u belong (U) and the cluster to which node i belong (I) with respect to the cluster to which node v belong (V). $r(i,v)$ indicates the sign of link from user node i to node v. The above formula can also be represented as

$$PS_{u,v} = \frac{1}{\sum_{I=1}^{C} Sim_{U.I|V}} \sum_{I=1}^{C} Sim_{U.I|V} \times m_{I,v} \times R_{I,v}\,, \qquad (12)$$

where C represents the number of clusters in the network, $m_{I,v}$ is the average sign of the links from nodes located in cluster I to node v and $R_{I,v}$ is the total number of such links, which, to an extent, shows how reliably nodes of cluster I vote for node v. Let us suppose 100 nodes of cluster A have rated node v (with an average sign of a), while only 3 nodes in cluster B have links to node v (with an average sign of b). Based on equation (12), the votes originated from cluster A (a) will be more reliable (and thus influential) than those originated from cluster B (b) in sign prediction.

The above equations outputs a real number in the range [–1,1]. We then fix a threshold to decide the sign of the value. A simple threshold could be 0 for which higher values results in positive sign and lower values to negative sign. However, we may select it more consciously to have the highest accuracy on train data.

3.6 Computational complexity of collaborative filtering based predictor

In this section we investigate the computational complexity of the proposed predictor based on collaborative filtering strategy. The algorithm is divided into two parts: clustering the sign network and computing the similarity scores between the clusters. We discuss the computation complexity of these sections separately.

Computational complexity for clustering part is $O(INCd)$, where I is the number of iterations, N number of nodes, C number of clusters, and d average number of neighbors for each node, which is indeed the average degree of the network. Since, we have $E = dN$ for directed networks where E is the number of edges, the complexity reduces to $O(ICE)$. Often, the number of clusters is much less than the number of nodes in real-world large-scale networks, and we have $E \sim N$ for sparse networks. Thus, the computational complexity of the clustering section of the algorithm is almost $O(IN)$ for sparse networks. In practice, the algorithm converges in a finite number of iterations with $I << N$, and thus the complexity is much less than $O(N^2)$.

Second part of the algorithm – computing the inter-cluster similarity scores – has a computational complexity of $O(C^3\bar{c})$, where $\bar{c}$ is the number of nodes in each cluster that can be obtained by dividing N with C. Thus, the complexity is $O(C^2N)$. As we described earlier, we used cosine measure to extract similarity between two clusters conditioned to a third one. The cosine measure is applied on $\bar{c}$ dimensional vector of two clusters, where the vectors correspond to votes pointing from these clusters to the nodes of a third cluster. Obviously, the algorithm examines cosine similarity for C^3 times. Clustering the network not only overcomes the problem of sparsity of the network but also significantly reduces the computational complexity as compared to standard collaborative filtering. The computational complexity of standard collaborative filtering – $O(N^3)$ – is much higher than that of our proposed collaborative filtering that is $O(C^2N)$, since $C << N$.

The integrated computational complexity of the algorithm composed of these two parts is $O(C^2N) + O(IN)$ for sparse networks. I and C are very small in comparison to N, and in practice, the computational complexity of the algorithm is almost $O(N)$. For instance, in our datasets, at most 10 clusters are enough to make a good clustering and the clustering algorithm converges in at most 10 iterations. The computational complexity of our algorithm is much less than most approaches based on machine learning framework. For example, the logistic regression that has been used in this work as a machine learning framework, has a computational complexity of $O(E^3)$ that is almost $O(N^3)$ for sparse networks.

4. RESULTS

4.1 Evaluation method

In order to evaluate the performance of the proposed model, we tested them on the datasets and used 10-fold cross validation algorithm. In each fold, 90% of the dataset is considered as training dataset (for which the similarity measures are computed) and 10% as test dataset. The datasets has considerably more positive links than those with negative sign, which might bias the results. One way to overcome this problem is to compute the accuracy of the prediction on balance dataset, i.e., datasets with equal number of positive and negative links. Previous works in this field often tried to solve this problem by randomly removing some of the positive links to make the positive and negative links equal and then generating training and test datasets from this balanced dataset [Guha, Kumar, Raghavan and Tomkins 2004; Leskovec, Huttenlocher and Kleinberg 2010; Shahriari, Askari, Gharibshah and Jalili 2012]. However, some information is lost in this approach. As mentioned, many real signed social networks including the ones considered in this work have much more positive edges than those with negative sign (positive edges are almost five times more than negative ones). Balancing these networks will cause to randomly remove more than half of the edges (with positive sign), which will completely modify the structure of the networks. An alternative way to avoid loss of useful data is to train the free parameters based on the whole network, and then test the model on balanced datasets. Since the positive edges are randomly removed in order to balance the networks, this process should be repeated for a number of time and the results averaged over these runs. In this work, we introduce another method and show that it results in the same expected value but with better reliability as compared the balancing method. Instead of balancing the test datasets for a number of times and making an average, we used the mean true positive (TP) rates of the two classes 1 and –1.

Let us consider two classes; 1 for the set of positive links and –1 for the set of those with negative sign. Let us also consider the TP rate of the classes –1 and 1 as μ^- and μ^+, respectively. The variances are σ^- and σ^+. Let us suppose the original dataset consist of m positive edges and n negative edges where $n < m$ and the balanced test set contains n negative and n positive edges. First, we prove that the methods result in the same expected value. In the following statements, notations "+" stands for positive connections and "–" for negative links. In order to have reliable statistics, balanced networks should be constructed a number of times and the obtained accuracy scores averaged over these runs. Expectation value for accuracy on balanced test set ($E(A_{balanced})$) can be calculated as

$$E\left(A_{balanced}\right)=\frac{\left(E\left(\overline{TP^+}\right)+E\left(\overline{TP^-}\right)\right)}{2}, \tag{13}$$

where ($\overline{TP}^+$) indicates the TP rate on positive edges in test set and TP^+ represents the TP rate on individual positive edges. The above equation can be extended as follows

$$E\left(A_{balanced}\right)=\frac{1}{2}\left(\frac{\sum_1^n E(TP^+)}{n}+\frac{\sum_1^n E(TP^-)}{n}\right)=\frac{1}{2n}\left(\mu^+ n+\mu^- n\right)=\frac{1}{2}\left(\mu^+ +\mu^-\right). \tag{14}$$

Now consider our method where we compute the TP rates for the original data. Expectation value of accuracy ($E(A)$) for our method is

$$\begin{aligned}E\left(A\right)&=E\left(\frac{\left(\overline{TP^+}+\overline{TP^-}\right)}{2}\right)=\frac{\left(E\left(\overline{TP^+}\right)+E\left(\overline{TP^-}\right)\right)}{2}\\&=\frac{1}{2}\left(\frac{\sum_1^m E\left(TP^+\right)}{m}+\frac{\sum_1^n E\left(TP^-\right)}{n}\right)=\frac{1}{2}\left(\frac{\mu^+ m}{m}+\frac{\mu^- n}{n}\right)=\frac{1}{2}\left(\mu^+ +\mu^-\right)\end{aligned}. \tag{15}$$

Thus, these methods result in the same mean accuracy. We now show our method is more reliable, i.e., it results in lower variance in the accuracy value. Variance of the accuracy in the balanced method ($Var(A_{balanced})$) can be calculated as

$$Var\left(A_{balanced}\right)=\frac{1}{4}\left(Var\left(\overline{TP^{+}}\right)+Var\left(\overline{TP^{-}}\right)\right)=\frac{1}{4}\left(\frac{\sum_{1}^{n}Var\left(TP^{+}\right)}{n^{2}}+\frac{\sum_{1}^{n}Var\left(TP^{-}\right)}{n^{2}}\right)$$
$$=\frac{1}{4n^{2}}\left(\sigma^{+}n+\sigma^{-}n\right)=\frac{1}{4n}\left(\sigma^{+}+\sigma^{-}\right), \quad (16)$$

and that of our method is calculated as

$$Var\left(A\right)=Var\left(\frac{\left(TP^{+}+TP^{-}\right)}{2}\right)=\frac{1}{4}\left(Var\left(\overline{TP^{+}}\right)+Var\left(\overline{TP^{-}}\right)\right)$$
$$=\frac{1}{4}\left(\frac{\sum_{1}^{n}Var\left(TP^{+}\right)}{m^{2}}+\frac{\sum_{1}^{n}Var\left(TP^{-}\right)}{n^{2}}\right)=\frac{1}{4}\left(\frac{\sigma^{+}m}{m^{2}}+\frac{\sigma^{-}n}{n^{2}}\right)=\frac{1}{4}\left(\frac{\sigma^{+}}{m}+\frac{\sigma^{-}}{n}\right). \quad (17)$$

Since $m > n$, $Var(A) < Var(A_{balanced})$. Therefore, our method leads to more reliable results than the previous one.

4.2 Experimental results and discussion

Two parameters which may largely influence the performance of the collaborative filtering based sign prediction are the number of clusters and α as expressed by equation (6). These parameters should be tuned in the clustering phase of the algorithm and before computing the similarity measures. Not a single value of these parameters is optimal in all networks, and for each of the datasets, a specific value is obtained as optimal one. In order to assess the performance of the methods, we computed the accuracy of the prediction resulted by the methods. The prediction accuracy is calculated as:

$$Accuracy = \frac{\text{Number of edges for which the sign is predicted correctly}}{\text{Total number of edges in the network}}\times 100. \quad (18)$$

As discussed, due to high sparsity levels in real signed social networks, it is likely that some of the edges do not participate in any triad, for which one cannot use social balance and status theories to make predictions on the signs. In order to make the performance of the collaborative filtering based method comparable to those based on social balance and status theories, we used the following metric to assess the accuracy of predictors based on social balance and status theories:

$$Accuracy_{\text{social banal and status theories}} = A_T \times R + 0.5(1-R), \quad (19)$$

where A_T indicates the accuracy (as computed by equation (18)) of the algorithm on the edges that participate in at least one triad in the test set and R is the ratio of such edges to all edges of the test set. For the above relation, it is assumed that for the edges that do not participate in any triad, predictions based on social balance and status theories work as a random machine and select one of the labels 1 or –1 at random (such a random assignment process will result in an accuracy of 50%, in average). Indeed, for such cases, social status and balance theories cannot make any prediction, and we used (19) in order to make them comparable to other methods. Therefore, while the standard definition of accuracy, as expressed by equation (18) was used to assess the performance of the method based on machine learning framework (logistic regression in this work) and the proposed collaborative filtering based method, the modified measure of accuracy, as expressed by equation (19) was used to assess the performance of the methods based on social balance and status theories.

Figure 2 shows the accuracy, as expressed by equation (18), as a function of the number of clusters for Epinions, Slashdot and Wiki datasets. To run these experiments, we set α = 0.5, which means the positive inter-cluster links and negative inner-cluster links were treated the same in the objective function (6). It is seen that not a single optimal value is obtained for all networks. As the number of clusters increases, we need more computations to obtain similarity measures, and the computational complexity increases, and thus, the less clusters we have, the better. We chose the number of clusters as 7, 10 and 10 for Epinions, Slashdot and Wiki datasets, respectively.

We next investigated how the prediction accuracy depends on α, i.e., how much the results depend on penalizing the clustering for positive inter-cluster and negative inner-cluster links. Figure 3 shows the accuracy as a function of α for Epinions, Slashdot and Wiki datasets, respectively. We obtained $\alpha = 0.7$ as an optimal value for Epinions and Slashdot datasets and α = 0.9 for Wiki dataset. Since the optimal α is higher than 0.5, this means that penalizing negative inner-cluster links is more influential than inter-cluster links with positive sign.

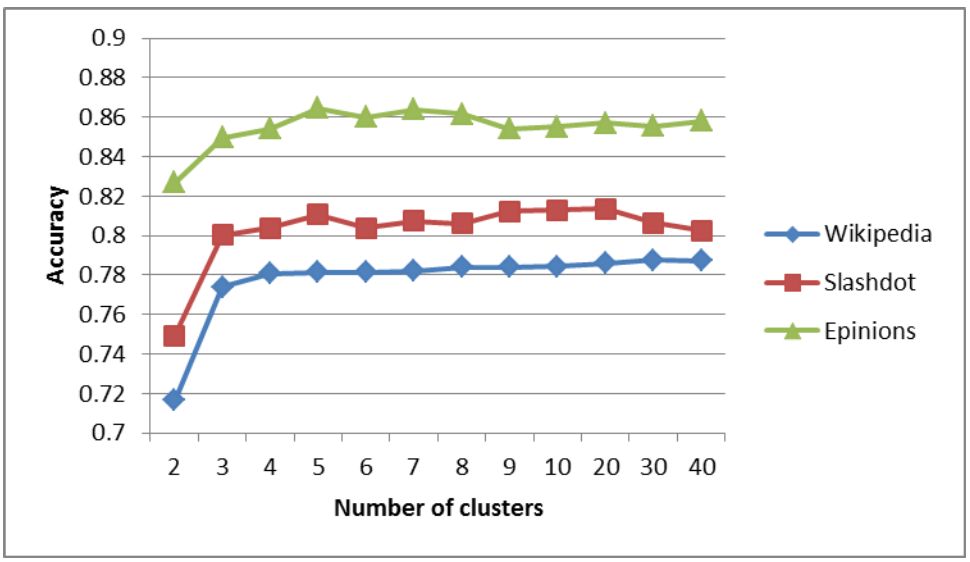

Figure 2: Accuracy of the predictor based on collaborative filtering, as a function of the number of clusters for Wikipedia, Slashdot and Epinions datasets.

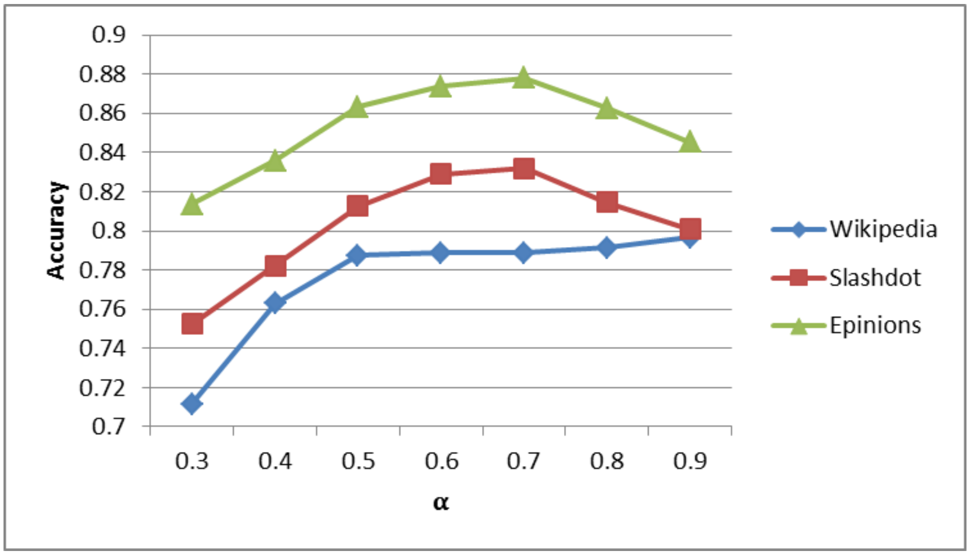

Figure 3: Accuracy of the predictor based on collaborative filtering, as a function of α (which is used in clustering the signed networks – as described in equations (6) and (8)) for Wikipedia, Slashdot and Epinions datasets.

We used the optimal values of the number of clusters and parameter α for our proposed collaborative filtering based sign predictor. The results were compared with those based on social balance and status theories, which are simple to compute without need for extensive calculations. We also compared the performance with a machine learning approach where a logistic regression, as a predictor, was used based on the features listed in table 1 [Leskovec, Huttenlocher and Kleinberg 2010]. This method is much more complex than those based on social theories. It is also more complex than our collaborative filtering based approach. Figure 4 shows the prediction accuracy of these four algorithms in the datasets. As it is seen, the proposed collaborative filtering based algorithm outperformed the other three in terms of prediction performance in all datasets. Collaborative filtering based predictor outperformed logistic regression by about 15% in Epinions dataset. Interestingly, in this dataset, simple predictions based on social balance theory resulted in a close accuracy to logistic regression; 75% for social balance theory and 76% for logistic regression. This means that in such a dataset, a simple and computationally-scalable prediction based on prime social theories can have a comparable performance with complex machine learning techniques.

In Slashdot dataset, collaborative filtering based predictor demonstrated the best performance with 83% accuracy, followed by the logic regression with 70%, and those based on social balance with 62% and status theory with 57%. The proposed predictor outperformed the logistic regression by 5% in Wikipedia dataset. Prediction based on social status theory resulted in better accuracy (71%) as compared to the one based on social balance theory (69%) for this dataset. The reason for better performance of status theory than balance theory in this dataset is its specific structure. In Wikipedia dataset, users with better position that could demonstrate their reliability to others, often receive more positive votes. This can be interpreted as their status, i.e., users with higher status receive more positive votes. Whereas in the two other datasets (Epinions and Slashdots), the votes are mainly based on tastes and preferences of the users, and thus, balance theory could predict the signs with higher accuracy than status theory.

A reason for higher accuracy of the proposed collaborative filtering based algorithm to logistic regression (as a machine learning framework) is that the collaborative filtering based algorithm considers global topological features by graph clustering and obtaining inter-cluster similarity scores. Whereas, the approach based on machine learning framework is based on local graph measures, which has limited information on the topological properties of the networks. One can also consider more complicated features for machine learning based techniques. However, this requires higher computational complexity resulting in scalability issue of the algorithm, which makes it difficult to apply to large-scale networks. Also, as the features become more complicated, online computational load of the system increases.

In order to further compare machine learning and collaborative filtering based approaches, let us consider a target edge between nodes a and b, in which a large portion of outgoing links from node a and incoming connections to node b are positive. In such a case, a method based on machine learning has a tendency to predict the sign of the target edge as positive. But, this prediction can make error when most of the nodes voted on b, have totally different tastes than node a, and thus, the real value for target edge may have negative sign. However, the collaborative filtering based algorithm considers the similarities between the node a and those that have already voted to b. If most of the nodes that have voted to b have negative similarity with a, the algorithm predicts a negative sign for the link from node a to node b. Indeed, the proposed collaborative filtering based predictor not only has lower computational complexity than the method based on machine learning but also it is capable of considering more useful information and resulting in higher prediction accuracy.

It is worth mentioning that since we do not balance the training dataset, the number of training samples for two classes 1 and –1 is not equal, resulting a bias towards a class with higher number of samples in methods based on a machine learning framework. Therefore, our results are not directly comparable to those obtained in [Leskovec, Huttenlocher and Kleinberg 2010], since they balance both the training and test datasets to avoid this bias. Balancing both training and test datasets, as reported in [Leskovec, Huttenlocher and Kleinberg 2010] achieve an accuracy of 86%, 80% and 80% for Epinions, Slashdot and Wikipedia datasets, respectively (for the case with embeddedness of 0). Except in Wikipedia dataset for which our proposed method resulted in an accuracy of 80% (similar to the one obtained by logistic regression on balanced networks), the proposed method outperformed the logistic regression (applied to balanced networks) in Slashdot and Epinions datasets.

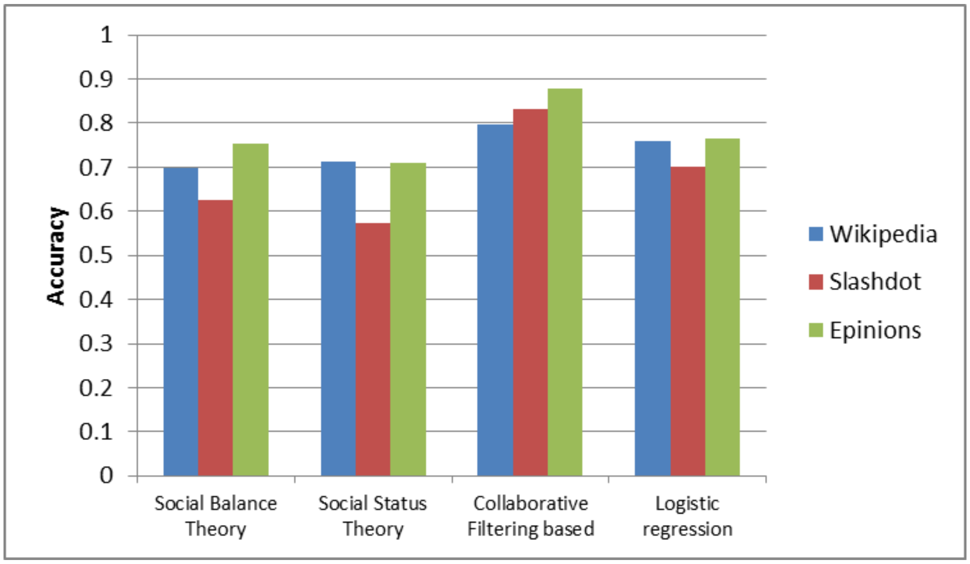

Figure 4: Accuracy of the predictors based on social balance theory, social status theory, logistic regression (a machine learning based approach), and collaborative filtering (our proposed method) for Wikipedia, Slashdot and Epinions datasets.

5. CONCLUSION AND FUTURE WORKS

In this manuscript, we introduced an algorithm for sign prediction in social networks with positive and negative signs on the links. The algorithm has two steps: extracting the community structure of the network and applying collaborative filtering for sign prediction. In the first step, the signed network is divided into a number of clusters such that the number of inter-cluster links with positive sign and inner-cluster negative links are minimized. This clustering is due to social balance theory; a signed network is denoted as a structurally balanced network, if and only if, it can be divided into clusters for which all the inner-cluster links are positive and all inter-cluster connections have negative sign. We then used a collaborative filtering algorithm for sign prediction. One of the key ingredients of collaborative filtering methods is the similarity values between the network elements. In this work, we used similarity measures between the clusters (based on the links between them). We applied the algorithm on a number of real singed networks and compared its performance with that of a number of algorithms including those based on social balance and status theories and the one based on machine learning framework (logistic regression in this work). While the formers are simple to compute (and hence can be applied to very large datasets), prediction based on machine learning based technique needs extensive computations and may not be applicable to large networks. Our proposed algorithm is much simpler than those based on machine learning frameworks (e.g., logistic regression), but more complex than those based on social theories. Our experiments showed that the proposed predictor outperformed both simple (i.e., those based on social balance and status theories) and complex (i.e., logistic regression) algorithms. Our strategy in this work is indeed to perform a preprocessing on the network (i.e., extracting the community structure based on minimizing inner-community negative links and inter-community positive connections) ahead of sign prediction problem. This steps makes it possible to use macro-scale properties (i.e., community-wise similarity scores) in predicting a micro-scale property that is the sign of a connecting link.

A limitation to our work is unavailability of diverse datasets on signed networks that makes it necessary to apply it in more diverse datasets, which yet to be introduced within the community of social networks. Another future direction to this research would be to apply the same strategy to link prediction problem. The major challenge in the link prediction problem is to introduce useful features for the nodes in order to predict the forthcoming edges. Having the community structure of the network, one can associate higher chance for the intra-community links to appear as compared to inter-community connections. A particular application of the proposed collaborative filtering method could in recommender systems for which identifying the community structure within items and users can help to build better recommendation systems. The method introduced in this work can be used in other applications as well. For example, similar methods could be used for analyzing inter-group relations in social networks. Such analysis can be used to construct trust-aware recommender systems.

REFERENCES

ANTAL, T., KRAPIVSKY, P.L. AND REDNER, S. 2005. Dynamics of social balance on networks. *PHYSICAL REVIEW E 72*, 036121.

BARABASI, A.-L. AND ALBERT, R. 1999. Emergence of scaling in random networks. *Science 286*, 5009-5012.

BO, Y., CHEUNG, W.K. AND JIMING, L. 2007. Community Mining from Signed Social Networks. *Knowledge and Data Engineering, IEEE Transactions on 19*, 1333-1348.

BOCCALETTI, S., LATORA, V., MORENO, Y., CHAVEZ, M. AND HWANG, D.U. 2006. Complex networks: structure and dynamics. *Physics Reports 424*, 175-308.

BOGDANOV, P., LARUSSO, N.D. AND SINGH, A. 2010. Towards Community Discovery in Signed Collaborative Interaction Networks. In *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, 288-295.

BRZOZOWSKI, M.J., HOGG, T. AND SZABO, G. 2008. Friends and foes: ideological social networking. In *Proceedings of the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Florence, Italy2008 ACM, 1357183, 817-820.

BURKE, M. AND KRAUT, R. Mopping up: modeling wikipedia promotion decisions. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, 27-36.

CARTWRIGHT, D. AND HARARY, F. 1956. Structure balance: A generalization of Heider's theory. *Psychological Review 63*, 277-293.

CHIANG, K.-Y., NATARAJAN, N., TEWARI, A. AND DHILLON, I.S. 2011. Exploiting longer cycles for link prediction in signed networks. In *Proceedings of the ACM Conference on Information and Knowledge Management*, Glasgow, Scotland, UK.2011.

DOREIAN, P., BATAGELJ, V. AND FERLIGOJ, A. 2005. *Generalized Blockmodeling*. Cambridge University Press, New York.

DOREIAN, P. AND MRVAR, A. 1996. A partitioning approach to structural balance. *Social Networks 18*, 149-168.

DUBOIS, T., GOLBECK, J. AND SRINIVASAN, A. 2011. Predicting Trust and Distrust in Social Networks. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, 418-424.

GIRVAN, M. AND NEWMAN, M.E.J. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Science of the United States of America 99*, 7821-7826.

GUHA, R., KUMAR, R., RAGHAVAN, P. AND TOMKINS, A. 2004. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, 403 - 412

HEIDER, F. 1946. Attitudes and Cognitive Organization. *The Journal of Psychology 21*, 107-112.

HERLOCKER, J.L., KONSTAN, J.A., TERVEEN, L.G. AND RIEDL, J.T. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems 22*, 5-53.

HOLME, P. AND SARAMÄKI, J. 2012. Temporal networks. *Physics Reports 519*, 97-125.

HUANG, Z., CHEN, H. AND ZENG, D. 2004. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems 22*, 116 - 142

KONSTAN, J., MILLER, B., MALTZ, D., HERLOCKER, J., GORDON, L. AND RIEDL, J. 1997. GroupLens: applying collaborative filtering in usenet news. *Communications of the ACM 40*, 77-87.

KOSSINETS, G. AND WATTS, D.J. 2006. Empirical analysis of evolving social networks. *Science 311*, 88-90.

KUMAR, R., NOVAK, J. AND TOMKINS, A. 2006. Structure and evolution of online social networks. In *ACM International Conference on Knowledge Discovery and Data Mining*, 611-617.

KUNEGIS, J., LOMMATZSCH, A. AND BAUCKHAGE, C. 2009. The slashdot zoo: mining a social network with negative edges. In *Proceedings of the Proceedings of the 18th international conference on World wide web*, Madrid, Spain2009 ACM, 1526809, 741-750.

LESKOVEC, J., HUTTENLOCHER, D. AND KLEINBERG, J. 2010. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*, 641-650

LESKOVEC, J., HUTTENLOCHER, D. AND KLEINBERG, J. 2010. Signed networks in social media. In *28th international conference on Human factors in computing systems* acm, New York, NY, USA, 1361-1370.

LESKOVEC, J., KLEINBERG, J. AND FALOUTSOS, C. 2007. Graph evolution: densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data 1*, 1-40.

LIBEN-NOWELLY, D. AND KLEINBERG, J. 2003. The link prediction problem for social networks. In *Twelfth Annual ACM International Conference on Information and Knowledge Management*, 556-559.

LÜ, L. AND ZHOU, T. 2011. Link prediction in complex networks: A survey. *Physica A 390*, 1150-1170.

MANOS PAPAGELIS, D.P. 2005. Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents. *Engineering Applications of Artificial Intelligence 18*, 781–789.

MASSA, P. AND AVESANI, P. 2005. Controversial users demand local trust metrics: an experimental study on Epinions.com community. In *Proceedings of the Proceedings of the 20th national conference on Artificial intelligence - Volume 1*, Pittsburgh, Pennsylvania2005 AAAI Press, 1619354, 121-126.

NEWMAN, M.E.J. AND PARK, J. 2003. Why social networks are different from other types of networks. *PHYSICAL REVIEW E 68*, 036122.
O'DONOVAN, J. AND SMYTH, B. 2005. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interface*, 167-174.
PERC, M. 2009. Evolution of cooperation on scale-free networks subject to error and attack. *New Journal of Physics 11*, 033027.
PERC, M. AND SZOLNOKI, A. 2010. Coevolutionary games-A mini review. *BioSystems 99*, 109-125.
SARWAR, B., KARYPIS, G., KONSTAN, J. AND RIEDL, J. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*2001 ACM.
SHAHRIARI, M., ASKARI, O., GHARIBSHAH, J. AND JALILI, M. 2012. Predicting sign of edges in social networks based on users reputation and optimism. *submitted*.
STROGATZ, S.H. 2001. Exploring complex networks. *Nature 410*, 268-276.
TRUONG KHANH, Q., FUYUKI, I. AND SHINICHI, H. 2006. Improving Accuracy of Recommender System by Clustering Items Based on Stability of User Similarity. In *Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, 61-61.
WATTS, D.J. AND STROGATZ, S.H. 1998. Collective dynamics of ‘small-world’ networks. *Nature 393*, 440-442.
XUE, G.-R., YANG, Q., XI, W., ZENG, H.-J., YU, Y. AND CHEN, Z. 2005. Scalable collaborative filtering using cluster-based smoothing. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 114 - 121
YILDIRIM, H. AND KRISHNAMOORTHY, M.S. 2008. A random walk method for alleviating the sparsity problem in collaborative filtering. In *Proceedings of the 2008 ACM conference on Recommender systems*, 131-138
ZOLFAGHAR, K. AND AGHAIE, A. 2010. Mining trust and distrust relationships in social web applications. In *IEEE 6th International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania 73 - 80.