A User Interaction Based Community Detection Algorithm for Online Social Networks

Himel Dev [*]
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
himeldev@gmail.com

ABSTRACT

Existing community detection techniques either rely on content analysis or only consider the underlying structure of the social network graph, while identifying communities in online social networks (OSNs). As a result, these approaches fail to identify *active communities*, i.e., communities based on actual interactions rather than mere friendship. To alleviate the limitations of existing approaches, we propose a novel solution of community detection in OSNs.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*

Keywords

Community Detection, Social Networks

1. INTRODUCTION

The community detection involves grouping of similar users into clusters, where users in a group are strongly bonded with each other than the other members in the network. We propose a novel community detection technique that considers the structure of the social network and interactions among the users while detecting the communities. The key idea of our approach comes from the following observations: (i) the degree of interaction between each pair of users can widely vary, which we term as *the strength of ties*, and (ii) for each pair of users, the degree of interactions with common neighbors (e.g., mutual friends in Facebook), which we term the *group behavior*, play an important role to determine their belongingness to the same community. Based on these two observations, we propose an efficient solution to detect communities in OSNs. The detailed experimental study shows that our proposed algorithm significantly outperforms state-of-the-art techniques for both real and synthetic datasets.

2. RELATED WORK

SCAN [3] and Truss [1] use the neighborhood concept of common neighbors, which has similarity with our group interaction concept for un-weighted graphs. However, these are pure link-based methods based on topological structures. They focus only on the information regarding the linkage behavior (connection) for the purposes of community prediction and clustering. They do not utilize different attributes present in networks, and as a result, their performance degrades in networks with rich contents, e.g., OSNs.

3. METHODOLOGY

The proposed community detection algorithm has four phases. In the first phase, the algorithm quantifies the degree of interaction between every connected pair of users in the OSNs and based on these interactions, in the second phase, the algorithm quantifies the group behavior for every pair of users who are connected via common neighbors. In the third phase, the algorithm determines the probability of two users belonging to the same community using the impact of interaction between them and their group behavior. Finally, in the fourth phase, the algorithm applies hierarchical clustering to detect communities based on the computed probabilistic measure.

3.1 Quantifying the Interaction

Given a social graph $G(V, E)$ and user interaction data, the algorithm constructs an *Interaction Graph*, $G_I(V, E, W)$, where each weight $w_{uv} \in W$ quantifies the degree of interaction between two users u and v. We have considered the following three factors to quantify interaction between two users: interaction type, average number of interactions for a particular interaction type, and relative interaction.

Interaction Type: Now-a-days OSNs involve interactions of different types. For example, Facebook users can interact via personal messages, wall posts, photo tags, page likes etc. To quantify the degree of interaction between two users, it is necessary to consider all interaction types. In addition, we observe that, some of these interaction types indicate stronger bonding than the others. Thus, it is important to prioritize the interaction types in an order. Prioritizing the interaction types in terms of bonding is especially useful for applications such as friend recommendation, and influence analysis. We prioritize different interaction types using weights.

In addition, in an OSN, there could be both active and passive users. Sometimes there is no interaction involved in a link established by a passive user. This can also happen

[*] Under supervision of Dr. Mohammed Eunus Ali and Dr. Tanzima Hashem from CSE, BUET

SIGMOD/PODS'14, June 22–27, 2014, Snowbird, UT, USA.
ACM 978-1-4503-2376-5/14/06.
http://dx.doi.org/10.1145/2588555.2612664.

for newly joined users of an OSN. To determine the community in which a passive/new user belongs to, we consider the establishment of friendship between two users as a special type of interaction and to incorporate this special type of interaction, our communication detection algorithm provides a threshold value for each established friendship link.

Average Number of Interactions for a Particular Interaction Type: Another important factor to consider is the average number of interactions for a particular interaction type, which is not same for all interaction types. To address this issue, we normalize the number of interactions for each type using the average value corresponding to the type. Otherwise, interaction types with higher average values eliminate the effect of type prioritization.

Relative Interaction: We observe that the importance of an interaction can vary among users based on their activities. To incorporate this issue in quantifying the interaction between users for our community detection algorithm, we take relative interaction into account. To quantify the impact of relative interaction, we normalize each interaction in terms of total interaction of the involved users.

Quantification: Let $\{I^1, I^2, ..., I^t\}$ represent t interaction types in an OSN. Assume that weights $W^1, W^2, ..., W^t$ are associated with $I^1, I^2, ..., I^t$, respectively, where $W^i > W^j$, if I^i represents stronger bonding between users than that of I^j. We first quantify interaction between users based on $\{I^1, I^2, ..., I^t\}$, and then to incorporate the impact of links without interaction, we add an additional threshold value, ϵ, to the quantified interaction.

Let i^t_{uv} represent the number of interactions of type t between users u and v, and n is the number of users in the social graph $G(V, E)$. The average number of interactions for a particular type t, $\bar{I^t}$, is computed as $\sum_{u \in V} \sum_{v \in V \wedge u \neq v} i^t_{uv}/n$. The normalized number of interactions of type t between u and v, I^t_{uv}, is computed as $i^t_{uv}/\bar{I^t}$.

Considering prioritized interaction types, links without interaction, and the average number of interactions for each interaction type, the algorithm quantifies interaction between two users u and v as $\hat{w_{uv}}$ using the following equations:

$$\hat{w_{uv}} = I^1_{uv} \times W^1 + I^2_{uv} \times W^2 + ... + I^t_{uv} \times W^t + \epsilon \qquad (1)$$

Finally, to incorporate the impact of the relative importance of an interaction between users u and v, the algorithm considers the quantified interactions of u and v with their neighbors $N(u)$ and $N(v)$, respectively, using Equation 1 and modifies $\hat{w_{uv}}$ as w_{uv} using the following equation:

$$w_{uv} = \frac{1}{2}\left(\frac{\hat{w_{uv}}}{\sum_{x \in N(u)} \hat{w_{ux}}} + \frac{\hat{w_{uv}}}{\sum_{y \in N(v)} \hat{w_{yv}}}\right) \qquad (2)$$

3.2 Quantifying Group Behavior

Given an interaction graph, $G_I(V, E, W)$, the algorithm constructs a *Group Interaction Graph*, $G_{GI}(V, E', W')$, where there is an edge, $e'_{uv} \in E'$ between two users u and v, if they have at least a common neighbor. W' is a set of weights, where each weight quantifies the group behavior between two users with respect to common neighbors. Let $M_{uv} = \{m_1, m_2, ..., m_h\}$ represent h common neighbors (mutual friends) of u and v, where $m_1, m_2, ..., m_h, u, v \in V$. The interaction with a common neighbor $m_i \in M_{uv}$, $w^{m_i}_{uv}$ is quantified as $w^{m_i}_{uv} = \min(w_{um_i}, w_{vm_i})$.

The proposed algorithm computes the interaction of every pair of users who are connected via one or more common neighbor with respect to each of their common neighbors and then quantifies the group behavior for pair of users, $w_{M_{uv}} \in W'$ as follows:

$$w_{M_{uv}} = \sum_{i=1}^{h} w^{m_i}_{uv} \qquad (3)$$

3.3 Computing the Probabilities

Given an interaction graph, $G_I(V, E, W)$, and a group interaction graph, $G_{GI}(V, E', W')$, the algorithm constructs a *Probability Graph*, $G_p(V, E \cup E', P)$, where each weight $p_{uv} \in P$ represents a probability between two vertices u and v to belong in the same community.

Definition 3.1. ***(Probability of belonging to the same community)*** *Let w_{uv} and $w_{M_{uv}}$ represent the weight of interaction between u and v and their group behavior, respectively, and α be a parameter used to combine the impact of w_{uv} and $w_{M_{uv}}$, where $0 \leq \alpha \leq 1$. The probability of u and v to belong to the same community, p_{uv}, is defined as follows:*

$$p_{uv} = \alpha * w_{uv} + (1 - \alpha) * w_{M_{uv}} \qquad (4)$$

We use the parameter α to control the impact of w_{uv} and $w_{M_{uv}}$ on probability for users u and v. We experimentally find the appropriate value of α that identifies the communities with a high accuracy.

3.4 Hierarchical Clustering

The final phase of the proposed algorithm involves identifying communities by applying hierarchical clustering in the probability graph $G_p(V, E \bigcup E', P)$.

Distance Measure: In our approach, the probability $p_{uv} \in P$ of u and v to belong to the same community serves as the similarity measure, $(1 - p_{uv})$ serves as distance.

Linkage Criterion: To estimate the distance between two clusters C_i and C_j, we have used 'Unweighted Pair Group Method with Arithmetic Mean (UPGMA)'as the linkage criterion.

4. EXPERIMENTAL EVALUATION

We have observed that for a real Facebook user interaction dataset, our algorithm achieves the *highest* normalized mutual information (NMI) and pairwise F-measure (PWF) values, (0.79, 0.79), which are significantly higher than the NMI and PWF values achieved by other competitive methods such as Girvan-Newman (0.49, 0.38), Walktrap (0.78, 0.77), Leading Eigenvector (0.73, 0.75), Infomap (0.35, 0.3), Multilevel (0.63, 0.56) [2]. Moreover, our algorithm performs reasonably well for different benchmark datasets, for both weighted and un-weighted social graphs.

5. REFERENCES

[1] J. Cohen. Trusses: Cohesive subgraphs for social network analysis. 2008.

[2] H. Dev, M. E. Ali, and T. Hashem. User interaction based community detection in online social networks. In *Database Systems for Advanced Applications*, volume 8422, pages 296–310. 2014.

[3] X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger. Scan: a structural clustering algorithm for networks. In *ACM SIGKDD*, pages 824–833, 2007.