Community Detection based on Structural and Attribute Similarities

The Anh Dang, Emmanuel Viennet
L2TI - Institut Galilée - Université Paris-Nord
99, avenue Jean-Baptiste Clément - 93430 Villetaneuse - France
{*theanh.dang,emmanuel.viennet*}*@univ-paris13.fr*

***Abstract*—The study of social networks has gained much interest from the research community in recent years. One important challenge is to search for communities in social networks. A community is defined as a group of users such that they interact with each other more frequently than with those outside the group. Being able to identify the community structure can facilitate many tasks such as recommendation of friends, network analysis and visualization. In real-world networks, in addition to topological structure (i.e., links), content information is also available. Existing community detection methods are usually based on the structural features and do not take into account the attributes of nodes. In this paper, we propose two algorithms that use both structural and attribute information to extract communities. Our methods partition a graph with attributes into communities so that the nodes in the same community are densely connected as well as homogeneous. Experimental results demonstrate that our methods provide more meaningful communities than conventional methods that consider only relationship information.**

***Keywords*-social network; community detection; clustering;**

I. Introduction

Social networks of various kinds demonstrate a feature called community structure. Individuals in a network tend to form closely-knit groups. The groups are called communities or clusters in different context. Community detection is the task of detecting these cohesive groups in a social network [1] [2]. In many real-world networks, in addition to topological structure, content information is also available. Data is associated to the nodes and in the form of text, images, etc. For example in a social network, each user has information about age, profession, interests, etc. When content data is available, it might be relevant to extract groups of nodes that are not only connected in the social graph but also share similar attributes.

Many existing community detection techniques only focus on the topological structure of the graph. On the other hand, data clustering has been studied for a long time but most algorithms (e.g., k-means, EM) do not deal with relational data. The work of incorporating structural and attribute data has not been throughly studied yet in the context of large social graphs. This is the motivation of our work. Our key contributions are summarized next. In this paper, we study the relationship between semantic similarity of users and the topology of social networks (homophily concept). We propose two approaches to extract communities on several real-world datasets. Based on our evaluations, we conclude that our methods are able to discover more relevant communities.

II. Related Work

Detecting communities in a social network is still an open problem in social network analysis. In literature, many community detection methods have been proposed. According to [1], these approaches can be divided into four categories: node-centric, group-centric, network-centric and hierarchic-centric. Some popular methods are modularity maximization [3] [4], Givan-Newman algorithm [5], Louvain algorithm [6], clique percolation [7], link communities [8]. [2] and [9] provide a throughout review of the topic. However, these methods ignore the attributes of the nodes. Below are some studies that incorporate node attributes in the clustering process. Steinhaeuser et al. [10] proposed an edge weighting method NAS (Node Attribute Similarity) that takes into account node attributes. A community detection method is then proposed based on random walks. The complexity of the algorithm is $O(n^2 logn)$ (for random walks) or $O(n)$ (for scalable random walks) where n is the number of nodes. Zhou et al. [11] defined a unified distance measure to combine structural and attribute similarities. Attribute nodes and edges are added to the original graph to connect nodes which share attribute values. A neighborhood random walk model is used to measure the node closeness on the augmented graph. A clustering algorithm SA-Cluster is proposed, following the K-Medoids method. The time complexity of the algorithm is $O(n^3)$.

Coupling relationship and content information in social network for community discovery is an emerging research area because current methods do not focus on social graphs or they are not efficient for large-scale datasets.

III. Problem Statement

An attributed graph is denoted as $G = (V, E, X)$, where V is the set of nodes, E is set of edges, $X = X^1, ..., X^d$ is the set of d attributes associated with the nodes in V. Each vertex v_i is associated with an attribute vector $(x_i^1, ..., x_i^d)$. The goal of this work is to find communities in an attributed graph, that is to partition the graph into K disjoint groups (i.e., communities) $G_i = (V_i, E_i, X)$, where $V = \cup_{i=1}^{K} V_i$

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-176-2

and $V_i \cap V_j = \emptyset \; \forall i \neq j$. Nodes in the same communities are expected to be highly connected and have similar attributes.

Before clustering, a similarity measure must be determined. Our algorithms do not depend on the details of the measurement. Let $simA(i,j)$ be the similarity between a pair of nodes (i,j) in an attributed graph $G = (V, E, X)$. The measure should reflect the degree of closeness of the nodes in terms of their attribute values. An attribute can be classified as continuous, discrete or textual.

If the attributes are discrete, a commonly used similarity measure is based on the simple matching criterion. The similarity between two nodes in an attributed graph is determined by examining each of the d attributes and counting the number of attribute values they have in common.

For continuous attributes, the most commonly used metric is based on the Euclidean distance.

$$simA(i,j) = \frac{1}{1 + \sqrt{\sum_d (x_i^d - x_j^d)^2}}$$

If the attributes are textual, we first need to transform them into numeric values. A text document can be represented as bag of words. Each word is represented as a separate variable having numeric weight. The most popular weighting schema is tf-idf (term frequency-inverse document frequency). Each document is then represented as a vector of weight. To measure the similarity between two document vectors, cosine similarity is the most widely used metric.

IV. Community detection algorithms

In this section, we present two methods to discover communities in an attributed graph, given a similarity measure.

A. Algorithm SAC1

Our first approach is based on the modification of Newman's well-known modularity function. Given a graph of n nodes and m edges, $G_{i,j}$ represents the link (i,j), d_i is the degree of node i. If a graph is partitioned into K clusters, Newman's modularity [3] can be written as

$$Q_{Newman} = \sum_{l=1}^{K} \sum_{i \in C_l, j \in C_l} S(i,j) \qquad (1)$$

where the link strength $S(i,j)$ between two nodes i and j is measured by comparing the true network interaction G_{ij} with the expected number of connections $(d_i \cdot d_j)/2m$

$$S(i,j) = \frac{1}{2m} \cdot \left(G_{i,j} - \frac{d_i \cdot d_j}{2m} \right)$$

Newman's modularity does not include the attribute similarity between nodes. We define the "modularity attribute" Q_{Attr} of a partition as

$$Q_{Attr} = \sum_{C} \sum_{i,j \in C} simA(i,j) \qquad (2)$$

where $simA$ is the attribute similarity function.
Next, we introduce a composite modularity as a weighted combination of modularity structure (1) and modularity attribute (2)

$$Q = \sum_{C} \sum_{i,j \in C} (\alpha \cdot S(i,j) + (1-\alpha) \cdot simA(i,j)) \qquad (3)$$

α is the weighting factor, $0 \leq \alpha \leq 1$.

The next step is to find an approximate optimization of Q (direct optimization is a NP-hard problem [12]). We follow an approach directly inspired by the Louvain algorithm [6]. The algorithm starts with each node belonging to a separated community. A node is then chosen randomly. The algorithm tries to move this node from its current community. If a positive gain is found, the node is then placed to the community with the maximum gain. Otherwise, it stays in its original community. This step is applied repeatedly until no more improvement is achieved.
When moving node x to community C, the composite modularity gain is calculated as

$$\Delta Q = \alpha \cdot \Delta Q_{Newman} + (1-\alpha) \cdot \Delta Q_{Attr} \qquad (4)$$

in which

- Gain of modularity structure ΔQ_{Newman} :

$$\begin{aligned} \Delta Q_{Newman} &= \sum_{i,j \in C \cup x} S(i,j) - \sum_{i,j \in C} S(i,j) \\ &= \frac{1}{2m} \left(\sum_{i \in C} G_{i,x} - \frac{d_x}{2m} \sum_{i \in C} d_i \right) \end{aligned}$$

- Gain of modularity attribute ΔQ_{Attr} :

$$\begin{aligned} \Delta Q_{Attr} &= \sum_{i,j \in C \cup x} simA(i,j) - \sum_{i,j \in C} simA(i,j) \\ &= \sum_{i \in C} simA(x,i) \end{aligned}$$

The first phase is completed when there is no more positive gain by moving of nodes. Following Louvain, we can reapply this phase by grouping the nodes in the same communities to a new community-node. The weights between new nodes are given by the sum of the weight of the links between nodes in the corresponding communities [6]. To determine the attribute similarity between two communities, we propose two approaches. The first is to sum up the similarity of their members, the second way is to set to the similarity of their centroids.

B. Algorithm SAC2

Our first algorithm SAC1 repetitively checks all nodes, leading to $O(n^2)$ complexity. To reduce the computational cost, we propose another approach that only makes use of a node's nearest neighbors. Given an attributed graph:

Algorithm 1 Structure-Attribute Clustering Algorithm SAC1

Input: An attributed graph $G = (V, E, X)$ and a similarity matrix
Output: A set of communities
Phase 1 : Initialize each node to a separated community
repeat
for $i \in V$ **do**
for $j \in V$ **do**
Remove i from its community, place to j's community
Compute the composite modularity gain ΔQ
end for
Choose j with maximum positive gain (if exists) and move i to j's community
Otherwise i stays in its community
end for
until No further improvement in modularity
Phase 2
- Each community is considered as new node
- Reapply Phase 1

$G = (V, E, X)$, we define a k-nearest neighbor graph (k-NN) $G_k = (V, E_k)$ as a directed graph in which each node has exactly k edges, connecting to its k most similar neighbors in G. The similarity measure between 2 nodes i and j is defined as

$$S(i, j) = \alpha \cdot G_{i,j} + (1 - \alpha) \cdot simA(i, j)$$

where $simA(i, j)$ is the attribute similarity function, $G_{i,j}$ represents the link (i, j). Note that we can replace $G_{i,j}$ by other similarity measurements such as Jaccard similarity, cosine similarity, etc. [13] discussed several similarity metrics based on local information. Similar to the previous algorithm, we use α as a weighting factor.

We apply the measurement S in the first place to construct the nearest neighbor graph. In G_k, a structural edge represents the similarity between nodes (in terms of structure and attribute) in the original graph G.

The naive approach to build k-NN graph uses $O(n^2)$ time and $O(nk)$ space. However substantial effort has been devoted to speed up the process, such as parallel algorithms ([14], [15]), approximation algorithms ([16], [17]). In most recent work, [18] introduced $NN - Descent$, an algorithm for approximate k-NN construction with an arbitrary similarity measure. The method is scalable with the empirical cost $O(n^{1.14})$.

We propose a simple algorithm with two phases: constructing a k-NN graph G_k and finding structural communities in G_k to obtain the final clustering. In Phase 2, various methods can be employed to find communities. In our experiments, we choose Louvain as the detection method

Algorithm 2 Structure-Attribute Clustering Algorithm SAC2

Input: An attributed graph $G = (V, E, X)$
Output: A set of communities
Phase 1: Construct k-NN Graph G_k
Phase 2: Apply detection method to find structural communities in Gk. The result corresponds to the communities in G

because of its scalability. We set k equal to the average degree of the nodes in the graph G.

V. Experimental study

A. Experimental Datasets

We perform experiments to evaluate our algorithm on several real social networks:

Political Blogs Dataset: A directed network of hyperlinks between weblogs on US politics, recorded in 2005 by Adamic and Glance [19]. This dataset contains 1,490 weblogs with 19,090 hyperlinks between these webblogs. Each blog in the dataset has an attribute describing its political leaning as either *liberal* of *conservative*.

Facebook Friendship Datasets: The datasets contain the Facebook networks (from a date in Sept. 2005) from these colleges: Caltech, Princeton, Georgetown and UNC Chapel Hill [20]. The links represent the friendship on Facebook. Each user has the following attributes: ID, a student/faculty status flag, gender, major, second major/minor (if applicable), dormitory(house), year and high school.

DBLP Dataset: A co-authorship network with 10,000 authors, captured from the DBLP Bibliography data in four research areas: database (DB), data mining (DM), information retrieval (IR) and artificial intelligence (AI). Each author has two attributes: prolific and primary topic. Details of this dataset can be found in [11].

One of the most fundamental characteristic of social network is homophily [21]. The principle of homophily states that actors in a social network tend to be similar (i.e., to share some common attributes) with their connected neighbors, or "friends". In order to show this feature, for each attribute a in the dataset (e.g., political view, dormitory, year), we compute the probability that two friends are similar and compare to the probability of a random pairwise sample

$$P_{sl} = P(Similar|Link) = \frac{|(i, j) \in E : \ s.t. \ a_i = a_j|}{|E|}$$

$$P_s = P(Similar) = \frac{|(i, j) : \ s.t. \ a_i = a_j|}{|E| \cdot (|E| - 1)}$$

Table I shows that the similarities between friends are significant higher than random, according to a particular attribute. In Political Blogs, 90% of connected blogs are similar, compared to 49% of random pair. In Caltech network, similarity in dormitory are significant between friends (42%

Table I: Homophily measurement in experimental datasets

Graph	#Nodes	#Edges	Attribute	$\mathbf{P_{sl}}$	$\mathbf{P_s}$
Political Blogs	1,490	16,716	Leaning	0.90	0.49
Caltech	796	16,656	Dorm	0.42	0.12
Princeton	6,596	293,320	Year	0.53	0.13
Georgetown	9,414	425,638	Year	0.58	0.13
UNC	18,163	766,800	Year	0.43	0.15
DBLP	10,000	28,110	Topic	0.35	0.01

compared to 12%). In the graphs Princeton, Georgetown and UNC, friends are more likely to have the same class year. In DBLP, authors are most likely not connected if they do not share the primary topic.

The analysis of homophily demonstrates the correlation between structure and attribute information in real social networks. For that matter, node attributes could provide valuable information to facilitate community discovery.

B. Evaluation Measures

We extract the communities from the above datasets, using 6 different methods:

- Attribute-based clustering: K-means method is used to group nodes based on the similarity in attributes (link information is ignored).
- Random walks: Method proposed by Steinhaeuser et al. [10], based on random walks and hierarchical clustering. The walk length is set to the number of nodes.
- Louvain algorithm on unweighted graph.
- Fast greedy: Method proposed by Clauset et al. [22] based on the greedy optimization of modularity. The graph is weighted by node attribute similarities.
- Our proposed algorithms SAC1 and SAC2.

To evaluate the quality of these methods, we compare the number of communities, size of communities, modularity structure, modularity attribute and additional two measurements: density D and entropy E

$$D = \sum_{c=1}^{K} \frac{m_c}{m}$$

where m_c is number of edges in community c, m is the number of edges in G, K is the number of communities. D reflects the proportion of community intra-links over total number of links. High density denotes good separation of communities.

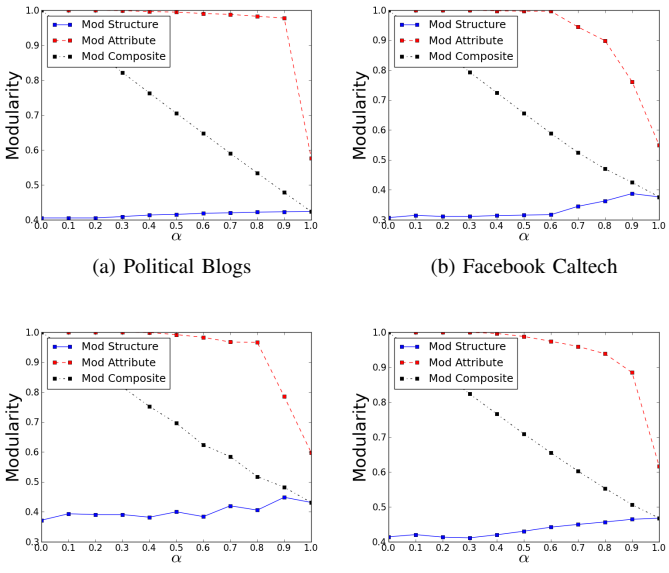
$$E = \sum_{c=1}^{K} \frac{n_c}{n} \cdot entropy(c)$$

$$entropy(c) = -\sum_{i} p_{ic}\ log(p_{ic})$$

where n_c is the number of nodes in community c, n is the number of nodes in G, p_{ic} is the percentage of nodes in c with attribute i. Communities with low entropy means they are more homogeneous with respect to the attribute a_i.

C. Comparison of SAC1 and SAC2

Because our approaches make use of the parameter α as a weighting factor between structural similarities and attribute similarities, we first examine the community qualities with different values of α. Figure 1 plots the modularity structure (E.q (1)), modularity attribute (E.q (2)) and modularity composite (E.q (3)) of SAC1's communities (in 4 graphs), for $\alpha \in [0, 1]$. The x-axis represents the values of α, the y-axis represents the modularities values. There is an increasing trend of modularity structure and decreasing trend of modularity attribute since the algorithm gives more favor to structural similarities as α increases. For SAC2 (not shown here), the modularities also follow the similar patterns.



(a) Political Blogs (b) Facebook Caltech

(c) Facebook Princeton (d) Facebook Georgetown

Figure 1: SAC1 modularity structure, modularity attribute and modularity composite for $\alpha \in [0, 1]$

Table II reports the average entropy and density of SAC1 and SAC2 on the datasets. Average entropy of SAC2 is lower than SAC1's whereas density of SAC1 is higher than SAC2's. That is, SAC2's communities are more homogeneous, but in terms of density, SAC1's communities are more dense.

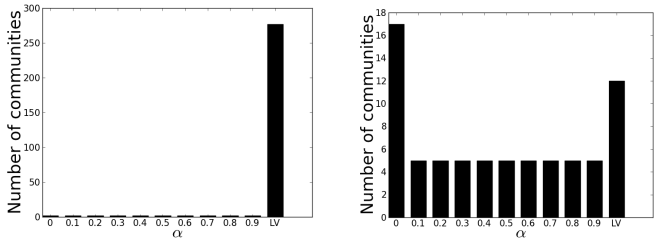
Copyright (c) IARIA, 2012. ISBN: 978-1-61208-176-2

Table II: Average entropy and density of SAC1 and SAC2

Graph	Average Entropy		Average Density	
	SAC1	SAC2	SAC1	SAC2
Political Blogs	0.06	0.1	0.91	0.90
Caltech	0.75	0.33	0.50	0.46
Princeton	1.04	0.41	0.64	0.55
Georgetown	0.91	0.41	0.68	0.60
UNC	1.76	0.51	0.64	0.45
DBLP	3.01	1.24	0.82	0.52

D. Comparison against other methods

1) Number of communities and size distribution: We observe that SAC1 and SAC2 result in less number of communities than other methods. Figure 2 shows the number of communities found by Louvain and SAC1. The x-axis represent values of α, the y-axis is the number of corresponding communities. The outermost right bar is the number of communities from Louvain. It is clear that Louvain results in more communities. The result is similar for SAC2 (Table III). However, many of the communities found by Louvain are very small. For instance in Political Blogs, although 276 communities are found, the biggest two communities already consist of 80 percent of nodes. The rest of communities have the maximum size of 5 nodes. On the other hand, our algorithms correctly identified two communities in this graph, which correspond to two political views: liberal and conservative. It is observed that for large networks, Louvain often results in a few mega-sized communities and numerous small-sized communities. Our methods achieved a more balanced distribution of community sizes.



(a) Political Blogs (b) Facebook Georgetown

Figure 2: Number of communities in SAC1 (plot of α) and Louvain

Table III: Number of communities in SAC2($\alpha = 0.5$), Louvain and Fast greedy

Graph	SAC2	Louvain	Fast greedy
Political Blogs	2	277	277
Caltech	7	10	9
Princeton	7	20	24
Georgetown	9	12	42
UNC	7	19	31
DBLP	47	566	864

2) Community quality: Table IV and V compare the clustering entropy and density (with $\alpha = 0.5$) on two datasets. It shows that SAC1 and SAC2 result in communities with lower entropy (higher attribute similarities) than Louvain and Fast greedy's communities. For example, in Caltech graph, the entropy of SAC1 and SAC2 is 0.75 and 0.33 respectively, while the entropy of Louvain and Fast greedy is 1.65 and 1.71. On the other hand, the density of our methods is a little lower than the density of these two methods but higher than attribute-based clustering and random walks. For other datasets, the results are also similar.

Table IV: Entropy and Density of Caltech's communities

Method	Entropy	Density
Attribute-based	0	0.42
Random walks	0	0.35
Louvain	1.65	0.57
Fast greedy	1.71	0.56
SAC1	0.75	0.50
SAC2	0.33	0.46

Table V: Entropy and Density of Princeton's communities

Method	Entropy	Density
Attribute-based	0	0.53
Random walks	0	0.47
Louvain	1.71	0.62
Fast greedy	1.80	0.74
SAC1	0.84	0.62
SAC2	0.41	0.55

VI. Discussions

Both of our methods are parameterized, i.e., using α as a weighting factor, the natural question is how to choose α. Note that the results are quite stable with respect to α. With no domain knowledge, it is difficult to determine the value of α a priori. However, in social networks, we expect the links contain more information than attribute values. Based on this idea, we propose a strategy to approximate α. It is illustrated below:

init:

- $\alpha = 1$
- Set an interval i (e.g., $i = 0.1$ in our experiments)

repeat

- Compute the optimized clustering corresponding to α
- Let $Q_{Newman}(\alpha)$ and $Q_{Attr}(\alpha)$ be the modularity structure and modularity attribute of the partition
- Let $\alpha' = \alpha - i$
- Let $\Delta = (Q_{Newman}(\alpha') - Q_{Newman}(\alpha)) + (Q_{Attr}(\alpha') - Q_{Attr}(\alpha))$
- $\alpha = \alpha'$

until $\Delta <= 0$

Table VI reports the value of α found using the aforementioned strategy for SAC1 algorithm. It shows that the communities found are reasonably good in terms of modularity values.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-176-2

Table VI: Optimum α found for the graphs

Graph	α	Q_{Newman}	Q_{Attr}
Political Blogs	0.5	0.41	0.99
Caltech	0.6	0.31	0.99
Princeton	0.7	0.42	0.96
Georgetown	0.5	0.43	0.98
UNC	0.6	0.33	0.91
DBLP	0.5	0.27	0.83

VII. Conclusion and perspectives

In this paper, we studied the issue of community detection in attributed graphs. We propose two methods that couple topological structure as well as attribute information in the detection process. Experimental results in real social networks demonstrated that our methods achieve a flexibility in combining structural and attribute similarities, hence able to bring in more meaningful communities. As future work, we try to bring further enhancements to our methods, e.g., reduce the algorithms' complexity, explore different similarity functions. We will apply our methods in different scenarios, for example with textual data or missing attribute values. We try to understand the roles of links and content information in the formation of online communities in order to devise adapted discovery strategies and to model the dynamic of the networks.

Acknowledgment

This work was partially supported by the projects ANR Ex DEUSS and DGCIS CEDRES.

References

[1] L. Tang and H. Liu, *Community Detection and Mining in Social Media (Synthesis Lectures on Data Mining and Knowledge Discovery)*. Morgan-Claypool, 2010, ch. 3.

[2] S. Fortunato, "Community detection in graphs," *Physics Reports 486, 75-174 (2010)*, 2009.

[3] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, p. 066111, 2004.

[4] K. Wakita and T. Tsurumi, "Finding community structure in mega-scale social networks," *Computer Science - Computers and Society, Physics - Physics and Society*, 2007.

[5] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in network," *Phys. Rev. E 69, 026113*, 2004.

[6] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008 (12pp), 2008.

[7] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, Jun. 2005.

[8] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, Jun. 2010.

[9] J. Leskovec, K. J. Lang, and M. W. Mahoney, "Empirical comparison of algorithms for network community detection," *CoRR*, vol. abs/1004.3539, 2010.

[10] K. Steinhaeuser and N. V. Chawla, "Identifying and evaluating community structure in complex networks," *Pattern Recognition Letters*, Nov. 2009.

[11] Y. Zhou, H. Cheng, and J. X. Yu, "Graph clustering based on structural/attribute similarities," *Proc. VLDB Endow.*, vol. 2, pp. 718–729, August 2009.

[12] U. Brandes, D. Delling, M. Gaertler, R. Goerke, M. Hoefer, Z. Nikoloski, and D. Wagner, "Maximizing modularity is hard," *ArXiv Physics eprints*, vol. physics.da, no. 001907, 2006.

[13] T. Zhou, L. Lu, and Y.-C. Zhang, "Predicting missing links via local information," *European Physical Journal B*, vol. 71, no. 4, pp. 623–630, 2009.

[14] M. Connor and P. Kumar, "Fast construction of k-nearest neighbor graphs for point clouds." *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 4, pp. 599–608, 2009.

[15] M. D. Lieberman, J. Sankaranarayanan, and H. Samet, "A fast similarity join algorithm using graphics processing units," *2008 IEEE 24th International Conference on Data Engineering*, vol. 25, no. April, pp. 1111–1120, 2008.

[16] J. Chen, H. Fang, and Y. Saad, "Fast approximate knn graph construction for high dimensional data via recursive lanczos bisection," *Journal of Machine Learning Research*, vol. 10, no. 2009, pp. 1989–2012, 2009.

[17] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions," in *ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS*, 1994, pp. 573–582.

[18] W. Dong, C. Moses, and K. Li, "Efficient k-nearest neighbor graph construction for generic similarity measures," in *Proceedings of the 20th international conference on World wide web*, ser. WWW '11, 2011.

[19] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 u.s. election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*, ser. LinkKDD '05, 2005.

[20] A. L. Traud, E. D. Kelsic, P. J. Mucha, and M. A. Porter, "Comparing community structure to characteristics in online collegiate social networks," 2010, sIAM Review, in press (arXiv:0809.0960).

[21] P. F. Lazarsfeld and R. K. Merton, "Friendship as a social process: A substantive and methodological analysis," in *Freedom and Control in Modern Society*. Van Nostrand, 1954.

[22] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, pp. 1–6, 2004.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-176-2