Maximal Labeled-Cliques for Structural-Functional Communities

Debajyoti Bera

IIIT-Delhi, New Delhi, India 110020,
dbera@iiitd.ac.in

Abstract. Cliques are important building blocks for community structure in networks representing structural association between entities. Bicliques play a similar role for bipartite networks representing functional attributes (*aka.* labels) of entities. We recently proposed a combination of these structures known as labeled-cliques and designed an algorithm to identify them. In this work we show how to use these structures to identify structural-functional communities in networks. We also designed a few metrics to analyse those communities.

1 Introduction

A clique represents a set of mutually related entities in a network and has played an important role in *community detection* and *graph clustering* [19, 6]. Many network analysis methods, e.g., *clique-percolation method* [18] and *maximal clique centrality* [4], rely on the set of maximal cliques of a graph. Therefore, it natural to ask how to extend these results to networks with additional information.

One way to extend cliques would be to incorporate attributes on the nodes. The last decade has witnessed a massive increase in the collection of richer network datasets. These datasets not only contain the inter-entity relationships, but they also contain additional attributes (*aka.* "labels") associated with each entity. For example, social network datasets contain both "structural relationships" (social links between users) and "functional attributes" (interests, likes, tags, etc.). A recent experimental study concluded that real-life communities are formed more on the basis of functional attributes of entities (like interests of users, functions of genes, etc.) rather than their "structural attributes" (those defined using cliques, cuts, etc.)[25]. Naturally, given *both* structural and functional information, we expect to find communities that are bonded on both.

The notion of cliques playing the role of seeds in a community structure ought to be strengthened if we also mandate functional similarity. In this work we address the question "*what is the role of such cliques in discovering cohesive structural-functional clusters?*" We are aware of only two prior solutions for this problem. Modani et al. [13] resolved the problem of finding "like-minded communities in a social network" by reducing it to that of finding maximal cliques in an unlabeled graph. Their solution was applying any graph clustering technique on a subgraph constructed using those maximal cliques. Motivated by a

similar problem, Wan et al. [24] studied the problem of finding communities that are strongly related in terms of both node attributes and inter-node relationships; their solution was a heuristic to avoid generating all maximal cliques. To the best of our knowledge, the first comprehensive graph-theoretic model for structural-functional clusters was given by Bera et al. [2] in the form of *maximal cliques of entities with a maximal set of shared labels*, *aka.* MLMCs. In that work the authors presented the idea, gave an algorithm to find those structures, and merely suggested a use for finding communities. In this work we outline tools and methods to employ MLMCs to analyse networks.

Overview of results: We answer two specific questions. First, how to analyse a graph with the help of its MLMCs? In particular, what would be the statistics of MLMCs in a random graph? And, how far is a network from attaining stability, i.e., when the structural and functional linkages have converged to the same? To answer these questions, we propose a *null model* for labeled-graphs, and then use this null model to define *structural-functional divergence*.

The communities that we focus in this work are built on cliques; however, a clique in itself may be too strict a definition for a community. We devise an extension of the clique-percolation method [18] to labeled-graphs named **CBCPM** that incorporates similarity of labels also while constructing communities. For evaluating the functional cohesion of the communities found by our algorithm, we devise a new metric **ΦC** to overcome a shortcoming of the *likemindedness* measure proposed earlier [13].

The interest in labeled graphs has recently gained popularity and there are now quite a few techniques for clustering them [1, 5].However, every clustering technique emphasises a different notion of community and it appears to be difficult to decide one clear winner. The relevance of this paper is limited only to the scenarions where clique-based communities are logical.

2 Background: maximal-labeled cliques

We represent an undirected unweighted graph G by its sets of vertices and edges, *i.e.*, $G = \langle V, E\rangle$. Similarly, we represent an undirected bipartite graph G by $G = \langle U, V, E\rangle$ where U and V represent the two sets of vertices and E represents the edges going between U and V. Suppose L is a finite discrete set of labels. A labeled-graph $G_L = \langle V, E, L, l\rangle$ is defined as a graph whose vertices have an associated subset of elements chosen from L. For any vertex v, $l(v) \subseteq L$ will be used to denote the labels of that vertex. A labeled-clique (LC) of G_L is defined to be any subset of vertices $V' \subseteq V$ and a subset of labels $L' \subseteq L$ such that (i) there is an edge between every pair of vertices in V', and (ii) for every $v \in V'$, v is labeled using *all* the labels in L'; we denote it $\langle L', V'\rangle$.

Our next notion is for unlabeled graphs that can be considered as a join of a bipartite graph and a general graph. Given a general graph $G_1 = \langle V, E_2\rangle$ and a bipartite graph $G_2 = \langle U, V, E_1\rangle$, a joined-graph is denoted by $\langle U, V, E_1, E_2\rangle$and defined as a network on U and V consisting of both sets of edges E_1 and E_2. Observe that there are edges among vertices in V (E_2) and between vertices in

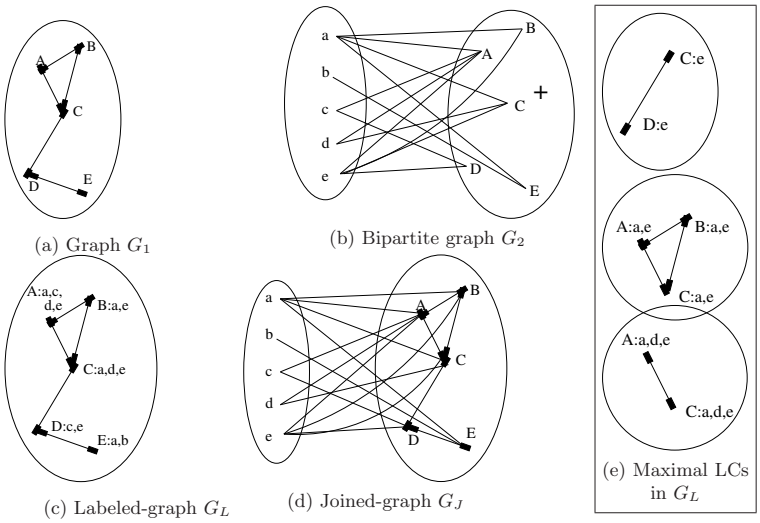


Fig. 1. Labeled-graph G_L combines G_1 and G_2. G_J is the joined-graph representation of G_L. (Figure is reproduced from [2] with permission.)

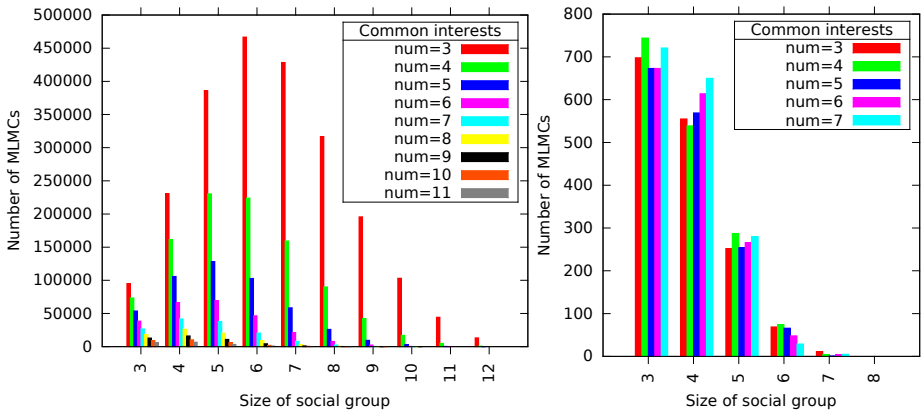
U and V (E_1) but none among vertices in U. It was shown by Bera et al. [2] that a labeled graph can be treated as a joined graph and vice versa.

An MLMC — maximal clique with maximal set of labels, is a labeled-clique which does not remain an LC if we add any more vertex or label.

All of these concepts can be understood with the help of Figure 1. It shows a network of entities $\{A, B, C, D, E\}$ as the general graph G_1 and Figure 1b shows their association with labels from $\{a, b, c, d, e\}$ as the bipartite graph G_2. Figure 1c shows a labeled-graph G_L that combines the information from G_1 and G_2 and $\langle\{a, e\}, \{A, C\}\rangle$ is an LC in G_L.

Examples: We present two examples to illustrate how MLMCs can help in analysing networks. Figure 2 presents the number-*vs*-size distribution of the MLMCs of two social-network datasets with tens of thousands of links and labelings (representing "user interests") Not only the number of MLMCs of different sizes follow markedly different distributions, observe that the number of MLMCs with 5 (or 3 or 4) users are mostly same in the "Last.fm" dataset, whereas, the same number follows a rapidly decreasing trend in the "The Marker Cafe" dataset. Our explanation is that users of networks based on user-ratings (Last.fm) do not necessarily compare and correlate their ratings but users of a social network (The Marker Cafe) have a natural tendency to bond over shared interests. Such insights are attractive for targeted advertisement and personalized recommendation.

Table 1 shows some of the patterns we obtained by analysing the MLMCs of a DBLP dataset of papers published within 1984–2011 in data mining and related venues [23] — considering only the top venues and authors with 40+ papers in them. We wanted to know which scientists are not collaborators but



(a) The Marker Cafe (b) Last.fm

Fig. 2. MLMC profiles of social network datasets.

Table 1. Groups of prolific authors who share (pairwise) a common coauthor but are not collaborators despite having concurrent papers at common venues

Authors	@ **Venues**
Philip S. Yu, Heikki Mannila, Tao Li	@ TKDE(2008,2009), Know. Inf. Sys. (2005–2008,2010,2011), ICDM (2002,2006), SDM(2008–2010)
Jian Pei, Christos Faloutsos, Wei Fan	@ ICDM(2005,2006,2008,2010), SDM(2007,2008,2011), CIKM(2009), KDD(2004,2006,2008-2011)

could easily be so. For that we constructed a labeled-graph of scientists in which the labels represented the venues of their papers. We linked two scientists if they have *do not* have a joint paper but share a common coauthor — roughly indicating a shared interest. We discovered 58 MLMCs that consisted of at least 3 authors and at least 10 venues; two such MLMCs are shown in Table 1. All such MLMCs represent potential collaborative groups that could have been formed due to familiarity (common coauthors) and concurrency (same venues).

3 Community Detection

Now we discuss how our labeled-cliques can help us find tightly bonded communities. Our objective is to establish *proof-of-concept application* of MLMCs; in reality, each network requires its own bespoke notion of community. The reader may refer to a recent survey [5] for many such techniques for labeled graphs.

3.1 Null model for labeled-graphs

A common tool in network analysis is a *null model* that is a random graph with specific desirable properties. They are used to analyse networks, e.g., distinctness of network from a randomly formed one, quality of a network clustering [7], etc.

For example, the well-known notion of network modularity [16] uses a null model that preserves the expected degree of vertices. We use a null model that preserves the degree distribution of labeled-graphs. Given a labeled-graph $G = \langle V, E, L, l \rangle$, consider an equivalent joined-graph and denote its *bipartite component* as G^B and *general component* as G^N. We define null model for the labeled-graph G by simply joining the null models for G^N and G^B, which we describe below.

Null model for G^N: For the non-bipartite component we use the well-studied *Configuration Model*(CM) [14, 15] that creates a degree-preserving random graph. Consider a random approach that starts with an empty graph, picks two of the "unsaturated" vertices uniformly at random and connects them by an edge; a vertex is saturated when its number of edges equals its degree in G^N.

Null model for G^B: We extend CM and generate a random graph with the same degrees as in G^B. The BiCM null model also generates graphs with the same properties [20], however, they use entropy-maximization unlike our combinatorial approach. We will, anyhow, denote our model too by BiCM.

We will follow the exact same approach as in CM and add edges between two randomly chosen unsaturated vertices, one each from L and V. Clearly, the final random graph has the same degrees as in G^B and also the same number of edges. Favoring simplicity, we allow the random graph to have multiple edges between vertices just like in CM.

Next we state a technical lemma on the expected number of common labels in BiCM. Consider any labeled-graph G from BiCM, and further, consider any two nodes $u, v \in V$ and any label $l \in L$. Let $N^l_{u,v}$ denote the indicator variable that is 1 iff l is the labeling of both u and v; further, let $N_{u,v} = \sum_{l \in L} N^l_{u,v}$ denote the number of common labels. Let m denote the number of edges, d_u and d_v denote the degrees of u and v and c_l denote the number of nodes which have the label l.

Lemma 1. *The expected value of* $N_{u,v}$ *is* $\displaystyle\sum_{\substack{l \in L \\ c_l \geq 2}} \frac{1}{\binom{m}{c_l}} \sum_{\substack{r=1\ldots d_u \\ g=1\ldots d_v}}^{r+g \leq c_l} \binom{d_u}{r} \binom{d_v}{g} \binom{m - d_u - d_v}{c_l - r - g}$

Proof (Proof sketch). A standard approach is to attach $deg(x)$ stubs to a vertex x and connect to unassigned stubs at each step. Then the probability of selecting c_l stubs from the nodes, where there are d_u stubs from u, d_v stubs from v and $(m - d_u - d_v)$ other stubs, follows a trivariate hypergeometric distribution. $\mathbb{E}[N^l_{u,v}]$, which is same as the probability of selecting at least one stub of u and v each, can be now easily calculated from which the lemma follows.

An equivalent, but easier to compute, expression for $\mathbb{E}[N^l_{u,v}]$ can be obtained by applying the Chu-Vandermonde identity:

$$\mathbb{E}[N^l_{u,v}] = \left[\binom{m}{c_l} + \binom{m-d_u-d_v}{c_l} - \binom{m-d_u}{c_l} - \binom{m-d_v}{c_l}\right] / \binom{m}{c_l}$$

3.2 Structural-Functional Divergence

The labeled-graphs represent two networks — one composed of structural links between nodes and another representing functional attributes. We conjecture

Algorithm 1 CBCPM: Finding overlapping SF clusters

Input: Labeled-graph $G_L = \langle V, E, L, l\rangle$
Output: Overlapping clusters of V
Percolation parameters: $k_l, k_s \in \mathbb{Z}+$

$\mathcal{L} \leftarrow$ list of MLMCs of G_L with $\geq k_l$ labels & $\geq k_s$ vertices.
Form MLMC-overlap network $\mathcal{N}$:
    Each node of $\mathcal{N}$ is an MLMC M_i of $\mathcal{L}$
    Edge between $M_i = \langle L_i, V_i\rangle$ & $M_j = \langle L_j, V_j\rangle$ if
        $|L_i \cap L_i| \geq k_l - 1$ & $|V_i \cap V_j| \geq k_s - 1$
Obtain list $\mathcal{C}$ of connected components of $\mathcal{N}$
for all connected component $C \in \mathcal{C}$ **do**
    Output cluster $\{v \ : \ \exists \langle L', V'\rangle \in C, \quad v \in V'\}$

that in many domains these two networks may converge with time as the nodes forge new structural links based on functional similarities or acquire new functionalities based on structural linkages. One way to measure the (dis)similarity of these two networks is to compare the general component with a monopartite projection of the bipartite component. For the latter, we fall back on the BiCM null model instead of other proposed approaches [21, 11]; the correct projection method really depends upon the application and was not investigated further. $\mathbb{E}[N_{u,v}]$ is computed on G^B in the definition below.

Definition 1. *Given a labeled-graph $G = \langle V, E, L, l\rangle$, define its (λ, κ)-functional projection as an unlabeled graph G' on V in which an edge exists between u & v if $|l(u) \cap l(v)| \geq \min\{\lambda, \kappa\mathbb{E}[N_{u,v}]\}$. Let $CC(G)$ and $CC(G')$ denote the mean clustering coefficient of G and G', respectively. (λ, κ)-structural-functional divergence of G is defined as: $\Delta^{SF}_{\lambda,\kappa}(G) = CC(G)/CC(G')$.*

Choose some $\kappa > 1$. If there are $\kappa\mathbb{E}_{u,v}$ or more common labels between u and v, then this indicates a strong functional similarity between u and v when compared to the null model. The parameter λ is used for additional restrictions on the minimum functional similarity.

3.3 Structural-functional clustering

The *clique percolation method* (CPM) is a popular method for clustering of entities in a network considering only the structural links. This method identifies overlapping clusters which are composed of several (overlapping and maximal) cliques [18]. We are interested in clustering entities that are closely related both structurally and functionally. A previous approach by Modani et al. [13] first finds all MLMCs with a minimum number of nodes and common labels. Then it obtains the subgraph induced by the nodes of the MLMCs. They rightly claim that this subgraph is made up of those nodes that are better connected both structurally and functionally. The authors then proposed to run any suitable overlapping (or non-overlapping) algorithm (e.g., CPM) on this subgraph.

However, we think better clusters can be obtained if the functional similarity is in-grained deeper in the cluster finding algorithm. Hence, we propose a

"Clique-Biclique Percolation Algorithm" (**CBCPM**) outlined in Algorithm 1. Like CPM, the clusters discovered by **CBCPM** are composed of maximal LCs. Each cluster is constructed from several LCs that are "connected" — two LCs are said to be connected if they overlap in at least k_s nodes and at least k_l labels. The output of the algorithm are clusters of nodes from the connected components of the network of maximal LCs.

3.4 Quality of structural-functional clustering

Finally, we study how to quantify the *quality* of *overlapping* clusters in a network. Following the approach of Modani et al., we consider one measure for the structural closeness of clusters and another for their functional similarity (or cohesion). If necessary, a weighted sum of both the measures can be used to construct a single measure of quality.

Suppose we are given clusters $\mathcal{C} = \{C_1, C_2, \ldots C_k\}$ in a labeled-graph $G_L = \langle V, E, L, l\rangle$ where C_is are subsets of V, not necessarily disjoint. We will use e to denote the number of structural links in G_L. Let $\delta(u, v)$ denote an indicator variable for u and v co-occurring in some cluster together and similarly, $E(u, v)$ indicate an edge between u and 4. $d(u)$ will denote the degree of a node $u \in V$ within the general component and $l(u)$ will denote its labels. For any label s, let $c(s)$ denote the set of users that have s as one of their labels. $comm(u_1, u_2, \ldots)$ shall denote the set of clusters that contain all of the nodes $u_1, u_2, \ldots$. Even though the clusters constitute only nodes, we will informally store the maximal set of common labels of all the nodes within each cluster.

Structural quality: There are already a large number of options to choose from for structural quality. For our experiments, we chose a generalization of the highly popular Newman-Girvan "modularity" measure [16] that was proposed by Shen et al. [22]. These are built upon the notion of "coverage" and a null model. Coverage of a clustering is defined as the fraction of intra-cluster edges: $Cov(\mathcal{C}) = \frac{1}{2e}\sum_{C\in\mathcal{C}}\sum_{u,v\in C} E(u, v) = \frac{1}{2e}\sum_{u,v} E(u, v)\delta(u, v)$

Modularity was initially defined for disjoint clusters. To apply this to overlapping clusters, a common trend is to use the notion of "belongingness" [17]. Shen et al. defined the contribution of a node u towards a cluster C as $\beta_{u,C} = \frac{1}{|comm(u)|}$ if $u \in C$ and 0 otherwise, and used it to define a generalized modularity OQ [22].

$$OCov(\mathcal{C}) = \frac{1}{2e}\sum_{C\in\mathcal{C}}\sum_{u,v\in C} E(u, v)\beta_{u,C}\beta_{v,C}$$

$$OQ(\mathcal{C}) = OCov(\mathcal{C}) - \mathbb{E}[OCov(\mathcal{C})] = \frac{1}{2e}\sum_{C\in\mathcal{C}}\sum_{u,v\in C}\left[E(u, v) - \tfrac{d_u d_v}{2e}\right]\beta_{u,C}\beta_{v,C}$$

Functional quality: Despite several measures to quantify the similarity of nodes in a bipartite network, the only measure we found that was given explicitly for functional cohesion was "likemindedness" (LM) [13]. Let $\mathcal{S} : V \times V \to \mathbb{R}[0, 1]$ be a relevant measure for the functional similarity of two vertices, e.g., Jaccard similarity, Hadamard similarity, etc. Modani et al. defined likemindedness as

the average similarity of all intra-cluster pairs of nodes (including pairs with duplicates, to remain consistent with modularity as hinted by the authors): $LM(\mathcal{C}) = \sum_{C\in\mathcal{C}} \sum_{u,v\in C} \mathcal{S}(u,v) / \sum_{u,v} \delta(u,v)$

Consider a clustering in which there is one cluster with the two most similar nodes and all other nodes are in a single-member cluster each. It is easy to show that these clusters attain the maximum LM of $\max_{u\neq v} \mathcal{S}(u,v)$ among all clusterings. This led us to conclude that LM favors smaller, in fact, single or two membered, communities — not really a worthwhile measure of cluster quality.

This prompted us to define a new metric ΦC for functional cohesion. First, we define "cohesion" of a clustering as the fraction of intra-cluster similarities over total similarity, enhanced with belongingness.

$$Coh^{\mathcal{S}}(\mathcal{C}) = \sum_{C\in\mathcal{C}} \sum_{u,v\in C} \mathcal{S}(u,v)\beta_{u,C}\beta_{v,C} / \sum_{u,v} \mathcal{S}(u,v)$$

Definition 2. *For any similarity metric $\mathcal{S}$ and a clustering $\mathcal{C}$ of a labeled-graph, let $\mathbb{E}[Coh^{\mathcal{S}}(\mathcal{C})]$ be the expected cohesion in a corresponding BiCM random graph. Then functional modularity can be defined as: $\Phi C^{\mathcal{S}}(\mathcal{C}) = Coh^{\mathcal{S}}(\mathcal{C}) - \mathbb{E}[Coh^{\mathcal{S}}(\mathcal{C})]$*

Construct a complete weighted graph G' on V with weight of any edge (u,v) equal to $\mathcal{S}(u,v)$. By construction, the functional modularity on G is same as the overlapping modularity of G'.

For our experiments we used the Hamming similarity metric $\mathcal{S}_H$ which is simply the fraction of labels that u and v have in common. Note that Coh and $\mathbb{E}[Coh]$ are not affected by the normalization factor. Instead, $\mathbb{E}[Coh]$ depends upon the edges which is governed by the null model. The following lemma will be useful in simplifying the denominator of $\mathbb{E}[Coh^{\mathcal{S}_H}]$. Recall that in the BiCM null model, the degree sequence of all nodes and all labels are fixed.

Lemma 2. *Consider all graphs with a fixed set of labels, say L, and in which, $|c(l)|$ is fixed for every $l \in L$. Then,*

$$\sum_{u,v} \mathcal{S}_H(u,v) = \frac{1}{\sigma} \sum_{l\in L:|c(l)|>1} \binom{|c(l)|}{2}$$

The proof uses a simple double-counting of the nodes with a particular label. The denominator in $\mathbb{E}[Coh^{\mathcal{S}_H}]$ (and also in $Coh^{\mathcal{S}_H}$) therefore becomes a constant independent of the (random) graph. Furthermore, observe that $\mathbb{E}[\mathcal{S}_H(u,v)]$ in the random graph is same as $\mathbb{E}[N_{u,v}]$ in G^B (defined earlier).

Theorem 1. *Functional modularity of a clustering $\mathcal{C}$ under Hamming similarity can be computed as:*

$$\Phi^{\mathcal{S}_H}(\mathcal{C}) = \sum_{C\in\mathcal{C}} \sum_{u,v\in C} \Big[\mathcal{S}(u,v) - \mathbb{E}[N_{u,v}]\Big]\beta_{u,C}\beta_{v,C} \Bigg/ \sum_{\substack{l\in L \\ |c(l)|>1}} \binom{|c(l)|}{2}$$

4 Evaluation Results

To evaluate the effectiveness of our approaches, we applied them to several real-life datasets (described in Table 2). The "Twitter-small" dataset is constructed from the Twitter dataset [10] with edges representing "following a celebrity"; we selected as labels those users with followers between 15000 and 16000 (i.e., celebrities) and for nodes, those non-celebrities with 6000-65000 followers.

Table 2. Labeled-graph datasets used for experimental evaluation.

Dataset	Type	Links represent ...	Labels represent ...	Nodes	Labels	Node links	Labelings	Num. of MLMC
Ning Creators' Net. (Ning) [12]	Social network	Friends	Group affiliation	11011	81	76262	4812	5459
'Café TheMarkers's (CTM) [12]	Social network	Friends	Group affiliation	93664	88	1.74M	221610	34.7M
Ciao DVD (Ciao) [8]	Ratings of DVD reviews	Mutual trust	Reviews rated more than 2/5	20336	66109	7017	1.52M	79029
Filmtrust (FT) [9]	Movie ratings	Mutual trust	Movies rated more than 2/5	1530	1881	544	28580	1996
Last.fm (Lfm) [3]	Social net. of music listeners	Friends	Artists listened to	1892	17632	25434	92834	32344
Twitter-small (TwS) [10]	Social network	Mutual followers	Celebrities followed	1150	276	45360	42658	140M

4.1 SF-divergence

First we report the SF-divergence of our labeled-graph datasets in Table 3; we skip Ciao since it involved computing CC for a large number of nodes and labels which did not finish within a day.

A SF-divergence value less than one of indicates that there are several nodes that share functionalities but are yet to form structural links. On the other hand, a value more than one indicates that nodes are yet to fully acquire functionalities from structurally connected nodes. We conjecture that the SF-divergence of a static social network (in which users are not joining or leaving) should approach one in long term. We can see that the CTM and TwS networks display this behavior better than the other networks. This is expected for the TwS dataset since the "labels" in this network are celebrities and two users who follow each are more likely to follow the same celebrities. CTM users anyway show a highly "matured" behavior as was observed earlier in Figure 2a.

4.2 Discovering overlapping communities

Now we report the quality of overlapping communities obtained by our CBCPM algorithm (Algorithm 1). Our goal was to show that, for similar setting of pa-

Table 3. SF-divergence values

Dataset	$\Delta^{SF}_{2,3}$
FT	0.28
Ning	0.39
Lfm	0.27
CTM	0.82
Tws	0.85

Table 4. Quality of Ning and FT communities

Dataset	Parameters		Method	**OQ** [22]	**LM** [13]
Ning (*)	$k_l = 3$	$k_s = 4$	CBCPM	0.05	3.38
		$k_s = 5$	CPMCore	0.03	2.17
FT	$k_l = 3$	$k_s = 3$	CBCPM	0.35	5.27
			CPMCore	0.41	4.91
	$k_l = 4$	$k_s = 3$	CBCPM	0.27	5.51
			CPMCore	0.39	4.93

(*) Best k_s for $k_l = 3$ is used that maximized **LM**.

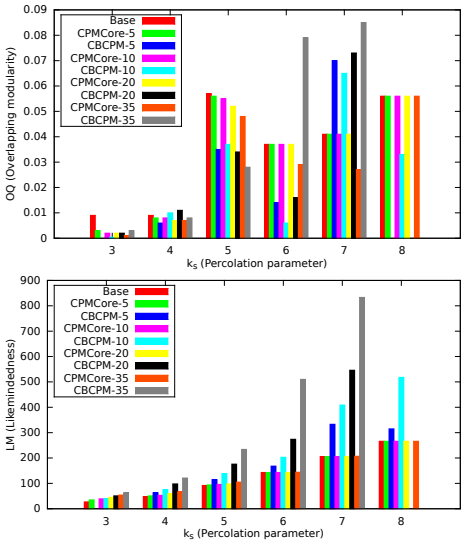


Fig. 3. Quality of Ciaodvd communities

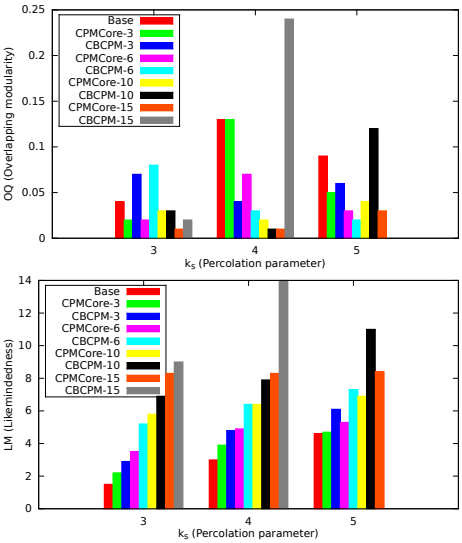


Fig. 4. Quality of Last.fm communities

rameters k_s and k_l, CBCPM creates communities with better likemindedness than the existing CPMCore method [13] of running the CPM algorithm on the subgraph of nodes that are present in the MLMCs with at least k_l labels and k_s nodes. These parameters are related to the "percolation" of clique/labeled-cliques and has to be chosen carefully that was beyond our scope. Too large values may not find any community and too small values will create a single community. Therefore, we conducted experiments with different values of $k_s \geq 3$ and that of $k_l \geq 3$ and only considered clusters with at least two communities. We compared the overlapping modularity [22] (**OQ**) and the likemindedness [13] (**LM**) of the communities obtained by our CBCPM algorithm *vs.* those given by CPMCore [13]. We used the unnormalized Hamming similarity for $\mathcal{S}()$.

The Ning and the FT datasets generated very few MLMCs for some parameters. Therefore, we set $k_l = 3,\ k_s \geq 3$ for Ning which generated 118 MLMCs. Similarly, we used $k_l \geq 3,\ k_s = 3$ for the FT dataset that gave us 72 MLMCs. Results for the two clustering algorithms are presented in Table 4.

The quality measures of the larger Ciao and Lfm datasets are illustrated in Figures 3 and 4, respectively. We tried several different values of k_l (indicated as CBCPM-k_l and CPMCore-k_l) and k_s (X-axis). We observed that CBCPM consistently found communities with higher LM compared to those found by CPMCore. Due to the stronger enforcement of functional similarity, CBCPM modularities are expected to be lower; however, we observed that the change is highly non-uniform here and sometimes even higher. We conclude that, in comparison to CPMCore, CBCPM finds communities with better functional qualities and with competitive structural qualities.

5 Conclusion

Labeled-graphs are a richer representation of networks that can also store attributes of nodes, apart from the usual node-node relationship, and has been gaining popularity. In this work we show how to analyse the maximal labeled-cliques of these graphs, a concept that was recently introduced [2], and then show how to use those structures to identify clique-based communities. We also introduce a null model and a statistic to represent the attribute-level similarities within a community.

References

1. Baroni, A., Conte, A., Patrignani, M., Ruggieri, S.: Efficiently clustering very large attributed graphs. In: 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). pp. 369–376 (2017)
2. Bera, D., Esposito, F., Pendyala, M.: Maximal labelled-clique and click-biclique problems for networked community detection. In: 2018 IEEE Global Communications Conference (GLOBECOM). pp. 1–6 (2018)
3. Cantador, I., Brusilovsky, P., Kuflik, T.: 2nd Workshop on Information heterogeneity and fusion in recommender systems (HetRec 2011). In: Proceedings of the 5th ACM Conf. on Recommender systems (2011)
4. Chin, C.H., Chen, S.H., Wu, H.H., Ho, C.W., Ko, M.T., Lin, C.Y.: cytohubba: identifying hub objects and sub-networks from complex interactome. BMC Sys. Bio. 8(4) (2014)
5. Chunaev, P.: Community detection in node-attributed social networks: A survey. Computer Science Review 37, 100286 (2020), http://www.sciencedirect.com/science/article/pii/S1574013720303865
6. Faghani, M.R., Nguyen, U.T.: A study of malware propagation via online social networking. In: Mining Social Networks and Security Informatics. Springer Netherlands (2013)
7. Fortunato, S.: Community detection in graphs. Physics Reports 486(3) (2010)
8. Guo, G., Zhang, J., Thalmann, D., Yorke-Smith, N.: Etaf: An extended trust antecedents framework for trust prediction. In: Proceedings of the 2014 International Conf. on Advances in Social Networks Analysis and Mining (2014)
9. Guo, G., Zhang, J., Yorke-Smith, N.: A novel bayesian similarity measure for recommender systems. In: Proceedings of the 23rd International Joint Conf. on Artificial Intelligence (2013)
10. Kwak, H., Lee, C., Park, H., Moon, S.: What is Twitter, a social network or a news media? In: Proceedings of 19th International Conf. on World Wide Web (2010)
11. Latapy, M., Magnien, C., Vecchio, N.D.: Basic notions for the analysis of large two-mode networks. Social Networks 30(1) (2008)
12. Lesser, O., Tenenboim-Chekina, L., Rokach, L., Elovici, Y.: Intruder or welcome friend: inferring group membership in online social networks. In: Social Computing, Behavioral-Cultural Modeling and Prediction (2013)
13. Modani, N., Nagar, S., Shannigrahi, S., Gupta, R., Dey, K., Goyal, S., Nanavati, A.A.: Like-minded communities: bringing the familiarity and similarity together. World Wide Web 17(5) (2014)
14. Newman, M.E.J.: The structure and function of complex networks. SIAM Review 45(2) (2003)

15. Newman, M.E.J.: Networks: an introduction (2010)
16. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E 69(2) (2004)
17. Nicosia, V., Mangioni, G., Carchiolo, V., Malgeri, M.: Extending the definition of modularity to directed graphs with overlapping communities. Journal of Statistical Mechanics Theory and Experiment 2009 (2008)
18. Palla, G., Derényi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature 435(7043) (2005)
19. Plantié, M., Crampes, M.: Survey on social community detection. In: Social media retrieval (2013)
20. Saracco, F., Di Clemente, R., Gabrielli, A., Squartini, T.: Randomizing bipartite networks: the case of the world trade web. Scientific Reports 5 (2015)
21. Saracco, F., Straka, M.J., Clemente, R.D., Gabrielli, A., Caldarelli, G., Squartini, T.: Inferring monopartite projections of bipartite networks: an entropy-based approach. New Journal of Physics 19(5) (2017)
22. Shen, H., Cheng, X., Cai, K., Hu, M.B.: Detect overlapping and hierarchical community structure in networks. Physica A: Statistical Mechanics and its Applications 388(8) (2009)
23. Spyropoulou, E., De Bie, T., Boley, M.: Interesting pattern mining in multi-relational data. Data Mining and Knowledge Discovery 28(3) (2014)
24. Wan, L., Liao, J., Wang, C., Zhu, X.: Jccm: Joint cluster communities on attribute and relationship data in social networks. In: Proceedings of 5th International Conf. on Advanced Data Mining and Applications (2009)
25. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. Knowledge and Information Systems 42(1) (2015)