A Novel Criterion for Overlapping Communities Detection and Clustering Improvement

Alessandro Berti
Department of Mathematics
University of Padova
35121 Padova
Email: berti@math.unipd.it

Alessandro Sperduti
Department of Mathematics
University of Padova
35121 Padova
Email: sperduti@math.unipd.it

Andrea Burattin
Department of Mathematics
University of Padova
35121 Padova
Email: burattin@math.unipd.it

***Abstract*—In community detection, the theme of correctly identifying overlapping nodes, i.e. nodes which belong to more than one community, is important as it is related to role detection and to the improvement of the quality of clustering: proper detection of overlapping nodes gives a better understanding of the community structure. In this paper, we introduce a novel measure, called cuttability, that we show being useful for reliable detection of overlaps among communities and for improving the quality of the clustering, measured via modularity. The proposed algorithm shows better behaviour than existing techniques on the considered datasets (IRC logs and Enron e-mail log). The best behaviour is caught when a network is split between micro-communities. In that case, the algorithm manages to get a better description of the community structure.**

I. Introduction

In business contexts, social networks related research questions are gaining popularity [1], [2]. Indeed, top management can be interested in analysing social networks involving *professional relations* (i.e. relations inside an organization [3]) to better understand the functioning of the company ("mining" it, see [4]). Maps involving professional relations can be built by direct observation of the relations that hold among workers (e.g., John exchanges very often e-mails with Tom), or by using metrics related to business processes. Examples of works covering this topic are [5] and [6], where the focus is in approximating and exploring corporate social networks built on information gathered from e-mails and web use, or [7] which does study social and temporal structures in everyday collaboration among workers. Concerning metrics related to business processes, Van der Aalst et al. [3] describe some metrics, computed over event logs[1], between workers. For example, *Handover of Work* (HoW) is, roughly speaking, a measure of how many times the work of an individual for a given case is followed by the work of another individual; *Similar Activities* (SA) is a measure of similarity between activities performed by two workers.

Given a social network involving a specific relation, a common analysis consists in studying community structure of the organization [9], [10], [11], which is all about grouping the individuals by their similarity. This can be done using a clustering algorithm[2] (e.g., [12], [13], [14], [15], [16], [17], [10]). Unfortunately, clustering is an ill-defined task, which makes it difficult to evaluate the quality of the output of clustering algorithms. In the context of social networks, the most popular criteria to judge the quality of a clustering is *modularity*. Modularity is a concept, described also in [12], that aims to measure group cohesion inside communities and separation between them. Although with some limits (see [18]), modularity is still the most adopted approach in judging clustering quality. Some clustering algorithms try to maximize directly modularity (e.g. [12]). Finding the global maximum of modularity (i.e. the best value) is an hard task, infeasible for large graphs. However finding a good value of modularity can be done in nearly linear time, using the Multilevel algorithm described in [12]. Recently, there has been a growing interest in *spectral* clustering algorithms (see e.g., [16], [19]) as well as on algorithms exploiting special matrix factorization algorithms, such as Nonnegative Matrix Factorization (e.g., [20], [21], [22]). These algorithms, however, are generally expensive from a computational point of view, so iterative algorithms not requiring expensive matrix operations, such as the Multilevel algorithm, are often used in practice since they can handle larger graphs. Although the results obtained by the Multilevel algorithm are in general satisfying, in presence of overlapping communities the performance of the algorithm degrades.

Detection of nodes that lays at the intersection of overlapping communities (hereafter we will refer them as *overlapping nodes*) is important for two main reasons: *i)* they represent individuals that cover an important bridging role among communities, and often they are key individuals when considering the social dimension of the network; *ii)* being at the borders of communities, they are starting points for strategies aiming at improving modularity.

We feel then necessary to introduce a new method to detect overlapping nodes, in order to effectively improve the quality of the communities detected in presence of overlaps. Specifically, we have two main, strongly interconnected, objectives:

- to find the set of nodes that are overlapping among distinct communities, in order to extract informations about them;
- to improve the quality of clustering in presence of nodes at the intersection of overlapping communities, so to allow the user to get more reliable information about the different communities constituting the

[1]Event logs are time-ordered collections of data concerning executions of activities performed by workers or by support systems. See [8] for more details.

[2]We are considering crisp clustering, where a node belongs only to a cluster.

Fig. 1. Visualization of a social network where communities overlap. Overlapping nodes that threaten the quality of a clustering are emphasized. Also other nodes have edges that go outside their cluster, but they belong more clearly to a single community.

network (see Figure 1 to understand the reason why overlapping nodes are dangerous for clustering).

The basic idea underpinning the approach proposed in this paper, which shares some similarities with RaRe [23], consists in: *i)* computing an initial hypothesis of community detection (clustering); *ii)* detecting overlapping nodes; *iii)* computing a new clustering after removal of overlapping nodes; *iv)* inserting back the removed nodes in such a way to improve modularity of the resulting clustering. This process can be iterated multiple times, till modularity converges to a (local) maximum value. Overlapping nodes detection is performed by a novel measure called *cuttability*.

Overall, the proposed approach requires a little more computational time than algorithms like RaRe, however this extra work is rewarded by an effective detection of overlapping nodes and by an improvement of clustering quality, according to modularity. This is demonstrated on social networks built from datasets involving real data and by using the HoW and SA metrics previously introduced.

II. Preliminaries

We represent a social network as a weighted graph $G = (V, E, W)$, where:

- nodes represent individuals (workers), and are identified by integers. Thus V, the set of nodes, is a subset of $\mathbb{N}$;
- edges represent relations between individuals, and are identified by couples $e = (i, j)$ (where i and j are identifiers of nodes). The set of edges E is a subset of $V \times V$;
- weights W are associated to edges, and they represent the *strength* of the relationship represented by the corresponding edge. Mathematically, they can be understood as functions from E to $\mathbb{R}$. Given an edge $(i, j) \in E$ the associated weight is denoted as $w_{i,j} \in \mathbb{R}$.

Weighted graphs can be directed (i.e. edge (i, j) can have a different weight in comparison to edge (j, i)) or undirected (i.e. $(i, j) \in E \iff (j, i) \in E$ and $w_{i,j} = w_{j,i}$).

A clustering C of G is a family of subsets $S_1, \ldots, S_n$ of V for which $S_i \cap S_j = \emptyset$ for $i \neq j$ and $\cup_{k=1,\ldots,n} S_k = V$. So, each node is assigned to exactly one cluster and we can define a function $C : V \to \mathbb{N}$ where $C(v) = i \iff v \in S_i$ (v belongs to the cluster S_i). Clustering methods, as explained in the introduction, try to maximize a quality function, for example Modularity [12]. Modularity can be defined as

$$Q = \frac{1}{2m} \sum_{i,j} \left[w_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$

where $k_i = \sum_j w_{ij}$ is the sum of the weights of the outgoing edges of node i, c_i is the community to which node i is assigned, the δ-function $\delta(u, v)$ is 1 if $u = v$ and 0 otherwise, and $m = \frac{1}{2} \sum_{ij} w_{ij}$.

The Multilevel algorithm [12] aims to reach a good modularity value. It is an agglomerative method starting with each node forming an isolated cluster. Then, following a given order over nodes[3], each node is examined: if there exists an assignment of the current node to a different cluster which maximizes the overall modularity[4] the node is assigned to that cluster. This reassignment process is terminated when no further changes for any node can be performed, i.e. a *local* maximum of modularity is reached.

III. Overlapping nodes

Overlapping communities [10], [11], [17] are communities, in the considered social network, with strong connections between them. Some individuals can be considered to be part of each of the overlapping communities, although the clustering algorithm has to assign them to a single community. These individuals have an important role in community detection as their correct assignment is key to get a good quality clustering.

Finding overlapping nodes is related to finding roles inside an organization [24]. Indeed, individuals which are overlapping between communities are usually "strong communicators" (i.e. lawyers, HR, ...) or belong to management. In addition, their detection can be important for business process improvement, because communities have usually a specific role inside an organization (for example, are related to a specific process or activity). Finding communities with a large number of overlapping nodes can be a signal of inefficiency inside the processes of an organization.

Current approaches for detecting overlapping communities do not seem satisfying when related to approaches that maximize modularity (in presence of overlaps). The most famous approach for overlapping node detection is the "Palla percolating cliques algorithm" [10]. It is based on the observation that intercommunity edges are not likely to form cliques (i.e. a closed path); so, two communities are overlapping when there are many intercommunity edges which form cliques. This does not help to maximize modularity as it does not offer a

[3]The authors of [12] claim that the choice of the ordering does not significantly affect the quality of the obtained clustering.

[4]In [12], authors provide an efficient difference formula for recalculating modularity.

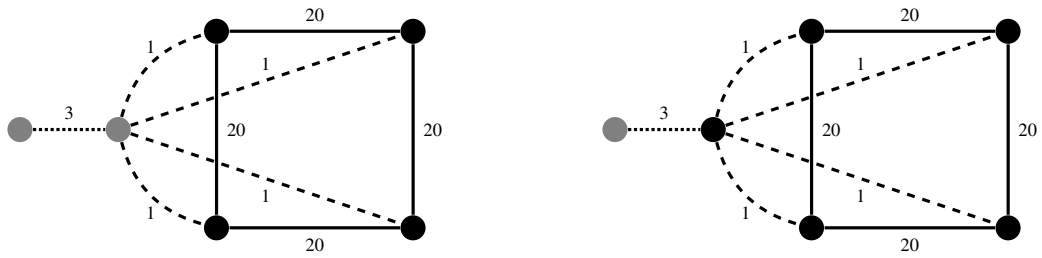

Fig. 2. An example of how a single node exchange can improve cuttability. Edges styles represent different weight values (dashed edges correspond to a weight value 1, dotted edges to a weight value 3, solid edges to a weight value 20). Multilevel algorithm does a clustering with communities as in the left picture; a single exchange of the second node (from the left), as represented by the second picture, can improve cuttability (from -4 to -3). The second node is overlapping between the two communities.

way to assign overlapping nodes to the correct community. In this paper, we have assessed the Sequential Clique Percolating (SCP) [25] algorithm, which is fast enough.

Another approach is RaRe and it is described in [23]. It consists in calculating, prior to clustering, a measure of centrality (for example, Rank or Betweenness) of nodes, removing them according to their centrality, computing the clustering on the reduced graph and then inserting the removed nodes back. This does not seem to be a good idea when the target objective function is modularity and the clustering algorithm is Multilevel, as we will see later in the section devoted to experiments.

Other approaches [20], [21], [22] are related to Non-Negative Matrix Factorization. Even for these techniques, a poor performance with respect to modularity was observed in practical experiments.

IV. Cuttability

In this section, we define a novel measure over edges that we use to detect overlapping nodes given a clustering of the network nodes. We call this measure *cuttability* since, when aggregating its values over all the edges entering a node, it returns a score that is high when the node is "in the middle" between two or more communities, i.e., the node can be cut away without affecting the coherence of the clusters.

Specifically, *cuttability* can be defined for each edge of an undirected weighted graph, provided that its nodes have been previously clustered. It takes into account the neighbourhoods of the two nodes i and j (of the edge). We can indeed define two ancillary functions:

- the sum of the weights of edges connected to q whose other node belongs to the same cluster of q:

$$this(q) = \sum_{(q,k)\in E, C(q)=C(k)} w_{q,k},$$

- the sum of the weights of edges connected to q whose other node belongs to the same cluster of r:

$$other(q,r) = \sum_{(q,k)\in E, C(r)=C(k)} w_{q,k},$$

which can be used to define the cuttability of an edge (i,j) as

$$cut(i,j) = \min\{this(i) - other(i,j), this(j) - other(j,i)\}.$$

Applying the previous definition[5], if the two nodes belong to the same cluster, then the cuttability is 0. The cuttability of the graph G can be defined as the sum of the cuttability of its edges:

$$cuttability(G) = \sum_{(i,j)\in E_G} cut(i,j).$$

When we work with directed weighted graphs, e.g. adopting the HoW or SA relations, cuttability can be calculated transforming the graph into an undirected one (i.e. making edges (i,j) and (j,i) to have the same weight). Although with this transformation information about directionality is lost, this does not prove harmful[6] to finding overlapping nodes, as they usually have strong in-out relationships between several communities.

There are some differences between good-cuttability clusterings and good-modularity clusterings. For example, considering the Multilevel algorithm there are cases where node exchanges between clusters can improve cuttability. This happens because even if it is true that the Multilevel algorithm can return a good placement of the node regarding to its neighbours, this placement is not always the most synthetic description of communities and their affiliation. Considering social networks, it may be more significant to describe relations among groups when there are few strong communicators that do intergroup relationship work, than to describe them with a myriad of weak relations between different people. The first scenario is exactly the one where cuttability is maximized.

The nodes where modularity and cuttability approaches differ are the overlapping ones. Indeed, they belong in a slender way to a community and different descriptions (of communities) are possible. So, the suggested method is about considering a local maximum of modularity (i.e. no change of a node's community can improve the modularity), obtained using the Multilevel algorithm, and trying to see whether it is a local maximum of cuttability. If a node's change of community can improve cuttability, then the node is considered to be an overlapping one (see Figure 2 for an example).

[5]Note that $this(q) = other(q,q)$, but in our opinion it's clearer to leave the definition as it is.

[6]A discussion on this topic can be found in [17].

Overlapping_nodes(G,C)
Require: A weighted graph G, a clustering C of nodes in G
Ensure: A set of overlapping nodes L
$L \leftarrow \emptyset$; ▷ Set of overlapping nodes, initially empty
for all $n \in V$ **do**
$N_n \leftarrow \{C(k) \mid (n,k) \in E\} \setminus \{C(n)\}$ ▷ Compute the set of different communities in the neighbourhood of n
if $N_n \neq \emptyset$ **then**
local_cut $\leftarrow 0$
for all $k \in$ *Neigborhood*(n) **do**
local_cut $\leftarrow$ *local_cut* + $\texttt{cut}((n,k), C(n))$
▷ $\texttt{cut}((u,v),c)$ computes cuttability of edge (u,v) using $C(u) = c$
end for
end if
for all $i \in N_n$ **do**
$\delta \leftarrow 0$;
for all $k \in$ *Neigborhood*(n) **do**
$\delta \leftarrow \delta + \texttt{cut}((n,k), i)$
end for
if $\delta >$ *local_cut* **then**
$L \leftarrow L \cup \{n\}$
skip for ▷ Terminate **for**
end if
end for
end for
return L

Fig. 3. Algorithm for discovering the set of overlapping nodes.

V. Finding overlapping nodes and improving modularity

The algorithm we propose for finding overlapping nodes is described in Figure 3. The algorithm requires in input a weighted graph and a clustering of its nodes. It returns a set of overlapping nodes, the set L, if any of them is present. To check if a node is overlapping, its actual community and a list of other adjacent communities are considered. In turn, the community of the considered node is set to one of the (different) adjacent community and cuttability is checked. If cuttability increases, the node is added to the set of overlapping nodes L. It is easy to see that the computational complexity of the algorithm is linear in the number of edges of the graph. The character of overlapping nodes can be quite different. Indeed, overlaps can exist between communities that are effectively different, or between communities that are similar but were somehow split by the clustering algorithm. For example, the Multilevel algorithm can erroneously get them split in order to try to get a clustering with higher modularity. To verify if a node belongs to the "first" category (i.e., it overlaps between two really distinct communities) the restricted graph, deprived of that single node, should have better modularity. Indeed, in that case, there are many intercommunity edges passing through the node which would be removed. The "first" category is really the one that matters as it describes real overlaps.

One could argue that, instead of introducing cuttability, it would have been better to evaluate directly whether nodes have strong connections between different communities. However, doing that would rely on a criteria to judge whether a node is or not between different communities, looking only to the strength of connections; cuttability does this work and, in addition, considers also the structure of the graph, as it seeks for the "most synthetic" description.

Improve_modularity(G,L)
Require: A weighted graph G, a set L of overlapping nodes of G
Ensure: A set of clusters for G
$C \leftarrow \texttt{clustering}(G \setminus L)$ ▷ We use the Multilevel Algorithm
Clusters $= \{C(k) | k \in G \setminus L\}$
$C_L \leftarrow \emptyset$ ▷ Set extending C to nodes in L, initially empty
for all $n \in L$ **do**
$L_n \leftarrow L \setminus \{n\}$ ▷ L without n
$G_n \leftarrow G \setminus L_n$ ▷ $G_n \subset G$ with nodes of $G \setminus L$ plus n
$c^* = \arg\max_{c \in \textit{Clusters}} \texttt{modu}(G_n, C \cup \{C(n) = c\})$
▷ arg max returns the smallest cluster index with highest modularity value for G_n
$C_L \leftarrow C_L \cup \{C(n) = c^*\}$
end for
return $C \cup C_L$

Fig. 4. Algorithm for improving modularity.

Iterated_cuttability(G)
Require: A weighted graph G
Ensure: A set of clusters for G
$C \leftarrow \texttt{clustering}(G)$ ▷ We use the Multilevel Algorithm
$m \leftarrow \texttt{modularity}(G, C)$
repeat
$L \leftarrow \texttt{Overlapping\_nodes}(G, C)$
$C_{new} \leftarrow \texttt{Improve\_modularity}(G, L)$
new_m $\leftarrow \texttt{modularity}(G, C_{new})$
if *new_m* $\geq m$ **then**
$C \leftarrow C_{new}$
$m \leftarrow$ *new_m*
end if
until *new_m* $> m$
return C

Fig. 5. The Iterated Cuttability algorithm.

Once we have the list of overlapping nodes, a way we could consider to improve modularity of the overall graph clustering is to initially remove overlapping nodes from the graph, then finding a clustering of the "reduced" graph, and finally trying to insert back overlapping nodes in the correct cluster in order to maximize modularity. We try to maximize modularity because it is the most widely accepted approach in judging the quality of graph clustering. The algorithm implementing this approach is described in Figure 4.

Finally, it can be observed that the two steps described above, i.e., discovering overlapping nodes and improving modularity, can be iterated, till no more improvement on modularity is observed. This iterated algorithm is described in Figure 5. It is easy to verify that such algorithm, soon or later, is going to converge (to a local maximum), since the number of different clusterings is finite and the second step (improving modularity) avoids the possibility of cycling among different clustering with (local) suboptimal modularity. We can't do worse than the Multilevel algorithm because, in

TABLE I. RESULTS FOR HANDOVER (OF WORK)-BASED SOCIAL NETWORKS, CALCULATED ON DAILY LOGS OF UBUNTU IRC NETWORK FOR THE FIRST THREE MONTHS OF 2005. THE NUMBER OF OVERLAPPING NODES (O.N.) IS REPORTED IN THE FOURTH COLUMN. MODULARITY VALUES FOR THE DIFFERENT CONSIDERED ALGORITHMS ARE REPORTED IN COLUMNS 5-9. THE NUMBER OF DISCOVERED COMMUNITIES (CO.) IS REPORTED IN PARENTHESIS AFTER THE MODULARITY VALUE. THE ACRONYM ON+IM REFERS TO THE PROPOSED APPROACH.

Dataset name	#Nodes	#Edges	#o.n.	Q_{SCP} (#co.)	Q_{BNMF} (#co.)	Q_{RaRe} (#co.)	$Q_{Multilevel}$ (#co.)	Q_{ON+IM} (#co.)
IRC_0301_HO	288	2067	14	0.0296 (6)	0.2314 (8)	0.5806 (30)	0.6154 (32)	**0.6238 (41)**
IRC_1501_HO	318	2474	16	0.0975 (11)	0.3576 (8)	0.6079 (32)	0.6272 (36)	**0.6601 (42)**
IRC_2001_HO	297	2311	13	0.1050 (6)	0.2890 (7)	0.6009 (24)	0.6101 (31)	**0.6269 (38)**
IRC_0402_HO	323	2767	23	-0.0200 (3)	0.2132 (5)	0.5012 (49)	0.5615 (36)	**0.5831 (49)**
IRC_0602_HO	338	2626	16	0.0608 (9)	0.2432 (13)	0.5800 (31)	0.6261 (37)	**0.6308 (49)**
IRC_1402_HO	385	3036	19	0.0035 (6)	0.2699 (9)	0.5993 (35)	0.6025 (44)	**0.6240 (52)**
IRC_1802_HO	383	3022	20	0.0056 (7)	0.3337 (8)	0.6057 (34)	0.6148 (41)	**0.6295 (59)**
IRC_2102_HO	402	3380	25	-0.0141 (5)	0.2375 (15)	0.5910 (39)	0.5980 (44)	**0.6059 (61)**
IRC_2503_HO	457	3662	29	0.0053 (4)	0.2715 (8)	0.5809 (29)	0.6119 (44)	**0.6153 (62)**
IRC_2703_HO	474	3495	49	0.0127 (7)	0.2749 (11)	0.6065 (32)	0.4330 (162)	**0.6329 (101)**

TABLE II. RESULTS FOR SIMILAR ACTIVITIES-BASED SOCIAL NETWORKS, CALCULATED ON DAILY LOGS OF UBUNTU IRC NETWORK FOR THE FIRST THREE MONTHS OF 2005. THE NUMBER OF OVERLAPPING NODES (O.N.) IS REPORTED IN THE FOURTH COLUMN. MODULARITY VALUES FOR THE DIFFERENT CONSIDERED ALGORITHMS ARE REPORTED IN COLUMNS 5-7. THE NUMBER OF DISCOVERED COMMUNITIES (CO.) IS REPORTED IN PARENTHESIS AFTER THE MODULARITY VALUE. THE ACRONYM ON+IM REFERS TO THE PROPOSED APPROACH.

Dataset name	#Nodes	#Edges	#o.n.	Q_{RaRe} (#co.)	$Q_{Multilevel}$ (#co.)	Q_{ON+IM} (#co.)
IRC_1301_SA	232	42660	3	0.0172 (13)	0.0227 (2)	**0.0229 (2)**
IRC_2001_SA	297	72826	3	0.0229 (16)	0.0273 (3)	**0.0275 (3)**
IRC_3001_SA	375	113106	1	0.0356 (21)	0.0442 (3)	**0.0444 (3)**
IRC_0402_SA	323	88530	8	0.0206 (18)	0.0260 (2)	**0.0262 (2)**
IRC_2802_SA	390	119832	5	0.0416 (26)	0.0455 (2)	**0.0457 (2)**
IRC_0403_SA	402	131216	7	0.0177 (22)	0.0267 (2)	**0.0269 (2)**
IRC_1003_SA	362	109400	2	0.0177 (20)	0.0196 (2)	**0.0197 (2)**
IRC_1103_SA	395	120944	4	0.0279 (21)	0.0339 (2)	**0.0341 (2)**
IRC_2503_SA	457	172182	6	0.0208 (24)	0.0248 (2)	**0.0250 (2)**
IRC_3103_SA	545	227156	9	0.0272 (30)	0.0444 (2)	**0.0446 (2)**

the worst case, we keep its modularity.

VI. EXPERIMENTAL ASSESSMENT

In order to assess the proposed approach, we have used datasets involving IRC (Internet Relay Chat)[7] and Enron email log [26] data. Moreover, for the IRC datasets, we have compared our basic approach, hereafter refereed to with the acronym ON+IM (Overlapping Nodes plus Improve Modularity), versus the following algorithms present in literature: SCP [25], BNMF [20], RaRe [23], and Multilevel [12]. Non-Negative Matrix Factorization-based methods (like BNMF) require to specify a number of basis, that corresponds to the number of communities which are looked for in the graph. In order to have a proper comparison versus our method, we have searched the number of basis maximising modularity within the interval $[1, \ldots, 50]$. The results obtained for IRC datasets by the iterated version of our approach are discussed in Section VII.

For the Enron email log dataset, we have just performed the discovery of overlapping nodes by using our Overlapping Nodes algorithm described in Figure 3.

A. Application to IRC logs

IRC (Internet Relay Chat) is a popular chat system working in a client-server way: the client could connect to the server and join some channels where he/she could have some chatting. IRC logs are collections of these discussions, usually recorded by bots (non-human IRC clients).

[7] http://irclogs.ubuntu.com/

TABLE III. NICKNAMES OF USERS CORRESPONDING TO TOP OVERLAPPING NODES IN IRC_2703_HO AFTER THE APPLICATION OF THE PROPOSED APPROACH. THEY CORRESPOND TO USERS INVOLVED INTO THE MOST DIVERSIFIED DISCUSSIONS.

fabbione	cef	prego	DarthFrog
hou5ton	thom	jordi	Hayden
mvo	thoreauputic	tyler_	chillywilly
Xappe	Myrtti	niran	Levander
sic	icebalm	garrut	streetbmx
NeverTheLess	ali	racingcamel	glguy
omniwork	mirco	[dEvIL-mAN]	kanga
OC_Doppelganger	Anubis	brbr	gilles
trans_err	Bwl	edulix	phas
neighborlee	tofu	Nomikos	Snarfy
Riddell	jazzka	pauldaoust	Carl
TPC	dazedlap	lagCisco	rob

Seemingly far from a business context, IRC logs taken by various IRC channels have a structure that is similar to an organization. There are indeed handovers (of discussions) between individuals and two users are considered to have "similar activities" if they chat with the same intensity in the same channels.

The logs we have considered are from the Ubuntu IRC Network, related to the famous Linux distribution. This "specialised" network aims at offering help and know-how to Linux users. The main feature of this IRC network is that users usually chat in a single specific channel so, roughly speaking, almost every channel is a "closed" group having its users.

Modularity results obtained for the Handover (of Work)-derived network are reported in Table I. Both SCP and BNMF returned very poor modularity values, so we decided not to consider them for the experiments reported in the following.

TABLE IV. OVERLAPPING IS NOT JUST ABOUT TWO COMMUNITIES: RICHARD SANDERS IS OVERLAPPING BETWEEN THESE 4 DIFFERENT COMMUNITIES (EVEN IF NOT EXPLICITLY WRITTEN IN ANY ONE BELOW). ONLY THE 7 MOST ACTIVE MEMBERS FOR EACH COMMUNITY ARE SHOWN.

Com.A	Com.B	Com.C	Com.D
tana.jones@enron.com	kay.mann@enron.com	steven.kean@enron.com	louise.kitchen@enron.com
sara.shackleton@enron.com	benjamin.rogers@enron.com	richard.shapiro@enron.com	sally.beck@enron.com
mark.taylor@enron.com	suzanne.adams@enron.com	jeff.dasovich@enron.com	john.lavorato@enron.com
susan.bailey@enron.com	ben.jacoby@enron.com	james.steffes@enron.com	greg.whalley@enron.com
mark.haedicke@enron.com	sheila.tweed@enron.com	susan.mara@enron.com	jeffrey.shankman@enron.com
elizabeth.sager@enron.com	kathleen.carnahan@enron.com	paul.kaufman@enron.com	mike.mcconnell@enron.com
carol.clair@enron.com	carlos.sole@enron.com	mark.palmer@enron.com	rick.buy@enron.com

We have chosen the Multilevel algorithm as the base clustering algorithm for the RaRe technique; nodes removal threshold for RaRe was chosen to be 5%; other thresholds also did not show an improvement. In Table II we have reported modularity results obtained for the Similar Activities-derived network. While for the first metric we have obtained a good number of overlapping nodes and the quality of clustering really improves, for the second metric we have obtained only marginal improvements. This, indeed, is related to the "closed groups" structure of Ubuntu IRC Network, so there is no much space for overlaps among communities.

Considering the Handover (of Work) metric, we can observe that for log IRC_2703_HO modularity improves from 0.4330 (initial clustering returned by the Multilevel algorithm) to 0.6329, with a gain of almost 50%. It is clear that overlapping nodes ruined the result obtained by the Multilevel algorithm, while our algorithm helped to improve the situation. In this case, also the RaRe approach helped to improve modularity, even if the result obtained is slightly inferior compared to our method.

Considering again IRC_2703_HO, it is interesting to identify which are the top overlapping nodes detected by our algorithm (see Table III). They correspond to users that are not the most active in the network, according to the number of lines in the log, but the ones having the greatest number of discussions with different people: among the top detected overlapping nodes, we find the user *fabbione*, which is the Team Leader of Ubuntu Server, and *hou5ton*, which is an active "helper"[8]. Thus, our algorithm finds overlapping nodes which are meaningful in role and importance inside the relation map.

B. Application to Enron email log

The Enron email log [26] is a large log (it contains over 600000 emails) that was obtained and then released by a CS researcher after the crack of Enron company. It is particular interesting because reports email communications among top management. A social network can be built upon it with the use of "frequency of communication" metric: two individuals are connected by an edge if they exchange emails. Moreover, the value of the weight associated to each edge is proportional to the amount of communications occurred between them. We were expecting to find several communities inside Enron's email network, each one corresponding to different roles inside the company, as well as several overlapping nodes among the coordinators of these communities, as they effectively belong to more than one community.

[8]He has helped people to use Ubuntu IRC Network.

TABLE V. SOME OVERLAPPING NODES IN ENRON EMAIL NETWORK (BASED ON FREQUENCY OF COMMUNICATIONS). WE SEE THAT PART OF TOP MANAGEMENT (THE ONES WHICH HAVE A LARGE NUMBER OF DIFFERENT CONTACTS) APPEARS HERE.

Address	# received emails	Role
pete.davis@enron.com	very high (>5000)	Community organizer
richard.sanders@enron.com	high (>500)	Assistant general counsel
brent.hendry@enron.com	high (>500)	Seniour counsel
susan.scott@enron.com	high (>500)	Communications expert
robert.badeer@enron.com	high (>500)	Trader
bryan.hull@enron.com	high (>500)	Analyst
brian.redmond@enron.com	high (>500)	Managing Director
kimberly.hillis@enron.com	high (>500)	Executive Assistant
paula.rieker@enron.com	high (>500)	No.2 Executive
don.black@enron.com	high (>500)	Vice president
jonathan.mckay@enron.com	high (>500)	Vice president

Applying our algorithm, we find several overlapping nodes, and the majority of them cover important roles inside the Enron company (see Table V). Discovered individuals cover roles from senior counseling (Richard Sanders and Brent Hendry) to high management (Paula Rieker, Don Black, Jonathan Mick), communications and community experts (Pete Davis and Susan Scott) and financial (Robert Badeer and Bryan Hull). Trying to analyse the "network" behind a given person, it results that these overlapping nodes effectively belong to at least two different communities. Richard Sanders, (see Table IV), for example, did belong to four different communities. Thus, our algorithm succeeded in finding nodes which, given their role, effectively overlap.

Concerning modularity, we observe only limited improvements: the initial value was 0.6495, while the value after the application of our algorithm is 0.6542. We think this is mainly due to the fact that the Multilevel algorithm was not hampered so much by overlapping nodes, being thus able to reach a quite good modularity value from the beginning.

VII. DISCUSSION

A. Why in some logs are there only small improvements in modularity?

In the previous section, we have seen that, for some logs, our approach does get only a small improvement in modularity, even if there were a decent number of overlapping nodes. One could think that the clustering would have been more heavily hit by the presence of overlaps among communities. Why in most cases the Multilevel algorithms can get already a good modularity value?

This was explained in [27], and it is related to the particular topology of communities near the optimum modularity value.

Indeed, in [27] it is shown that there are many clusterings whose modularity is near to the "global maximum", and it is eventually easy for clustering algorithms which try to maximize modularity, like the Multilevel one, to get a good modularity value[9]. Thus, even if overlapping nodes occur, it is relatively easy for the Multilevel algorithm to reach a good clustering. Because of that, what we think our algorithm does is to get the clustering further nearer to the global maximum of modularity.

B. Where are the biggest improvements?

Considering again the IRC_2703_HO log, we can ask why there was such a big improvement in modularity. Let us start with few observations:

- IRC_2703_HO's clustering by the Multilevel algorithm exhibits 162 different communities, among 474 nodes. So, the detected communities are really small, and we can consider them micro-communities.
- For the same log, our algorithm detects 101 different communities and 49 overlapping nodes (a large quantity of overlapping nodes).

The fact that the Multilevel algorithm finds a clustering with a low modularity is mainly due to the modularity resolution problems, which are described in [18]. Roughly speaking, modularity maximization approaches (like the Multilevel algorithm) suffer, and in many cases fails, to find the actual community structure when the communities are small; those are eventually merged by the Multilevel algorithm into bigger ones, which fail to catch the complexity of the community structure, while getting to a "local maximum" of modularity. In IRC_2703_HO, the Multilevel algorithm was not somehow hampered by resolution problems, managing to find micro-communities. These, when found by the clustering algorithm, are more likely (see always [18]) to represent the actual community structure. The discovery of micro-communities, however, is paid for by a low value of modularity.

Our approach is still able to discover many micro-communities (101) while obtaining a good modularity value. Moreover, the micro-communities detected by our approach seem to be well defined, as shown by the large number of overlapping nodes, which is what is expected when many small communities are present and interact each other.

C. What happens to overlapping nodes?

After detecting overlapping nodes and trying to improve the community structure, one may ask if, applying the algorithm to the new clustering, the overlapping nodes found in the previous phase are "confirmed" as overlapping nodes (confirming that, even if the community description got improved, they are always between distinct communities) or instead "disappear".

In the considered applications, we have found three behaviours (see Table VI):

- Nodes that are overlapping both considering the old and the new clustering. These "persisting" overlapping nodes are clearly between at least two distinct communities, which are confirmed to be clearly distinct also considering the new clustering.
- Nodes that are overlapping considering the old clustering, but are not considered as overlapping in the new clustering. This happens because of the improvement of the community structure, which makes these nodes belonging more clearly to a community.
- Nodes that are overlapping in the new clustering, but were not overlapping in the old one. Due to the change of the community structure, it may happen that nodes get slender belonging to a community and, so, applying our algorithm to the new clustering, are detected as overlapping.

[9]A "local maximum" which is near to the "global maximum".

TABLE VI. NUMBER OF OVERLAPPING NODES FOUND BY APPLYING OUR METHOD (TO FIND OVERLAPPING NODES) TO THE INITIAL CLUSTERING AND TO THE CLUSTERING OBTAINED AFTER ONE APPLICATION OF THE ALGORITHM (TO IMPROVE MODULARITY).

	# overlapping nodes (o.n.)			% o.n.
Dataset Name	**start**	**1st iter.**	**shared**	**shared**
IRC_0301_HO	14	8	4	50,00 %
IRC_1501_HO	16	7	4	57,14 %
IRC_2001_HO	13	7	5	71,43 %
IRC_0402_HO	23	13	9	69,23 %
IRC_0602_HO	16	12	7	58,33 %
IRC_1402_HO	19	8	4	50,00 %
IRC_1802_HO	20	19	10	52,63 %
IRC_2102_HO	25	19	12	63,16 %
IRC_2503_HO	29	25	14	56,00 %
IRC_2703_HO	49	38	17	44,74 %

TABLE VII. APPLICATION OF OUT ITERATED ALGORITHM. IN SOME CASES, THE COMMUNITY STRUCTURE CAN BE FURTHER IMPROVED.

	# iter.	Modularity		
Dataset Name	**ON+IM**	**Multilevel**	**ON+IM 1st iter.**	**ON+IM final**
IRC_0301_HO	1	0.6154	0.6238	0.6238
IRC_1501_HO	1	0.6272	0.6601	0.6601
IRC_2001_HO	**3**	0.6101	0.6269	**0.6599**
IRC_0402_HO	1	0.5615	0.5831	0.5831
IRC_0602_HO	1	0.6261	0.6308	0.6308
IRC_1402_HO	1	0.6025	0.6240	0.6240
IRC_1802_HO	**2**	0.6148	0.6295	**0.6628**
IRC_2102_HO	1	0.5980	0.6059	0.6059
IRC_2503_HO	1	0.6119	0.6153	0.6153
IRC_2703_HO	1	0.4330	0.6329	0.6329

We see, in Table VI, that the first category is, as expected, the most numerous one, followed by the second.

Given the results obtained in Table VI, we applied to the IRC datasets the iterated algorithm defined in Figure 5. While in most of the considered datasets the iterated algorithm terminated after 1 iteration, there are cases when using it a clear improvement of the community structure is observed (see Table VII). Considering IRC_2001_HO, in the third iteration we find 5 overlapping nodes: 4 of them are "persisting" in comparison to the second iteration and 1 is a newly detected overlapping node, confirming the prevalence of "persisting" overlapping nodes.

VIII. Conclusions

In this paper, we have proposed a new concept, i.e., *cut-tability*, to detect overlapping nodes in social networks. Overlapping nodes are particularly interesting in business contexts because of their inter-community role. Correctly identifying overlapping nodes, indeed, is a key to get an improvement in detected community structure, and so an improvement on the information top management can get from "professional relation" networks, such as the one derived by Handover of Work or Similar Activities. While the improvements in modularity are not large, the proposed method seems more consistent (see Table I) than other existing approaches. Moreover, detected overlapping nodes are shown to have really an inter-community role (see Table IV). The proposed method seems to produce the biggest improvements in modularity when there is a micro-communities structure (so, there are many very small communities) as the correct assignment of the single node becomes truly important.

IX. Acknowledgements

This work has been supported by FSE fellowship 2105/201/17/1148/2013.

References

[1] M. Kilduff and D. J. Brass, "Organizational social network research: Core ideas and key debates," *The Academy of Management Annals*, vol. 4, no. 1, pp. 317–357, 2010.

[2] M. Kilduff and W. Tsai, *Social networks and organizations*. Sage, 2003.

[3] W. M. P. van der Aalst, H. A. Reijers, and M. Song, "Discovering social networks from event logs," *Proceedings of Computer Supported Cooperative Work (CSCW)*, vol. 14, no. 6, pp. 549–593, 2005.

[4] M. Fire, R. Puzis, and Y. Elovici, "Organization mining using online social networks," *CoRR*, vol. abs/1303.3741, 2013.

[5] A. Culotta, R. Bekkerman, and A. Mccallum, "Extracting social networks and contact information from email and the web," in *In Proceedings of CEAS-1*, 2004.

[6] S. Farnham, W. Portnoy, and A. Turski, "Using email mailing lists to approximate and explore corporate social networks," *Proceedings of Computer Supported Cooperative Work (CSCW)*, vol. 4, 2004.

[7] D. Fisher and P. Dourish, "Social and temporal structures in everyday collaboration," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2004, pp. 551–558.

[8] W. M. P. van der Aalst, T. A. J. M. M. Weijters, and L. Maruster, "Workflow mining: Discovering process models from event logs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, 2004.

[9] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, 2004.

[10] G. Palla, I. Derényi, 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, 2005.

[11] A. Lancichinetti, S. Fortunato, and J. Kertsz, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, 2009.

[12] 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, 2008.

[13] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, no. 3, 2007.

[14] A. Geyer-Schulz, "An ensemble learning strategy for graph clustering." *Graph Partitioning and Graph Clustering*, vol. 588, 2012.

[15] S. Liu, Q. Kang, J. An, and M. Zhou, "A weight-incorporated similarity-based clustering ensemble method," in *Networking, Sensing and Control (ICNSC), 2014 IEEE 11th International Conference on*. IEEE, 2014, pp. 719–724.

[16] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*. MIT Press, 2001, pp. 849–856.

[17] F. D. Malliaros and M. Vazirgiannis, "Clustering and community detection in directed networks: A survey," *Physics Reports*, vol. 533, no. 4, pp. 95–142, 2013.

[18] A. Lancichinetti and S. Fortunato, "Limits of modularity maximization in community detection," *Physical Review E*, vol. 84, 2011.

[19] Q. Kang, K. Wang, B. Huang, and J. An, "Kernel optimisation for KPCA based on gaussianity estimation," *IJBIC*, vol. 6, no. 2, pp. 91–107, 2014. [Online]. Available: http://dx.doi.org/10.1504/IJBIC.2014.060620

[20] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon, "Overlapping community detection using bayesian non-negative matrix factorization," *Physical Review E*, vol. 83, 2011.

[21] Y. Zhang and D.-Y. Yeung, "Overlapping community detection via bounded nonnegative matrix tri-factorization," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2012, pp. 606–614.

[22] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*. New York, NY, USA: ACM, 2013, pp. 587–596.

[23] J. Baumes, M. Goldberg, and M. Magdon-ismail, "Efficient identification of overlapping communities," in *In IEEE International Conference on Intelligence and Security Informatics (ISI*, 2005, pp. 27–36.

[24] A. McCallum, X. Wang, and A. Corrada-Emmanuel, "Topic and role discovery in social networks with experiments on enron and academic email," *Journal of Artificial Intelligence Research*, vol. 30, no. 1, pp. 249–272, 2007.

[25] J. A. Almendral, I. Leyva, D. Li, I. Sendiña˜Nadal, S. Havlin, and S. Boccaletti, "Dynamics of overlapping structures in modular networks," *Physical Review E*, vol. 82, 2010.

[26] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *Machine Learning: ECML 2004*. Springer Berlin Heidelberg, 2004, vol. 3201, pp. 217–226.

[27] B. Good, Y. D. Montjoye, and A. Clauset, "Performance of modularity maximization in practical contexts," *Physical Review E*, vol. 81, no. 4, 2010.