Overlapping Kernel-based Community Detection with Node Attributes

Daniele Maccagnola, Elisabetta Fersini, Rabah Djennadi and Enza Messina
DISCo, University of Milano-Bicocca, Viale Sarca 336, 20126, Milan, Italy

Keywords: Community Detection, Social Network Analysis, Kernel Communities.

Abstract: Community Detection is a fundamental task in the field of Social Network Analysis, extensively studied in literature. Recently, some approaches have been proposed to detect communities distinguishing their members between kernel that represents opinion leaders, and auxiliary who are not leaders but are linked to them. However, these approaches suffer from two important limitations: first, they cannot identify overlapping communities, which are often found in social networks (users are likely to belong to multiple groups simultaneously); second, they cannot deal with node attributes, which can provide important information related to community affiliation. In this paper we propose a method to improve a well-known kernel-based approach named Greedy-WeBA (Wang et al., 2011) and overcome these limitations. We perform a comparative analysis on three social network datasets, Wikipedia, Twitter and Facebook, showing that modeling overlapping communities and considering node attributes strongly improves the ability of detecting real social network communities.

1 INTRODUCTION

Community detection is an important task that allows to discover the structure and organization of online social networks. The problem of community detection (also called community discovery) has been largely investigated. Several algorithms have been proposed, ranging from cut- and conductance-based methods (Rosvall and Bergstrom, 2007), agglomerative-based (Newman, 2006b), model-based (Chang and Blei, 2009) and spectral clustering (Donetti and Munoz, 2004).

However, most of these methodologies do not consider that community structures of influential users (opinion leaders) are different from that of others. It has been shown in the literature that in many social network, especially online social networks such as Twitter, Facebook and Google Plus, the average degree of connections of opinion leaders is almost ten times more than other users (Wang et al., 2011).

Most of the approaches for community and opinion leader detection available in literature are based on the assumption that each influential user should be placed in a different community with its relative followers/friends. However, this assumption does not reflect the real world, where a community is likely to be composed of several kernels of users (as opinion leaders) and auxiliary members.

In order to define a community and detect its opinion leaders, the *community kernel detection* problem has been introduced in (Wang et al., 2011), composed of two subtasks: (1) the identification of kernel nodes, i.e. influential members of the network and (2) the identification of auxiliary nodes (non-influential members) and their association to a kernel to form a community.

In literature, very few approaches have been proposed to address this problem (Wang et al., 2011; Du et al., 2007). Among these, one of the most promising is the Greedy - Weight-balanced Community detection algorithm (Greedy-WeBA), which combines multiple steps to first identify the kernels, and subsequently the auxiliary nodes to form the communities.

However, this approach suffers from two important limitations:

- **Overlapping Communities.** Most actual social networks are made of highly overlapping cohesive subgroups of nodes, simply because individuals often belong to numerous different kinds of communities simultaneously (Leskovec and Mcauley, 2012). Members of a network may participate in many social circles according to their interests, hobbies, and relationships connected to their educational background, working environment and family.

  Greedy-WeBA does not take into account the possibility of overlapping communities when detect-

Maccagnola, D., Fersini, E., Djennadi, R. and Messina, E..
Overlapping Kernel-based Community Detection with Node Attributes.
In *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015) - Volume 1: KDIR*, pages 517-524
ISBN: 978-989-758-158-8

ing the auxiliary nodes, that can be associated only to one kernel (each of them is assigned only to the most similar kernel).

For this reason, we introduce a *Overlapping Auxiliary Community Detection* approach that can overcome the limits of the existing method.

- **Node Attributes** Existing approaches for community detection usually take into account only one source of information: the relationships among the network members, e.g. friendship or following/followee relationships.

  Social networks, however, often provide a large amount of information that is not directly included in the relationships. For example, online social networks like Twitter and Facebook allow their members to write and share textual messages (posts), which can be very informative attributes of the user representing interests and ideas.

  Still, most community detection algorithms do not exploit this information to improve their performance. The Greedy-WeBA algorithm is based on the assumption that each member of a kernel has more connections to/from the kernel than a vertex outside the kernel does. However, this assumption does not consider that two users may share similar interests even when not directly connected by a relationship.

  Therefore, we introduce an improved version of the Greedy-WeBA algorithm that includes both network structure and information from node attributes.

The paper is structured as follows: first, in Sec. 2 we summarize the existing related work, and in Sec. 3 we introduce some preliminary notation to better define the problem of kernel and community detection. Then, in Sec. 4 we present the proposed kernel community detection algorithm, highlighting the novel approaches we adopt to overcome the existing method's limitations. In Sec. 5 we outline the experimental investigation, detailing the datasets that will be used in this work, and in Sec. 6 we show the comparative results of the proposed approach with the baseline. Finally, in Sec. 7 conclusions are derived.

2 RELATED WORK

The problem of identifying and evaluating community has been addressed extensively by many papers (Papadopoulos et al., 2012). Most existing works are based on the hypothesis that communities are subsets of vertices which are densely connected internally, but sparsely connected to the rest of the network (Newman, 2004b; Newman, 2006a; Leskovec et al., 2008).

One of the most popular approaches is the algorithm developed by Girvan and Newman, which looks for disjoint communities in the social network based on a measure of betweenness and modularity (Newman, 2004a). Other works have also introduced information-theoretic frameworks for obtaining hierarchical communities in the networks (Rosvall and Bergstrom, 2007; Papadimitriou et al., 2008).

More recently, new methods have been proposed to detect communities that can overlap, and thus better represent the actual behavior in social networks. Mishra et al. (Mishra et al., 2008) proposed an algorithm based on the concept of (α, β) communities to allow close communities to overlap. Other methods allow users to belong to multiple communities, using either probabilistic generative processes (Yang and Leskovec, 2013) or using graph transformation approaches (Xie and Szymanski, 2012).

Considering that the above mentioned investigations do not consider node attributes when communities are created, some alternative methods have been proposed(Günnemann et al., 2013; Gunnemann et al., 2010; Chang and Blei, 2009; Liu et al., 2009; Yang et al., 2013). However, none of these approaches consider communities as composed of kernel members and auxiliary nodes, disregarding the real social network aggregation. In order to overcome this limitation, we extended one of the most recent and promising kernel-based approaches (Wang et al., 2011) to detect overlapping communities, also exploiting the information provided by node attributes as well as the network structure. To the best of our knowledge, no community detection algorithm with these characteristics has been proposed in literature.

3 PRELIMINARIES

Before discussing the details of the proposed method, we introduce some important notations. A social network is represented as a graph $G = (V, E)$, where the set of nodes V represents members of the network (users) and the set of edges E denotes connections among them.

Community detection in networks aims at finding a set of communities $C = \{c_1, c_2, \ldots, c_k\}$, where communities c_i are formed by groups of vertices with dense intra-community connections, but sparse inter-community links. Here we consider simple graphs only, i.e. graphs without self-loops or multi-edges.

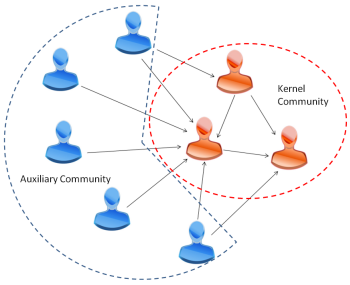

Figure 1: Example of kernel community (red members) and auxiliary community (blue members).

3.1 Kernel Communities

In this paper, communities are assumed to be composed by a kernel and an auxiliary community (see Fig.1 for example). They are defined as follows:

Def: Kernel Community. Given an oriented graph $G = (V,E)$, l disjoint subsets $\{K_1,\ldots,K_l\}$ of vertices are called kernel communities if:

$$|E(u,K_i)| \geq |E(v,K_i)| \wedge |E(K_i,u)| \geq |E(K_i,v)|, \quad \forall i \in \{1,\ldots,l\}, \forall u \in K_i, \forall v \notin K_i \quad (1)$$

where $E(A,B) = \{(u,v) \in E | u \in A, v \in B\}$ for $A,B \subseteq V$.

Def: Auxiliary Community. Given a set of kernel communities K, l associated subsets $\{A_{K_i},\ldots,A_{K_l}\}$ of vertices are called auxiliary communities if:

- $A_{K_i} \cap K_i = \varnothing, \forall i \in \{1,\ldots,l\}$;
- $|E(v,K_i)| \geq |E(v,K_j)|$, $\forall i \in \{1,\ldots,l\}, \forall j \neq i, \forall v \in A_{K_i}$;
- $|E(A_{K_i},K_i)| \geq |E(K_i,K_i)|, \forall i \in \{1,\ldots,l\}$.

For any $i \in \{i,\ldots,l\}$, each vertex in K_i is a kernel member and each vertex in A_{K_i} is an auxiliary member.

3.2 Node Attributes

In this paper we consider node attributes as additional information for detecting communities in networks. In order to model this information, we introduce a function $\tau(u) : u \rightarrow t^u$ which maps a network user $u \in V$ to its feature vector representation t^u as:

$$t^u = \left(t^u_1, t^u_2, \ldots, t^u_{|F|}\right) \quad (2)$$

where $|F|$ is the number of *attributes* shared by all the users. In our case, attributes can represent any kind of information related to the user (gender, age, job titles, etc.), denoted by binary values.

For any $u,v \in V$ represented as in Eq. 2, we derive a similarity matrix M, with $|M| = |V| \times |V|$, defined as follows:

$$M_{u,v} = \cos(t^u,t^v) = \frac{< t^u \cdot t^v >}{\|t^u\| \cdot \|t^v\|} \quad (3)$$

4 ALGORITHM

In order to overcome the limitations of the approaches reported in Sec. 2, we propose an extended and revised version of the kernel-based community detection algorithm WEBA, presented in (Wang et al., 2011).

This baseline algorithm consists of three main steps:

- A Greedy approach based on maximum cardinality search, aimed at finding l kernels nodes for each community with dense internal connections allowing also dense external relations;
- A Weight-balanced heuristic (WeBA) to tune the solution find by Greedy in order to revise the initial community of kernels taking into account information provided by the connection of non-kernel members;
- An Auxiliary Community Detection approach to find the auxiliary communities: it associates at each node a ranked list of kernels (kernel-based association).

In the following we detail the novel methods proposed in this paper: first, we describe the new Greedy and WeBA algorithm for exploiting node attributes in the detection of kernel communities; then, we introduce a variant of the Auxiliary Community Detection method that can detect overlapping communities.

4.1 Community Detection with Node Attributes

A major limit of the existing algorithm is its inability to take into account all the sources of information available in the networks. Specifically, node attributes can be considered to improve the performance of the community detection task.

In order to improve the original algorithm shown in (Wang et al., 2011), we separately modify the procedures for *Greedy* and *WeBA* as following:

Greedy. Given an undirected graph $G = (V,E)$ and kernel size k, initialize a subset $S \subseteq V$ to be a random vertex $v \in V$. Then, iteratively enlarge S by adding the vertex with the maximum number of connections to S. If there are multiple vertices with the

Algorithm 1 Revised Greedy

Input: $G = (V, E)$, similarity matrix M, kernel size k
Output: Community Kernels $K = \{k_1, k_2, \ldots, k_l\}$
$Y = V$
$K \leftarrow \varnothing$
repeat
  $S \leftarrow$ a random vertex $v \in V$
  while $|S| \leq l$ **do**
    $R^* = \left\{ \underset{u \notin S}{\text{argmax}} \left(\frac{|E(u,S)| + \sum_{t=1}^{|S|} M(u,t)}{2} \right) \right\}$
    if $|R^* = 1|$ **then**
      $S \leftarrow S \cup R^*$
    else
      $U^* = \left\{ u \in R^* | \underset{u \notin S}{\text{argmax}}\ p(u) \right\}$
      if $|U^* = 1|$ **then**
        $S \leftarrow S \cup U^*$
      else
        $S \leftarrow S \cup$ random $u \in U^*$
  if $S \notin K$ **then**
    $K \leftarrow K, S$
  $Y = Y - S$
until $Y \neq \varnothing$
return K

Figure 2: Pseudocode for the revised Greedy algorithm.

maximum number of connections to S, pick the one with the highest degree $d(u) = \sum_{v \in V} E(u,v)$ (if there are several nodes with the same highest degree, randomly pick one of them). This subroutine is repeatedly executed $O(|V|/k)$ times to obtain steady-state results and reduce the effect of the random selection of the initial point.

This original *Greedy* algorithm has been extended in order to take into account content similarity of nodes. The proposed algorithm takes as additional input the similarity matrix M (defined in Eq. 3), to evaluate how close are the attributes of each couple of nodes. When the algorithm selects a vertex u as kernel node, it will evaluate not only the number of edges $d(u)$, but also the similarity of contents among u and all the other kernel members already assigned to the same kernel community. In particular, instead of evaluating only the degree $d(u)$ as indication of node importance, we define $p(u)$ as:

$$p(u) = \sum_{v \in V} \frac{E(u,v) + M(u,v)}{2} \tag{4}$$

The pseudo-code is reported in Fig.2.

WeBA. Starting from the initial result generated by the *Greedy* algorithm, the kernels are refined and optimized by the *Weighted-Balanced* Algorithm (WEBA). Given a kernel size l and an initial subset S to refine, the original WeBA algorithms assigns a weight $w(v) = 1$ to each vertex $v \in S$, and a weight $w(v) = 0$ to each vertex $v \notin S$. Let $N(v)$ be the set of

Algorithm 2 Revised WeBA

Input: $G = (V, E)$, similarity matrix M, kernel size k
Output: Community Kernels $K = \{k_1, k_2, \ldots, k_l\}$
$K \leftarrow \varnothing$
$K_g \leftarrow \text{GREEDY}(G, M, l)$
for $S \in K_g$ **do**
  $\forall v \in S, \quad w(v) \leftarrow 1$
  $\forall v \notin S, \quad w(v) \leftarrow 0$
  while $\exists u, v \in V$ *satisfying the relaxation conditions a),b),c)* **do**
    $\alpha \leftarrow (1 - w(u)) \frac{E(u,v) + M(u,v)}{2}$
    $\beta \leftarrow w(v) \frac{E(u,v) + M(u,v)}{2}$
    $\gamma \leftarrow \frac{\text{nw}^*(u) + \text{nw}^*(v)}{2}$
    $\delta \leftarrow \min(\alpha, \beta, \gamma)$
    pick the pair u, v with the maximum δ value
    $w(u) = \min(w(u) + \delta, 1)$
    $w(u) = \max(w(v) - \delta, 0)$
  $C \leftarrow \{v \in V | w(v) = 1\}$
  if $C \notin K$ **then**
    $K \leftarrow \{K, C\}$
return K

Figure 3: Pseudocode for the revised WeBA algorithm.

neighboring vertices of v, i.e. $N(v) = \{u \in V | (u,v) \in E\}$. Then, at each iteration, the algorithm searches for a pair of vertices $u, v \in V$ satisfying both of the following relaxation conditions:

a) $w(u) < 1$

b) $w(v) > 0$

c) $nw(u) > nw(v)$

where $nw(u)$ is the neighboring weight of u, i.e. $nw(u) = \sum_{v \in N(u)} w(v) \cdot E(u,v)$.

Similarly to Greedy, also WeBA has been extended in order to deal with the content similarity. In order to include it, we consider the neighboring weight according to both links and content similarity:

$$nw^*(u) = \sum_{v \in N(u)} w(v) \cdot \frac{E(u,v) + M(u,v)}{2} \tag{5}$$

The pseudocode for the revised WeBA is reported in Fig. 3.

4.2 Overlapping Auxiliary Communities

The detection of auxiliary communities has been revised and improved to allow auxiliary communities to overlap. Given a node v, the proposed approach takes into account a *popularity measure* relative to v when choosing the auxiliary community A_{K_i}. In particular, v is associated to A_{K_i} if two conditions are satisfied:

Algorithm 3 Revised Auxiliary Community

Input: Community Kernels $K = \{k_1, k_2, \ldots, k_l\}$
Output: Auxiliary Communities $A = \{A_{k_1}, A_{k_2}, \ldots, A_{k_l}\}$
$\forall i \in \{1, \ldots, l\}, \quad A_{k_i} \leftarrow \varnothing$
repeat
  $C_i = \bigcup\{k_i, A_{k_i}\}, \quad \forall i \in \{1, \ldots, l\}$
  for $i \leftarrow 1$ **to** l **do**
    $S \leftarrow \{v \in C_i | \forall u \in \bigcup C_i, \forall j \in \{1, \ldots, l\},$
    (1) $|E(v, C_i)| \geq |E(v, C_j)| > 0$
    (2) $\sum_{n=1}^{l} |E(v, C_n)| \geq \sum_{n=1}^{l} |E(u, C_n)|\}$
    $A_{k_i} \cup S$
until *no more vertices can be added*
return A

Figure 4: Pseudocode for the revised Auxiliary Community detection algorithm.

- v is the node with the highest number of edges pointing to the community $C_i = \bigcup\{K_i, A_{K_i}\}$, i.e.

$$|E(v, C_i)| \geq |E(v, C_j)| \quad \text{for } j \neq i \tag{6}$$

- There is no other node $u \notin C_i$ such that u has more edges pointing to all the communities C_n than v, i.e.

$$\sum_{n=1}^{k} |E(v, C_n)| \geq \sum_{n=1}^{k} |E(u, C_n)| \tag{7}$$

While the first condition was included in the original version of the algorithm, the second one ensures that we consider first the nodes having a higher number of connections (as indication of popularity) to all the communities.

If both conditions are satisfied for more than one community C_i, the node is associated to all of the corresponding A_{K_i}.

In Fig. 4 we report the pseudocode for the algorithm.

The final communities C_i will be formed by the association of the kernel community K_i with the corresponding auxiliary community A_{K_i}.

5 EXPERIMENTAL SETTINGS

Datasets Description. In order to evaluate the performance of the proposed kernel-based community detection method, we considered three benchmarks used in the state of the art:

- **Philosophers.** The philosophers network (Ahn et al., 2010) consists of Wikipedia articles about famous philosophers. Nodes represent Wikipedia articles about philosophers, and directed edges indicate whether one article links to another. The attributes of a given node u are represented by a binary indicator vector of out-links from node u to other non-philosopher Wikipedia articles (e.g. if a philosopher page links to a Wikipedia article "Mathematician", the binary value of the attribute "Mathematician" for the corresponding philosopher will be equal to one). The Wikipedia network is formed by 1546 nodes and 7971 edges.

  Moreover, Wikipedia provides categories (e.g. "Hindu philosophers", or "Austrian psychologists") for each article. We consider each category with more than five philosophers as a ground-truth community, obtaining a total of 907 overlapping communities.

- **Twitter.** The Twitter network is a ego-network available from the Stanford Large Network Dataset Collection (http://snap.stanford.edu/data) (Leskovec and Mcauley, 2012). The ground truth communities are obtained from Twitter "lists" manually labeled by the owner of the ego-network (only a subset of the nodes will belong to a community). Node attributes are defined by processing the tweets (posts) generated by each user of the network. We use a "bag of words" representation, where each binary attribute indicates that a specific word appeared in the user's tweets. In particular, we consider only specific words called "hashtags", i.e. words appearing in the tweets preceded by the character "#". The network contains a total of 125120 nodes and 2,248,406 edges, and a total of 3,140 communities.

- **Facebook.** Like the Twitter network, the Facebook network is composed of ego-networks from the Stanford Large Network Dataset Collection (Leskovec and Mcauley, 2012). Node attributes are extracted from user profiles, such as gender, job titles, institutions, etc. Ground truth communities have been manually labeled by the owner of the ego-network, and represent his "social circles". The size of the full network is 4089 nodes and 170174 edges, with 193 communities.

Table 1: Datasets statistics. N: number of nodes, E: number of edges, C: number of communities, K: number of node attributes, S: average community size, A: community membership per node.

Dataset	N	E	C	K	S	A
Philosophers	1546	7971	907	5770	6,86	6,87
Twitter	125120	2248406	3140	33569	15,54	0,39
Facebook	4089	170174	193	175	28,76	1,36

The statistics related to the benchmarks are reported in Table 1.

Baseline for Comparison. In order to investigate whether overlapping communities and node attributes

can aid the community detection task, we perform a comparative analysis with the following algorithms:

- **Standard Greedy-WeBA Algorithm.** We first test the performance of the original algorithm, without node attributes and with non-overlapping auxiliary community detection.
- **Overlapping Greedy-WeBA.** The second algorithm is the original version of Greedy-WeBA, but with the addition of our algorithm for detecting overlapping auxiliary communities.
- **Overlapping Greedy-WeBA with Node Attributes.** Finally, we test the complete version of our method, considering both overlapping communities and the availability of node attributes.

Evaluation Metrics We quantify the performance in terms of the agreement between the ground-truth communities and the communities detected by the algorithms. As some datasets contain nodes not belonging to any community, we do not include them when computing the performance. To compare a set of ground truth communities C^* to a set of detected communities C, we use the following measures: Precision (P), Recall (R) and F-Measure (F) (Eq. 8-10), which evaluate the number of correct pairs of vertices clustered into the same community kernel.

$$P(C_i, C_j^*) = \frac{\left|C_j^* \cap C_i\right|}{|C_i|} \tag{8}$$

$$R(C_i, C_j^*) = \frac{\left|C_j^* \cap C_i\right|}{\left|C_j^*\right|} \tag{9}$$

$$F(C_i, C_j^*) = \frac{2 \times P(C_i, C_j^*) \times R(C_i, C_j^*)}{P(C_i, C_j^*) + R(C_i, C_j^*)} \tag{10}$$

Moreover, we consider Jaccard Index (J) to measure the pairwise resemblance of C with C^* (Eq. 11).

$$J(C_i, C_j^*) = \frac{\left|C_j^* \cap C_i\right|}{\left|C_j^* \cup C_i\right|} \tag{11}$$

Finally, we introduce an index, based on the Jaccard measure, that evaluates the percentage of ground truth communities that have been successfully associated to the generated communities. This measure, called Equivalence (Q), takes value in the range $[0,1]$ and is defined as follows:

$$Q(C, C^*) = \frac{1}{|C^*|} \left| \left\{ \underset{C_j^* \in C^*}{\operatorname{argmax}} J\left(C_i, C_j^*\right), \forall C_i \in C \right\} \right| \tag{12}$$

6 RESULTS

In this section we report the detailed results of our experimental investigation. In the first part of the section we describe a sensitivity analysis of the considered algorithms. In the second part, we report the best results obtained by each algorithm for the datasets shown in Sec. 5.

6.1 Sensitivity Analysis

The number of communities to be detected in the network depends on the parameter k, that regulates the number of kernel members of each community. In order to evaluate the performance of the algorithms varying the parameter k, a sensitivity analysis has been performed. In Fig. 5 we report the results of our analysis, performed on the Philosophers dataset, computed in terms of Equivalence (as detailed in Eq. 12). We can see that, in general, all three algorithms show their best performance when the kernel size k is small.

In particular, for Standard and Overlapping the performance decreases sharply for $k \geq 7$, indicating that the nodes forming a kernel are usually very few. When we consider node attributes, however, the performance remains high for a larger value of k. This behavior is mainly due to the attribute similarities considered as "textual relationships" between nodes. These "relationships" derived by the textual similarity usually outnumber structural relationships, therefore leading to larger kernels. However, the performance starts dropping since $k = 6$, a value consistent with the result obtained by the other two algorithms.

An analogous sensitivity analysis has been performed on the other two benchmarks. It emerges that, also for bigger datasets, the number of kernel members are quite low. The results of this sensitivity analysis suggests that the experiments should be performed considering a small kernel size, within the range of 3-6 nodes.

6.2 Comparative Results

We perform experiments on the three benchmarks starting from the conclusions drawn from the sensitivity analysis step.

In Table 2 we report the results relative to the first dataset. In order to make the results comparable, we run the three algorithms with a kernel size $k = 3$, which has been previously proven as a good value for all three algorithms. In this case, the equivalence measure highlights a performance of $7,12 \pm 1,97$ for the Standard algorithm, $87,21 \pm 2,71$ for Overlapping Greedy-WeBA, and $90,91 \pm 3,95$ for Overlap-

Table 2: Performance results on the Philosophers dataset. Best results for each row are marked in bold.

Measures	Standard Greedy-WeBA	Overlapping Greedy-WeBA	Overlapping Greedy-WeBA with Node-Attributes
Recall (average)	39,05 ± 3,41	30,95 ± 0,77	**44,30** ± 1,39
Precision (average)	16,25 ± 2,27	**48,35** ± 1,44	35,77 ± 0,99
F1 Score (average)	21,66 ± 2,31	32,41 ± 0,47	**36,44** ± 1,77
Jaccard Index (average)	12,20 ± 1,47	20 ± 0,42	**23,08** ± 0,81

Table 3: Performance results on the Twitter dataset. Best results for each row are marked in bold.

Measures	Standard Greedy-WeBA	Overlapping Greedy-WeBA	Overlapping Greedy-WeBA with Node-Attributes
Recall (average)	31,61 ± 1,36	**58,22** ± 2,75	47,51 ± 2,93
Precision (average)	19,96 ± 0,98	31,73 ± 1,14	**40,10** ± 1,12
F1 Score (average)	24,43 ± 3,01	32,47 ± 2,53	**36,04** ± 2,37
Jaccard Index (average)	10,98 ± 0,89	19,74 ± 0,77	**22,53** ± 1,01

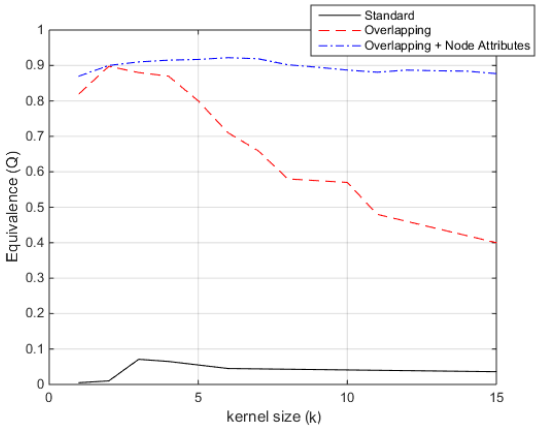

Figure 5: Sensitivity analysis for the three algorithms on Philosophers dataset.

ping Greedy-WeBA with Node Attributes. The first thing we can observe is a large increment in the equivalence measure when adding overlapping communities. This effect can be explained by the number of communities detected by the original Greedy-WeBA algorithm, which is very low compared to the number of ground truth communities. A similar behaviour can be observed for all the other measures: the introduction of overlapping communities in the algorithm lead to a better performance in terms of F-Measure (from 22% to 32%) and Jaccard index (from 12% to 20%).

When considering node attributes, we can observe that the equivalence value is relatively unchanged (the value increase from 87% to 90%). This means that the addition of overlapping communities is generally sufficient for detecting the majority of the ground truth communities. However, we can see that the values of F-Measure and Jaccard score increase significantly (from 32% to 36%, and from 20% to 23% respectively). Thus, the communities obtained by the algorithm that exploits node attributes are closer to the ground truth communities than overlapping communities only. In Table 3 and Table 4 we report the results obtained on the Twitter and Facebook datasets. Although Overlapping Greedy-WeBA with Node Attributes always outperforms the other two approaches for all the considered performance measures and for any value of the kernel size k, we only report the results obtained for $k = 3$. This kernel size has been selected because it provides a good tradeoff between performance and computational cost for the three algorithms.

We can observe that the results for Twitter and Facebook are consistent with those obtained on the Philosophers dataset. Allowing overlapping communities strongly improves the performance of the algorithm for both datasets (+8% and +7% for F-Measure, and +9% and +%7 for Jaccard index). This confirms that this improvement is essential when dealing with online social networks, whose users usually belong to multiple communities. The performance improvement obtained by considering node attributes together with overlapping communities is higher on the Facebook dataset than the Twitter one. This behavior is mainly related to the nature of the node attributes that have been considered. While Twitter attributes are related to words used by the social network users in their posts, Facebook attributes are related to their personal information (school institution, name of the company where they work, etc.) which may be more informative when determining the community to which they belong. The increment in the Twitter dataset, however, suggests that node attributes play a fundamental role even when they are obtained from a noisy source of information like user generated posts.

7 CONCLUSION

In this paper we introduced a kernel-based community detection algorithm that can discover overlapping communities using both the network structure and node attributes. The comparison with the baseline algorithm shows that the ability to find overlap-

Table 4: Performance results on the Facebook dataset. Best results for each row are marked in bold.

Measures	Standard Greedy-WeBA	Overlapping Greedy-WeBA	Overlapping Greedy-WeBA with Node-Attributes
Recall (average)	27,85 ± 2,01	40,02 ± 1,92	**55,60** ± 2,47
Precision (average)	32,10 ± 3,30	37,16 ± 3,05	**48,75** ± 2,99
F1 Score (average)	29,80 ± 1,40	36,45 ± 1,73	**51,99** ± 1,54
Jaccard Index (average)	17,48 ± 0,78	22,64 ± 1,66	**35,19** ± 1,44

ping communities is fundamental for detecting the correct groups of users in social networks, where often users can belong to several social circles (due to various interests, hobbies or relationships). Moreover, we showed that the inclusion of node attributes can provide important additional information, leading to results which better fit the real communities.

There are several possible directions for future work. For instance, we would like to improve the current algorithm by including a method for automatic inferring the best kernel-size. Moreover, we would like to study how the community kernels change dinamically over time, and how this affects auxiliary communities.

REFERENCES

Ahn, Y.-Y., Bagrow, J. P., and Lehmann, S. (2010). Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764.

Chang, J. and Blei, D. M. (2009). Relational topic models for document networks. In *International Conference on Artificial Intelligence and Statistics*, pages 81–88.

Donetti, L. and Munoz, M. A. (2004). Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(10):P10012.

Du, N., Wu, B., Pei, X., Wang, B., and Xu, L. (2007). Community detection in large-scale social networks. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 16–25. ACM.

Günnemann, S., Boden, B., Färber, I., and Seidl, T. (2013). Efficient mining of combined subspace and subgraph clusters in graphs with feature vectors. In *Advances in Knowledge Discovery and Data Mining*, pages 261–275. Springer.

Gunnemann, S., Farber, I., Boden, B., and Seidl, T. (2010). Subspace clustering meets dense subgraph mining: A synthesis of two paradigms. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 845–850. IEEE.

Leskovec, J., Lang, K. J., Dasgupta, A., and Mahoney, M. W. (2008). Statistical properties of community structure in large social and information networks. In *Proceedings of the 17th international conference on World Wide Web*, pages 695–704. ACM.

Leskovec, J. and Mcauley, J. J. (2012). Learning to discover social circles in ego networks. In *Advances in neural information processing systems*, pages 539–547.

Liu, Y., Niculescu-Mizil, A., and Gryc, W. (2009). Topic-link lda: joint models of topic and author community. In *proceedings of the 26th annual international conference on machine learning*, pages 665–672. ACM.

Mishra, N., Schreiber, R., Stanton, I., and Tarjan, R. E. (2008). Finding strongly knit clusters in social networks. *Internet Mathematics*, 5(1-2):155–174.

Newman, M. E. (2004a). Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330.

Newman, M. E. (2004b). Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133.

Newman, M. E. (2006a). Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104.

Newman, M. E. (2006b). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582.

Papadimitriou, S., Sun, J., Faloutsos, C., and Philip, S. Y. (2008). Hierarchical, parameter-free community discovery. In *Machine Learning and Knowledge Discovery in Databases*, pages 170–187. Springer.

Papadopoulos, S., Kompatsiaris, Y., Vakali, A., and Spyridonos, P. (2012). Community detection in social media. *Data Mining and Knowledge Discovery*, 24(3):515–554.

Rosvall, M. and Bergstrom, C. T. (2007). An information-theoretic framework for resolving community structure in complex networks. *Proceedings of the National Academy of Sciences*, 104(18):7327–7331.

Wang, L., Lou, T., Tang, J., and Hopcroft, J. E. (2011). Detecting community kernels in large social networks. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 784–793. IEEE.

Xie, J. and Szymanski, B. (2012). Towards linear time overlapping community detection in social networks. In Tan, P.-N., Chawla, S., Ho, C., and Bailey, J., editors, *Advances in Knowledge Discovery and Data Mining*, volume 7302 of *Lecture Notes in Computer Science*, pages 25–36. Springer Berlin Heidelberg.

Yang, J. and Leskovec, J. (2013). Overlapping community detection at scale: A nonnegative matrix factorization approach. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pages 587–596. ACM.

Yang, J., McAuley, J., and Leskovec, J. (2013). Community detection in networks with node attributes. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1151–1156. IEEE.