Detecting Overlapping Communities in Two-mode Data Networks using Formal Concept Analysis

Abir Messaoudi, Rokia Missaoui
Mohamed-Hamza Ibrahim

LARIM, Université du Québec en Outaouais, Québec, Canada
{mesa08,missro01,ibrm05}@uqo.ca

Abstract. Social networks frequently feature complex structures such as two-mode data expressed by bipartite graphs. Most research work on community detection in bipartite graphs focus on either finding non-overlapping communities or identifying overlapping ones by first projecting two-mode (bi-dimensional) data into two one-mode tables which are further analyzed. However, this often leads to a loss of information and produces inaccurate communities. Therefore, efficiently detecting communities in such two-mode data networks often remains a key challenge in social network analysis. In this paper, we introduce a novel three-step strategy to detect overlapping as well as hierarchically nested communities in bipartite graphs. First, we extract the formal concepts that represent potential groups in the social network. Then, we rank and filter the obtained groups to keep only core ones that have a high mean of stability and separation. Finally, we detect communities by refining the core groups using a Silhouette Analysis. Our experiments on real-world social networks show that our method can accurately identify overlapping communities.

1 Introduction

Although numerous community detection methods have been proposed, relatively few ones are designed for heterogeneous or multi-layer networks or even two-mode data ones (*i.e.*, two types of nodes and one type of links). The main focus is generally on homogeneous (commonly called one-mode data) networks, which are extracted from real-world networks by considering one type of nodes (objects) and one type of links. Moreover, most of the studies are concerned with disjoint communities rather than overlapping and hierarchically nested ones which are very common in real-life situations and applications. For the reasons cited above, the number of research studies on identifying overlapping and nested communities in heterogeneous networks is growing up. This is an issue we seek to address in the following paper. Our method is completely unsupervised and can automatically determine both the number of communities and their description. It relies on the inherent structure discovered from data using Formal Concept Analysis (FCA), and two relevancy metrics associated with formal concepts, namely stability and separation. Finally, it makes use of Silhouette analysis to refine the process of community detection.

The rest of this paper is organized as follows: Section 2 provides background on social network analysis (SNA) and FCA while Section 3 gives a brief overview of related work. In Section 4 we describe and illustrate our approach. Section 5 presents the empirical study while the conclusion and future work are given in Section 6.

2 Background

2.1 Social Network Analysis

There are many types of social networks. A particular one is the two-mode data network with two types of nodes and one type of ties that are only established between two nodes belonging to different sets. The first set represents objects (actors) while the second one represents attributes (characteristics, properties). Such a network is expressed by a bipartite graph $\mathcal{B} = (U, V, E)$, where U and V are two independent/disjoint sets of vertices, and E is the set of edges between one node in U and one node in V. For instance, a community in a social network may represent researchers closely connected to publications according to a co-authorship link.

The most studied topics in SNA are link prediction, influence maximization and propagation, position and role computation, network destabilization vs reinforcement, and community detection. The latter topic (Fortunato, 2010) aims at finding clusters as sub-graphs within a given network.

Communities can be either disjoint or overlapping (Wang and Fleury, 2013; Xie et al., 2013). Partial or full nesting is an important aspect of real-world networks, and means that communities are strongly hierarchical. For example, a community of graduate students having the same supervisor is mainly a subset of a larger group of students enrolled in the same program.

2.2 Formal Concept Analysis

Formal Concept Analysis is a branch of applied mathematics, which is based on a formalization of concept and concept hierarchy (Ganter and Wille, 1999; Ganter and Obiedkov, 2016). It uses a formal binary context to construct a concept (Galois) lattice whose nodes are formal concepts.

A formal context is a triple $\mathbb{K} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$, where $\mathcal{G}$ is a set of objects, $\mathcal{M}$ a set of attributes, and $\mathcal{I}$ a binary relation between $\mathcal{G}$ and $\mathcal{M}$ with $\mathcal{I} \subseteq \mathcal{G} \times \mathcal{M}$. For $g \in \mathcal{G}$ and $m \in \mathcal{M}, (g, m) \in \mathcal{I}$ holds (i.e., $(g, m) = 1$) iff the object g has the attribute m, and otherwise $(g, m) \notin \mathcal{I}$ (i.e., $(g, m) = 0$). Given arbitrary subsets $A \subseteq \mathcal{G}$ and $B \subseteq \mathcal{M}$, the following derivation operators are defined:

$$A' = \{m \in \mathcal{M} \mid \forall g \in A, (g, m) \in \mathcal{I}\},\ A \subseteq \mathcal{G}$$

$$B' = \{g \in \mathcal{G} \mid \forall m \in B, (g, m) \in \mathcal{I}\},\ B \subseteq \mathcal{M}$$

where A' is the set of attributes common to all objects of A and B' is the set of objects sharing all attributes from B. The closure operator $(.)''$ implies the double application of $(.)'$. The subsets A and B are closed when $A = A''$, and $B = B''$.

A formal concept of the context $\mathbb{K} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$, is a pair $c = (A, B) \subseteq \mathcal{G} \times \mathcal{M}$ where $A' = B$ and $B' = A$. A is called the *extent* of c while B is its *intent*.

A partial order exists between two concepts $c_1 = (A_1, B_1) \leq$ and $c_2 = (A_2, B_2)$ if $A_1 \subseteq A_2 \iff B_1 \supseteq B_2$.

The set $\mathcal{C}$ of all concepts together with the partial order form a concept lattice.

The selection of the most relevant concepts from a possibly huge set of elements is a crucial task. Stability and separation indices (Kuznetsov, 2007; Buzmakov et al., 2014; Kuznetsov and Makhalova, 2018) are among the relevancy measures used for concept selection.

For a given formal concept $c = (A, B)$, the *intensional stability* is defined as follows:

$$\sigma(c) = \frac{\mid \{e \in \mathcal{P}(A) | e' = B\} \mid}{2^{|A|}} \tag{1}$$

It measures the strength of dependency between the intent B and the objects of the extent A. More precisely, it expresses the probability to maintain B closed when a subset of noisy objects in A are deleted with equal probability. In fact, this measure quantifies the amount of noise in the extent A and overfitting in the intent B. Obviously, the stability index highlights concepts with high internal cohesion since most of the subsets in A have the intent B whenever the stability is high. Such a feature can then be exploited to retrieve communities within networks.

The separation $\alpha(c)$ of a formal concept $c = (A, B)$ (Klimushkin et al., 2010) is computed as follows where g' is the set of attributes of an object g while m' gives the objects that have the attribute m:

$$\alpha(c) = \frac{\mid A \mid \cdot \mid B \mid}{\sum_{g \in A} \mid g' \mid + \sum_{m \in B} \mid m' \mid - \mid A \mid \cdot \mid B \mid} \tag{2}$$

It expresses the proportion of cells in c among the area corresponding to the intent of each element in A and the extent of every attribute in B. Therefore, it estimates the specificity of the object-attribute relation of a concept with respect to the formal context, and assesses how much noise exists in the concept.

Silhouette coefficient ($\mathcal{S}$) is mainly used to capture the quality or goodness of a clustering output (Rousseeuw, 1987). For a given object o_i in the data set, let C_a denote the cluster to which it has been assigned to, and $a(o_i)$ the average distance between o_i and all other objects within the same cluster C_a. Now, let us consider any cluster C_c different from C_a and compute the average distance between o_i and all other objects in C_c. Let $b(o_i)$ be the lowest average distance of o_i to all points in any other cluster, of which o_i is not a member. Silhouette coefficient for object o_i is given by:

$$\mathcal{S}\left(o_i\right) = \frac{b\left(o_i\right) - a\left(o_i\right)}{max\{a\left(o_i\right), b\left(o_i\right)\}} \tag{3}$$

where $\mathcal{S}(o_i) \in [-1, 1]$. A negative value indicates that o_i is assigned to the wrong cluster while a positive value means that o_i is in the right cluster and far away from its neighboring clusters. When $\mathcal{S}(o_i) = 0$, it shows that o_i is very close to the decision boundary between two neighboring clusters.

The silhouette coefficient of a cluster is then computed as the average of the Silhouette coefficient of its elements.

3 Related work

Although numerous community detection methods have been developed for social network analysis, most of them cannot be directly applied to mine two-mode data networks.

There are two main categories of work for community detection in two-mode data networks. The first one assumes a projection of the initial two-mode network into two one-mode data networks where nodes of the same type are connected if they share links to the same nodes of the second node type. One of these methods is "dual-projection" (Everett and Borgatti, 2013) that produces non overlapping biclusters expressing regular or structural equivalence. However, the number of biclusters produced by this method must be set in advance by the user.

The second category, known as direct or combined approach, extends existing methods to determine the community structure of both modes simultaneously. The related studies are mainly based on modularity, clustering, biclustering, or block modelling. Modularity is defined (Newman and Girvan, 2004), (Blondel et al., 2008) as a quality function that evaluates clusters based on the idea that a cluster is a set of nodes connected based on sharing common properties in the network. Concretely, starting from a node or a small set of nodes, a community can be obtained by adding neighboring nodes that improve a given quality function. The choice of the quality function depends on the context and application requirements.

Block modelling, which was generalized to two mode-data networks in (Borgatti, 2009) consists to group nodes inside the same block whenever they are statistically equivalent in terms of their connectivity to nodes within the block.

There are a few studies that exploit FCA for community identification in two-mode data networks. (Roth et al., 2008) use FCA to consider only concepts whose support is over a given threshold. Clearly, some interesting rare concepts may be discarded. In order to avoid this flaw, (Jay et al., 2008) rely on concept stability and support measures to detect communities. Computed concepts that exceed a given threshold of these measures are kept. In fact, the proposed method retains the rare but stable concepts and the frequent but unstable concepts. Another method was proposed in (Crampes and Plantié, 2012). It takes only the concepts of the first two layers of the concept lattice, and computes cohesion, separation and autonomy of concepts to identify communities. The cohesion of a given community is based on the Jaccard coefficient, which is not the best adapted score for FCA (Kuznetsov and Makhalova, 2018).

4 Proposed Approach

At a conceptual level, our overall strategy contains the following key steps: (i) generate the whole set of concepts - without the partial order - from the formal context that describes the two-mode data network, (ii) compute the autonomy of concepts as the harmonic mean of its stability and separation indices to further select the concepts with the highest autonomy scores without relying on any threshold, and (iii) refine the core communities using the Silhouette coefficient analysis to get the final overlapping communities.

4.1 Generating Formal Concepts

At the beginning we build the formal context $\mathbb{K}$ (see Table 1) of the two-mode network $\mathcal{B}$ by computing *the incidence matrix* as follows:

$$\mathbb{K} = (\mathcal{G}, \mathcal{M}, \mathcal{I}) = \begin{cases} (g_i, m_j) = 1 & \text{If } g_i \in \mathcal{G}, m_j \in \mathcal{M}, \exists\, (g_i, m_j) \in \mathcal{I}, \\ (g_i, m_j) = 0 & \text{Otherwise.} \end{cases} \tag{4}$$

Then, the set of concepts is computed. For instance, The concept $c = (\{6, 7\}, \{E, F\})$ is highlighted in Table 1.

	A	B	C	D	E	F	G
1	1	0	1	0	1	0	1
2	0	1	0	1	0	0	0
3	0	1	1	0	0	0	0
4	0	0	0	1	0	0	1
5	1	0	1	1	0	0	0
6	0	0	0	0	1	1	0
7	0	0	0	0	1	1	0
8	0	0	0	0	1	0	0

TAB. 1 – *An example of formal context.*

We used the *In-Close* algorithm (Andrews, 2011) to generate the set $\mathcal{C}$ of concepts from the constructed formal context.

4.2 Selecting Cohesive and Separable Concepts

This step selects a relatively small set of concepts from $\mathcal{C}$ that can serve as core communities without relying on any threshold. We consider a concept $c \in \mathcal{C}$ as a core community if its extent represents a cohesive group and the objects are separable from the rest of objects found in other concepts. This means that this concept represents a very likely standalone community or a portion of a larger potential community. Now, given a concept $c = (A, B)$, how can we measure the cohesion and separability of its objects? Here the stability and separation indices come to play to quantify the strength of ties between objects inside c and the weakness of ties between the objects in the extent of c and all other objects in other concepts. This in fact implies that a high stability or separation index of a concept indicates that the objects inside this concept are very cohesive (i.e., they have strong ties among each other) and are very separable (i.e., they have weak ties with all other objects that exist outside their concept). Given a concept c, we can compute the harmonic mean of its stability $\sigma(c)$ and separation $\alpha(c)$ indices to obtain a new score that we call autonomy: $\zeta(c) = 2 \times \left(\frac{\sigma(c) \times \alpha(c)}{\sigma(c) + \alpha(c)} \right)$

For example, the concept $c = (\{6, 7\}, \{E, F\})$ in our example, has a stability value of $\frac{3}{4} = 0.75$ and a separation value of $\frac{(2\times2)}{((2+2)+(4+2)-(2\times2))} = 0.66$. Thus, its autonomy score is equal to $2 \times \frac{0.75\times0.66}{0.75+0.66} = 0.70$. This concept represents likely a core community.

4.3 Refining Core Communities

This stage aims at identifying the final overlapping communities by refining the core communities $\tilde{\mathcal{C}}$ using the Silhouette coefficient. This helps us check that each object is in the proper communities. If it is not the case, then the object is moved to the right community.

Algorithm 1 Community detection procedure

Input: Formal context $\mathbb{K} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$
Output: Set of overlapping communities ($\tilde{\mathcal{C}}$)

$\Gamma \leftarrow \mathcal{D} \leftarrow \emptyset$
$\mathcal{C} \leftarrow$ Compute the concepts of $\mathbb{K}$
for *each concept* $c = (A, B) \in \mathcal{C}$ **do**
    $t_c \leftarrow$ Compute the autonomy $\zeta(c)$
    $\Gamma \leftarrow \Gamma \cup \{(c, t_c)\}$
end for
// Sort the concepts in a descending order of t_c in Γ
$\mathcal{D} \leftarrow$ Sort(Γ)
// Select the core communities
$\tilde{\mathcal{C}} \leftarrow \mathcal{O} \leftarrow \emptyset$
while $\mathcal{O} \neq \mathcal{G}$ **do**
    $(A, B) \leftarrow \mathcal{D}$.**pop**()
    if $(A \not\subset \mathcal{O})$ **then**
        $\tilde{\mathcal{C}} \leftarrow \tilde{\mathcal{C}} \cup \{(A, B)\}$ // (A, B) is a core community
        $\mathcal{O} \leftarrow \mathcal{O} \cup \{A\}$
    end if
end while
// Refine core communities in $\tilde{\mathcal{C}}$ to detect the final ones.
for *each core community* $c = (A, B) \in \tilde{\mathcal{C}}$ **do**
    for $e \in A$ **do**
        $s_e \leftarrow \mathcal{S}(e)$ // Compute the Silhouette coefficient of e
        if $s_e < 0$ **then**
//Move e from $c = (A, B)$ to the nearest community
            $A \leftarrow A \setminus \{e\}$
            $(A_1, B_1) \leftarrow$ Find the closest core community to e
            $A_1 \leftarrow A_1 \cup \{e\}$
        else if $s_e = 0$ **then**
//Add e of c to the nearest community
            $c_1(A_1, B_1) \leftarrow$ Find the closest core community to e
            $A_1 \leftarrow A_1 \cup \{e\}$
        end if
    end for
end for
return($\tilde{\mathcal{C}}$)

Algorithm 1 gives the pseudo-code of our procedure in which the input is the formal context

	Core	Autonomy
$\tilde{C}_1$	({6, 7}, {E, F})	0.7
$\tilde{C}_2$	({1, 6, 7, 8}, {E})	0.53
$\tilde{C}_3$	({2, 4, 5}, {D})	0.46
$\tilde{C}_4$	({3}, {B, C})	0.44

TAB. 2 – *Core communities*

$\tilde{C}_1$	({6, 7}, {E, F})
$\tilde{C}_2$	({6, 7, 8}, {E})
$\tilde{C}_3$	({4, 5}, {D})
$\tilde{C}_4$	({3,1,5}, {C})

TAB. 3 – *Final communities*

associated with the network. First, it computes the set of concepts. Then, it calculates the autonomy score of all concepts (lines 3-6) to further sort the concepts in Γ in a descending order of their autonomy value (line 7). Subsequently, it constructs the core community collection $\tilde{\mathcal{C}}$ by selecting the concepts with the highest autonomy values until the set of selected concepts cover all objects (and their attributes) in the formal context (lines 8-15). At a later step (lines 16-28), it refines the group of core communities by calculating the Silhouette coefficient $\mathcal{S}(e)$ of each object $e \in A$ in each core community $c = (A, B) \in \tilde{\mathcal{C}}$. If the Silhouette coefficient value is less than 0, then e is not in the correct community c and is then moved to the closest core community. If the Silhouette coefficient value is equal to 0, then e will also appear in another close core community. Otherwise, e is kept in its community c. After refining all core communities, the algorithm outputs the final detected overlapping communities as given in Tables 2 and 3. One can notice that two changes occurred at the last step (Silhouette analysis) since object 1 is moved from $\tilde{C}_2$ to $\tilde{C}_4$ and object 5 is shifted from $\tilde{C}_3$ to $\tilde{C}_4$. Community $\tilde{C}_1$ is nested into $\tilde{C}_2$ while the other communities are now overlapping.

It is important to note that the description (*i.e.*, the shared attributes) of communities is given by the intent B of $c = (A, B)$ whenever a core community is not altered by an object insertion or elimination. Otherwise, it can be easily obtained by computing A', *i.e.*, the set of attributes associated with the objects in A.

Complexity Analysis

The computation of formal concepts is $O(|\mathcal{G}|^2 \times |\mathcal{M}| \times |\mathcal{C}|$, where $|\mathcal{G}|$, $|\mathcal{M}|$, and $|\mathcal{C}|$ represent the size of the set of objects, attributes, and concepts respectively. To calculate the autonomy of a concept $c = (A, B)$, we need to compute both the stability and the separation. The first measure can be approximately computed using low-discrepancy sampling (Ibrahim and Missaoui, 2018) in $O(|\mathcal{S}|)$, where $|\mathcal{S}|$ stands for the number of samples generated from $\mathcal{P}(A)$. As the complexity of computing the separation for one concept is $O(|\mathcal{G}| \times |\mathcal{M}|)$, the calculation of the autonomy for all concepts is then $O(|\mathcal{C}| \times (|\mathcal{S}| + (|\mathcal{G}| \times |\mathcal{M}|))$. The sort of concepts is $O(|\mathcal{C}| \times Log(|\mathcal{C}|))$. Finally, since the Silhouette analysis needs to compare each element of a cluster with the objects of the other ones, its complexity is $O(|\tilde{\mathcal{C}}| \times |\mathcal{M}| \times |\mathcal{G}|^2)$, where $\tilde{\mathcal{C}}$ stands for the generated commmunity set. The overall complexity is therefore dominated by the first and last steps of the algorithm.

5 Experimental Evaluation

To evaluate the proposed approach, we analyze its performance and accuracy against four other community detection algorithms using real-world networks with built-in nested community structure. Algorithms are implemented in Python and the experiments were executed on an Intel Core i7 with 3.4 GHz and a RAM of 16 GB.

The datasets are as follows (see Table 4) where the first three sets have ground-truth communities: (1) Southern women Davis [1], which describes the participation of eighteen Southern women to fourteen social events, (2) Zoo [2], which gives the description of different types of animals in a zoo, (3) Customer-Product (C-P), which describes 1143 customers in terms of 865 products they ordered (see Gazelle.com), (4) senators x committees, which indicates the links between the senators in the 124-th Maine State Legislature and the legislative committees, (5) *DBpedia* languages which involves the semantic web of official languages spoken by people living in different countries, and (6) Star Alliance, which captures a set of airline companies and their flying destinations in Year 2000.

We then compare the accuracy and performance of our proposed approach with the following community detection algorithms (see Section 3): (i) Oslom (Lancichinetti et al., 2010), (ii) (Crampes and Plantié, 2012), (iii) (Jay et al., 2008), and (iv) Bitector procedure (Du et al., 2008) which exploits biclique computation.

	Objects	Attributes	Links	Density in %
Southern woman (S-W)	18	14	89	70
Zoo	101	17	746	43.4
Customer-Product (C-P)	1143	865	2008	0.4
Senators x Committees (Senat)	189	59	890	7.98
DBpediaLanguages (PL)	316	169	9022	16.8
Star alliance (Star)	28	58	579	35.6

TAB. 4 – *A brief description of the tested social networks*

Furthermore, we consider three metrics to assess the accuracy of the algorithms, namely Omega index, Overlapping Normalized Mutual Information (NMI), and link-belonging modularity (Collins and Dent, 1988; Chakraborty et al., 2017; Nicosia et al., 2009). The latter is used for networks without ground-truth communities.

5.1 Results and discussion

It can be observed from Figure 1 that our algorithm is tested under two variants: one in which the third (refinement) step is included (i.e., the three steps), and one in which the third step is excluded. One can see that it behaves well in terms of accuracy compared to the tested methods for the two kinds of datasets (with or without ground-truth communities). This is mainly the case when the third step of Silhouette analysis is used. Our algorithm is followed by Bitector for the first kind of datasets (with ground-truth for Southern women, Zoo and C-P)

1. https://networkdata.ics.uci.edu/netdata/html/davis.html
2. http://archive.ics.uci.edu/ml/datasets/zoo

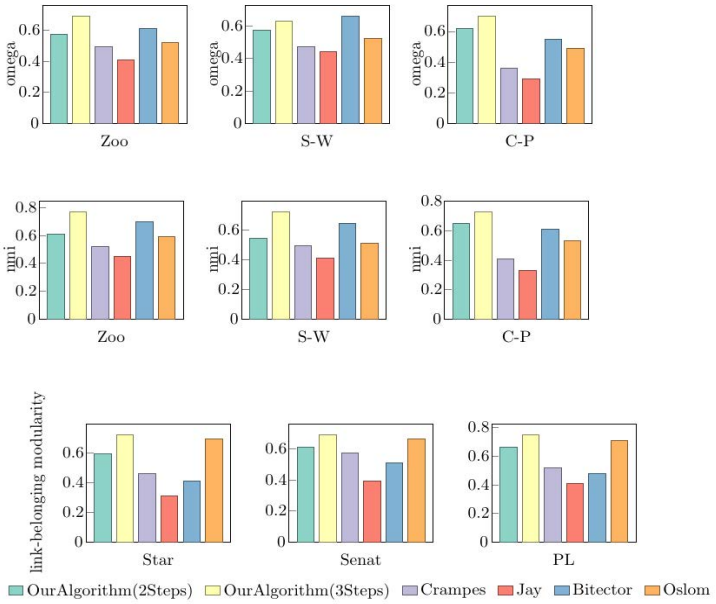

FIG. 1 – *Evaluation of the detected communities using NMI, OMEGA and link-belonging modularity*

and by Lancichinetti for the second kind. Indeed, the latter approach did not perform so well on networks with ground-truth communities although it did better on datasets without ground-truth. This may be due to the fact that the metric used to assess communities without ground truth was based on modularity. The less accurate method seems to be Jay's algorithm probably because some objects may be ignored using a threshold on the used metrics. BiTector finds small communities and works slightly better with sparse networks.

For our approach, the execution time is exhibited according to the two variants given earlier, and includes the cost of concept computation. The time is given in seconds and represents the average of five execution times of each one of the evaluated algorithms.

As shown in Figure 2, the execution time of our whole procedure is more important than the one for the first two steps. However, the accuracy is improved in the former variant as previously observed from Figure 1. The less performing algorithms are Crampes's and Jay's procedures.

6 Conclusion

In this paper, we proposed a new method for detecting overlapping and hierarchically nested communities in two-mode data networks. Our method does neither require a user-predefined number of groups nor thresholds on metrics. It can automatically identify cohesive and separable communities and their description through the shared features of formal concepts

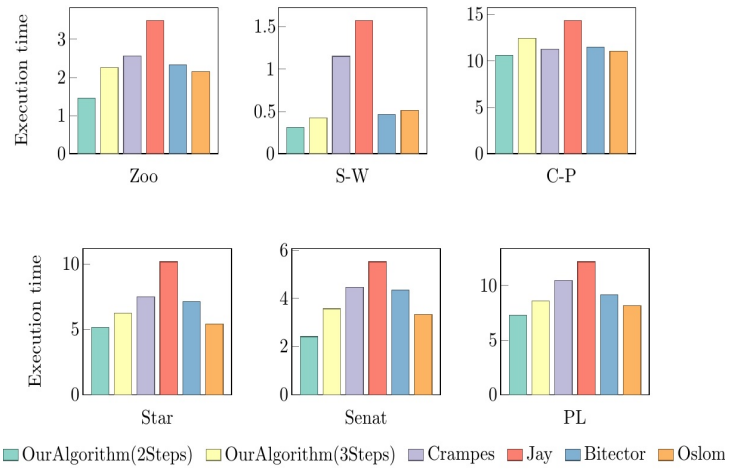

FIG. 2 – *Execution time (in secs) for the tested community detection algorithms*

using FCA-based metrics like stability and separation. Finally, it uses Silhouette coefficient to refine communities, which combines both the cohesion and separation measures.

We tested our algorithm on a set of networks with or without ground-truth communities and showed its accuracy and execution time against four other methods. Further empirical studies are needed to better assess its efficiency and accuracy in large and dense datasets.

From a computational complexity point of view, there is still room for improvement. The running time of the proposed algorithm can be further reduced using optimization techniques (e.g., computing a subset of concepts) and/or other relevancy concept measures, or exploiting the notion of context coverage (Ferjani et al., 2012) to identify core communities. We also plan to extend our work to identify communities in multi-layer networks where nodes from a given layer can be linked to some other nodes of another layer. Finally, we believe that when the set of generated communities is large, a percolation step can be added to reduce such a set.

Acknowledgments

The second author acknowledges the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Andrews, S. (2011). In-close2, a high performance formal concept miner. In *International Conference on Conceptual Structures*, pp. 50–62. Springer.

Blondel, V. D., J.-L. Guillaume, R. Lambiotte, and E. Lefebvre (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment 2008*(10), P10008.

Borgatti, S. P. (2009). 2-mode concepts in social network analysis. *Encyclopedia of complexity and system science 6*, 8279–8291.

Buzmakov, A., S. O. Kuznetsov, and A. Napoli (2014). Scalable estimates of concept stability. In *International Conference on Formal Concept Analysis*, pp. 157–172. Springer.

Chakraborty, T., A. Dalmia, A. Mukherjee, and N. Ganguly (2017). Metrics for community analysis: A survey. *ACM Computing Surveys (CSUR) 50*(4), 54.

Collins, L. M. and C. W. Dent (1988). Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions. *Multivariate Behavioral Research 23*(2), 231–242.

Crampes, M. and M. Plantié (2012). Détection de communautés dans les graphes bipartis. In *IC 2012*, pp. 125.

Du, N., B. Wang, B. Wu, and Y. Wang (2008). Overlapping community detection in bipartite networks. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pp. 176–179. IEEE Computer Society.

Everett, M. G. and S. P. Borgatti (2013). The dual-projection approach for two-mode networks. *Social Networks 35*(2), 204–210.

Ferjani, F., S. Elloumi, A. Jaoua, S. B. Yahia, S. A. Ismail, and S. Ravan (2012). Formal context coverage based on isolated labels: An efficient solution for text feature extraction. *Inf. Sci. 188*, 198–214.

Fortunato, S. (2010). Community detection in graphs. *Physics reports 486*(3-5), 75–174.

Ganter, B. and S. A. Obiedkov (2016). *Conceptual Exploration*. Springer.

Ganter, B. and R. Wille (1999). *Formal Concept Analysis: Mathematical Foundations*. Springer-Verlag New York, Inc. Translator-C. Franzke.

Ibrahim, M. H. and R. Missaoui (2018). An efficient approximation of concept stability using low-discrepancy sampling. In *Graph-Based Representation and Reasoning - 23rd International Conference on Conceptual Structures, ICCS 2018, Edinburgh, UK, June 20-22, 2018, Proceedings*, pp. 24–38.

Jay, N., F. Kohler, and A. Napoli (2008). Analysis of social communities with iceberg and stability-based concept lattices. In *International Conference on Formal Concept Analysis*, pp. 258–272. Springer.

Klimushkin, M., S. Obiedkov, and C. Roth (2010). Approaches to the selection of relevant concepts in the case of noisy data. In *International Conference on Formal Concept Analysis*, pp. 255–266. Springer.

Kuznetsov, S. O. (2007). On stability of a formal concept. *Annals of Mathematics and Artificial Intelligence 49*(1), 101–115.

Kuznetsov, S. O. and T. Makhalova (2018). On interestingness measures of formal concepts. *Information Sciences 442*, 202–219.

Lancichinetti, A., F. Radicchi, J. J. Ramasco, and S. Fortunato (2010). Finding statistically significant communities in networks. *CoRR abs/1012.2363*.

Newman, M. E. and M. Girvan (2004). Finding and evaluating community structure in networks. *Physical review E 69*(2), 026113.

Nicosia, V., G. Mangioni, V. Carchiolo, and M. Malgeri (2009). Extending the definition of modularity to directed graphs with overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment 2009*(03), 3–24.

Roth, C., S. Obiedkov, and D. G. Kourie (2008). On succinct representation of knowledge community taxonomies with formal concept analysis. *International Journal of Foundations of Computer Science 19*(02), 383–404.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics 20*, 53–65.

Wang, Q. and E. Fleury (2013). Overlapping community structure and modular overlaps in complex networks. In *Mining Social Networks and Security Informatics*, pp. 15–40. Springer.

Xie, J., S. Kelley, and B. K. Szymanski (2013). Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM computing surveys (csur) 45*(4), 43.

Résumé

Les réseaux sociaux ont fréquemment des structures complexes comme ceux à deux modes représentés par des graphes bipartis. Plusieurs travaux sur la détection de communautés mettent l'accent soit sur l'identification de groupes disjoints ou chevauchants en procédant d'abord à la projection des données à deux modes (dimensions) en deux tables à un seul mode qui sont ensuite analysées. Cependant, cela entraîne une perte d'information et aboutit à des communautés mal définies. Ainsi, la détection précise des communités dans un graphe biparti reste un défi majeur en analyse de réseaux sociaux. Dans cet article, nous introduisons une approche à trois étapes pour la détection de communautés chevauchantes et même imbriquées dans les graphes bipartis. Tout d'abord, on détermine les concepts formels à partir des données. Ensuite, les concepts ayant une valeur élevée de la moyenne de la stabilité et de la séparation sont retenues comme les communautés de base. Finalement, une analyse Silhouette permet de raffiner l'identification des communautés. Des tests préliminaires sur des réseaux réels montrent que notre approche permet d'identifier correctement des communautés chevauchantes.