Information integration for detecting communities in attributed graphs

Juan David Cruz Gomez, Cécile Bothorel

Information integration for detecting communities in attributed graphs

Juan David Cruz
Lab-STICC, UMR CNRS 3192
Dpt. LUSSI, Telecom-Bretagne
Technopôle Brest - Iroise, 29238 France
Tel: +33 (0)2 29 00 12 79
Email: juan.cruzgomez@telecom-bretagne.eu

Cécile Bothorel
Lab-STICC, UMR CNRS 3192
Dpt. LUSSI, Telecom-Bretagne
Technopôle Brest - Iroise, 29238 France
Tel: +33 (0)2 29 00 14 47
Email: cecile.bothorel@telecom-bretagne.eu

***Abstract*—Real social networks can be described using two dimensions: first a *structural* dimension that contains the social graph, e.g. the actors and the relationships between them, and second a *compositional* dimension containing the actors' attributes, e.g. their profile. Each of these dimensions can be used independently to cluster the nodes and explain different phenomena occurring on the social network, whether from a connectivity or an individual perspective. In the case of community detection problem, an emergent research field explores how to include relationships and node attributes in an integrated clustering process. In this paper, we present a novel approach which integrate two partitions, one structural and one compositional, *after* they habe been generated by dedicated and specialized clustering steps.**

We rely on a contingency matrix with structural groups in rows and compositional ones in columns. The problem is to manipulate rows and columns to provide a new partition which maintains a good trade-off between both dimensions. In this paper we propose two strategies to control the combination. Tested on real-world social networks, the final partitions are evaluated in terms of entropy and density, and compared to pure structural or compositional partitions. The unified partitions show interesting properties, such as cohesive and homogeneous groups of actors. The method offers fine control on the combination process, giving new search capabilities to analysts without requiring the recomputation of the partitions.

***Index Terms*—Graph clustering; contingency matrix.**

I. Introduction

According to Wasserman and Faust [1] social networks are represented by two dimensions or variables. The *structural* variable is used to describe the network in terms of the links between the actors. The *composition* variable describes each actor individually with attributes such as origin, preferences, number of sent messages or other profile information.

Each variable has been defined in different spaces (friendship vs. competencies for example) making unfeasible their direct comparison, and therefore the definition of a unique distance measure, in the case of clustering, where the goal is to group close objects. Additionally, the approaches differ for each variable. Clustering graphs or clustering attribute/value tables involve different techniques and different measures of quality.

Most of the existing methods for clustering attributed graphs propose an integrated process to find a partition based on both the links and the attributes. But if the combination of the variables does not fit the goal, the analyst has to recalculate the partition with different parameters or try another appproach. The idea of this preliminary work is to de-correlate the stages of clustering and combination of variables. First, favoring specialized techniques for each type of variable that produces good quality partitions. Second, a method offering the reuse of partitions is valuable in huge networks in terms of computational complexity. In this work, we present a method that is able to combine existing good structural and compositional partitions to find a new groups of well-connected and similar nodes.

We rely on a contingency matrix to describe the agreements between the partitions. We assume that each partition can be explained in terms of the other one. By manipulating the rows and columns, we offer opportunities to control how to decompose the structural groups according to the composition information (or vice versa).

The paper is organized as follows: Section II presents some important works in community detection in attributed graph, in Section III the problem and some basic notation are introduced. Section IV presents the algorithm and Section V presents some experiments on real-world networks, before the conclusion.

II. Related Work

Several methods have been developed to detect communities in an attributed graph. Neville et al. [2] present a clustering approach that uses a similarity metric S_{ij} to compare the attributes of each socially linked pair of nodes i and j. They modify the weights of the edges according to S_{ij}, and then compute the communities using either a Monte-Carlo recursive clustering or a $k-$means based algorithm. The proposed similarity function is called matching coefficient; this function will set the weight of each edge $e(u, v) \in E$ as the count of common attributes between u and v. Steinhaeuser et al. [3] use different similarity measures to value the edges: two structural measures (based on Jaccard or clustering coefficient) and one measure involving either discrete or continuous attributes. If the edge weight exceed a threshold, the linked nodes are assigned the same cluster. The authors gauge the quality of the

partition with the modularity function proposed by Newman and Girvan [4].

We have also proposed a way to modify the weights of the edges before the application of the community detection process (see Cruz et al. [5], [6]). Our approach differs on how the compositional information is exploited: we do not compute a similarity function between pairs of nodes, but we use a clustering method on the attributes of the nodes. We propose the use of a self-organizing map (SOM), a neural network trained to find low-dimensional latent information about the attributes; our choice fell on SOM because of its robustness to noise and its capability to translate high dimensionality into low dimensionality spaces [7] (attributes may represent textual personal web page for example). During the second stage, we change the weights of the edge according to the groups revealed by the compositional clustering: for each edge $e(u, v) \in E$, if u and v belong to the same group in the SOM, the weight of $e(u, v) \in E$ is changed to a value proportional to a constant $\alpha > 1$. In a last step, the graph is clustered using the Louvain method [8] which optimizes the modularity and takes the weights into account.

Recently Villa-Vialaneix et al. [9] present another SOM-based approach. Their idea is to rely on kernels to map the original data into an (implicit) Euclidean space where the standard SOM can be used. They define a multi-kernel similarity function to compute the distance between the graph i.e., the structure and the nodes attributes, and the neurons of the SOM. This multi-kernel is a linear combination of several functions allowing to integrate the structural and the attribute similarity of the nodes in the graph, i.e., in this case the kernel is composed by two functions. The use of a kernel allows to automatically tune the combination. This approach takes also advantage of the visual representation of the SOM, that is a bi-dimensional grid in which each neuron (represented as a pie chart) represents a group of nodes and the size of each neuron is proportional to the number of observations associated with the neuron.

Combe et al., [10], [11] present two approaches for clustering attributed graphs. The first one is similar to the one presented in [3] but with a different similarity function and as in [6], authors use the Louvain method to cluster the resulting graph. The second approach is the use of a linear combination of an attribute similarity measure and a structural measure. Additionally, authors define a framework for comparing the resulting partition with other ground truth methods with the Rand Index.

Li et al., [12] present an algorithm to find groups of papers, i.e., the nodes are documents and the links are defined by the coexistence of a reference between papers and the additional information is given by the text which is clustered using LDA. The algorithm is composed of four steps: detection of cores, core merging, affiliation and classification. The first step is designed to identify documents that are frequently referenced, seen as the community seeds. On the second step the algorithm merges the cores based on their similarity. During the third step the nodes are assigned to one or more cores according to a relationship propagation. The last step is used to fine-tune the communities and remove nodes that may be false hits.

Ge et al., [13] present an approach for clustering attributed graphs, using at the same time the structure and the attributes of the nodes. Authors propose the connected k centers – CkC method. This method is composed of three main steps: first pick k random nodes as clusters centers, second all the nodes are assigned to one of the k clusters by traversing the graph using breadth-first search, and third the centroids of the clusters are recalculated. The second and third steps are repeated until there are not further changes in the clusters centroids. This approach is based on the $k-$means method.

Zhou et al., [14], present an algorithm that uses a random walk through a predefined set of k clusters, and try to maximize the distance between clusters by moving nodes according to their similarity. First, they create an augmented graph from the node attributes, then they execute the random walk over the transition matrix generated by the augmented graph. This leads them to find k groups of semantically close nodes. To measure the clustering from a structural point of view, they use the density of edges within the clusters.

The approach proposed in this work is a community detection framework which integrates the two dimensions that describe a social network. Here, the process takes into account the very different nature of these dimensions; each dimension has an adapted representation, and we preserve also adapted treatments and optimization criteria: graph techniques for the graph, data mining techniques for the attributes, according to their nature. Our main contribution intervenes *after* the clustering step, and considers the resulting partitions of these adapted treatments as an input.

III. Problem Definition

In attributed graphs [14], the structural variable is represented as a graph. and the compositional variable describes each actor as a vector of features, a bag of words or images for example. Due to the differences in their representation these variables cannot be compared directly.

Our basic idea is to use clusterings produced for each one of the variables, providing adapted though very different knowledge about the networks and its actors. Our problem is then: how to use that knowledge in an integrated way? And how to offer opportunities of control on this integration without replaying the costly clustering step? We need first to represent both dimensions within the same space in order to manipulate them.

A. Notation and definitions

Let $G(V, E, F^*)$ be an attributed graph where V and E are the set of nodes and edges respectively and the composition variable represented by F^*, is defined as $F^* : V \to \mathbb{R}^r$. Let $f_G : G(V, E, F^*) \to \mathbf{C}_G$ be a function that finds a partition $\mathbf{C}_G$ of the nodes according to G and let $f_{F^*} : G(V, E, F^*) \to \mathbf{C}_{F^*}$ is a function that finds a partition $\mathbf{C}_{F^*}$ of the nodes according to the compositional information. Without loss of generality we define these partitions $\mathbf{C}_G$ and $\mathbf{C}_{F^*}$ as affiliation matrices.

B. Partitioning the nodes according to each dimension

The functions f_G and f_{F^*} produce a partition of the graph G and the compositional information respectively. Each function has to be designed to optimize a specific quality index. Typical quality indexes for the f_G function are based on the connectivity and structural configuration of the graph, for example those presented by Brandes et al. [15]. On the other hand, function f_{F^*} is defined over the features space of the nodes. These features are typically represented as vectors for which the distance may be measured, among others, by the cosine or the Euclidean distance.

Since the partitions in each case are generated using the adapted measures, they cannot be directly compared.

C. Comparing the structural and composition partitions

As mentioned before, partitions $\mathbf{C}_G$ and $\mathbf{C}_{F^*}$ are expressed as affiliation matrices of size $|V| \times m$ and $|V| \times r$ respectively, where m is the number of structural groups and r is the number of compositional groups.

All n nodes have been allocated twice (in each partition). A contingency matrix $\mathcal{C}$, as presented in Table I, is a matrix where each entry n_{ij} represents the number of common nodes between groups $i \in \mathbf{C}_G$ and $j \in \mathbf{C}_{F^*}$.

		Paritition $\mathbf{C}_{F^*}$				
	Class	v_1	v_2	...	v_r	Sums
	u_1	n_{11}	n_{12}	...	n_{1r}	$n_{1\cdot}$
	u_2	n_{21}	n_{22}	...	n_{2r}	$n_{2\cdot}$
Partition $\mathbf{C}_G$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
	u_m	n_{m1}	n_{m2}	...	n_{mr}	$n_{m\cdot}$
	Sums	$n_{\cdot 1}$	$n_{\cdot 2}$	...	$n_{\cdot r}$	n

TABLE I
CONTINGENCY MATRIX OF THE AGREEMENTS OF TWO PARTITIONS $\mathbf{C}_G$ AND $\mathbf{C}_{F^*}$

The matrix $\mathcal{C}$ can be calculated as:

$$\mathcal{C} = \mathbf{C}_G^T \mathbf{C}_{F^*} \tag{1}$$

In our previous work, we have demonstrated that the effect of modifying the weights in the graph according to some attribute similarity, results in a division of the structural partition according to the compositional groups. The process tends to group homogeneous attributed nodes while keeping some of the connectivity properties, but as a result, the density is lower than in exclusive structural clustering (idem for the entropy compared to SOM groups with attributes only).

To quantify this trade-off between two partitions, we use ARI on the contingency matrix. The Adjusted Rand Index proposed by Hubert and Arabie [16] gives us a notion of distance between two partitions.

IV. INTEGRATED PARTITION

The community detection process proposed in this work takes advantage of the configuration of the partitions generated from each type of variable in the network.

The matrix generated by equation 1 represents the relationships between 2 partitions; the rows of this matrix represent the groups on the structural partition while the columns represent the groups on the compositional partition. The idea of our algorithm is to manipulate the configuration of the rows of the matrix $\mathcal{C}$ in order to decompose the structural partition according to the compositional partition.

Data: $\mathbf{C}_G$, $\mathbf{C}_{F^*}$
Result: $\mathbf{C}^*$
1 $\mathcal{C}^* \leftarrow \emptyset$;
2 $\mathcal{C} \leftarrow \mathbf{C}_G^T \mathbf{C}_{F^*}$;
3 $i \leftarrow 0$;
4 **while** $i < rows(\mathcal{C})$ **do**
5 $\quad \mathbf{C}_i \leftarrow$ row_process $(\mathcal{C}_{i\cdot})$;
6 $\quad \mathcal{C}^* \leftarrow \mathcal{C}^* \oplus \mathbf{C}_i$;
7 $\quad i \leftarrow i + 1$;
8 **end**
9 $\mathbf{C}^* \leftarrow$ rebuild_partition $(\mathcal{C}^*)$;
10 **return** $\mathbf{C}^*$

Algorithm 1: Row manipulation community detection algorithm

Algorithm 1 outlines the row manipulation algorithm. The algorithm starts by generating the contingency matrix (line 2). Then each row of this matrix is processed (line 5) to evaluate the compositional characteristics of the structural community represented by that row. This process produces a matrix $\mathbf{C}_i$ of $s \times r$, where $1 \leq s \leq r$ is the number of subgroups that can be created from the $i-$th structural community. This matrix $\mathbf{C}_i$ is concatenated (line 6) to the matrix $\mathcal{C}^*$ that contains the new configuration of the partition.

The division of each row can be made according to different criteria depending only on the configuration of the contingency matrix, specially on the columns of the matrix which represent the composition partition.

For evaluating the line 5 in the algorithm in this paper we propose two approaches: the naive one and the variance based one.

- **Naive approach:** in this approach, for each row $\mathcal{C}_i$, if $\mathcal{C}_{ij} > 0$ then the nodes belonging to the structural group i and the composition group j will form a community, i.e., for each entry greater than 0 in the contingency matrix there will be a community in the new partition.
- **Variance based approach:** in this approach, for each element j of $\mathcal{C}_i$, if $\frac{(\mathcal{C}_{ij} - \mu_i)}{\sigma_i} \geq 1$ that element will be a new community. Here μ_i and σ_i are the average and the standard deviation of the row i respectively. Thus the structural communities are splitted according to the representativity of the compositional categories, i.e., those with greater positive variance.

These methods allow us to evaluate the structural groups under the light of the compositional variable and to decompose them if the condition is fulfilled.

A. Algorithm example

For this example we use a small social network composed of 24 nodes and 63 edges. This network has been divided into four structural groups and three compositional groups.

Each node in the social network belongs to one of three compositional categories. The first step is hence to construct the contingency matrix $\mathcal{C}$ which uses the structural and composition partitions, $\mathbf{C}_G$ and $\mathbf{C}_{F^*}$ respectively. This matrix is presented in Table II.

	$\mathbf{C}_{F^*}$		
$\mathbf{C}_G$	3	3	0
	2	3	1
	3	2	1
	0	0	6

TABLE II
CONTINGENCY MATRIX FOR THE SOCIAL NETWORK EXAMPLE

The ARI of these partitions is 0.1998. Each entry i, j of the matrix shows the agreements between partitions. The next step is the processing of each row of $\mathcal{C}$. The criterion used in this example is the basic one in which the group is subdivided into groups representing each composition community.

Following this naive approach the new partition is composed of 9 groups as presented in Table III. The ARI computed from this matrix is 0.4232, meaning that both partitions are more similar than before. The next step is to rebuild the affiliation matrix using $\mathcal{C}^*$ and the original affiliation matrices (algorithm's line 9.)

Table IV presents a summary of the results for this basic example. We use three measures to compare the final results. First we use the ARI to measure the similarity between partitions, second, the density, measuring the partition from a structural partition perspective and last, the entropy, which measures the order, i.e., the attribute homogeneity of the groups of the final partition.

The ARI between the new partition and the original composition partition shows that the distance between them has been reduced, which means that the new groups are more aligned with the compositional variable. This is also observed on the entropy value, which drops to 0 indicating that each new group is composed of nodes with similar attributes; however, the cost of the reduction of the entropy is the loss of the density, which implies a reduction of the quality of the partition in structural terms.

		$\mathbf{C}_{F^*}$		
$\mathbf{C}^*$	$\mathbf{C}_0$	3	0	0
		0	3	0
	$\mathbf{C}_1$	2	0	0
		0	3	0
		0	0	1
	$\mathbf{C}_2$	3	0	0
		0	2	0
		0	0	1
	$\mathbf{C}_3$	0	0	6

TABLE III
RESULTING CONTINGENCY MATRIX ONCE THE STRUCTURAL COMMUNITIES HAVE BEEN MODIFIED

When the variance criterion is used, the density of the partition is greater than the one of the naive case, hence the nodes within groups are still well connected. The groups are also more similar than in the pure structural partition (fortunately), providing the desired effect.

Partition	Groups	ARI (w.r.t $\mathbf{C}_{F^*}$)	Density	Entropy
$\mathbf{C}_G$	4	0.1998	0.9365	4.9467
$\mathbf{C}^*_{\text{Naïve}}$	9	0.4232	0.4444	0
$\mathbf{C}^*_{\text{Variance}}$	6	0.3229	0.6508	2.6593

TABLE IV
SUMMARY OF RESULTS OF THE ALGORITHM FOR THE EXAMPLE SOCIAL NETWORK

V. EXPERIMENTS AND RESULTS

To test the algorithm we have performed experiments with two real-world social networks, one from Facebook and another from DBLP representing co-authorship relations. Table V describes the configuration of the social networks.

Network	Nodes	Edges	Composition information
Facebook	334	5394	Information about the academic and professional competences of each actor
DBLP	10000	65734	Information about the topics, knowledge fields and intellectual production volume of each author.

TABLE V
DESCRIPTION OF THE SOCIAL NETWORKS USED DURING THE EXPERIMENTS

From each social network the two partitions have been derived. The structural partition $\mathbf{C}_G$ has been generated with the Louvain method [8] which is designed to optimize the modularity. The compositional partition $\mathbf{C}_{F^*}$ has been created with Self-Organizing Maps – SOM [7] which optimizes a distance measure such as the Euclidean distance.

A. Experiments with the Facebook network

In this network, the structural partition contains 6 groups corresponding to the following categories: (1) Math and science, (2) Business administration, (3) Law, (4) Social sciences, (5) Software eng., (6) Other eng. fields and (7) Arts. These categories are defined according to the areas of expertise of each actor in the network. The 6 structural groups are: (1) Group of former coworkers in a research project, (2) Family and family friends, (3) Group of students and researchers from the university where the network's owner made its undergraduate and graduate studies, (4) Group of former students from the school and high school, (5) Group of people from a consulting firm where the network's owner worked before starting its PhD, (6) Group of people known during PhD studies.

For this study, we have manually labeled the groups resulting from the clustering tasks. As we can see, the categories reveal a real relevance, and comfort our double idea of applying adapted process to specific data, and perform the integration on valuable partitions. The contingency matrix $\mathcal{C}_{FB}$ for these partitions is presented in Table VI.

The ARI of this contingency matrix is 0.0189. We apply algorithm 1 with both the naive and the variance separation methods. In the first case each structural group is divided into the number of composition groups with size greater than zero, producing in this case, 40 groups.

	$\mathbf{C}_{F*}$						
$\mathbf{C}_G$	9	10	2	1	2	0	3
	20	1	4	7	10	1	4
	15	8	0	10	21	12	2
	8	8	1	11	14	5	3
	11	9	1	9	25	7	5
	13	6	1	15	17	17	6

TABLE VI
CONTINGENCY MATRIX FOR THE PARTITIONS OF THE FACEBOOK NETWORK

Partition	Groups	ARI (w.r.t $\mathbf{C}_{F*}$)	Density	Entropy
$\mathbf{C}_G$	6	0.0189	0.9718	15.1475
$\mathbf{C}^*_{\text{naive}}$	40	0.2819	0.1294	0
$\mathbf{C}^*_{\text{Variance}}$	12	0.1063	0.6511	4.5502

TABLE VII
SUMMARY OF RESULTS OF THE ALGORITHM FOR THE FACEBOOK SOCIAL NETWORK

The second approach extracts only the more representative compositional communities included in the structural group. Algorithm results are presented in Table VII. The original structural partition is composed by 6 groups and it has the best density value but it has also the worst entropy for this network.

Using the variance approach we obtain 12 communities. In this case, the obtained density value is less than the optimal value; this is due to the division of the structural groups. Although the entropy is greater than 0, it still represents 30% of the reference value (pure compositional partition entropy), gaining here on the similarity of the nodes without destroying all the community structure.

The variance approach produces interesting results between two extreme cases: on the one hand a partition produced with solely structural criteria i.e., maximizing the modularity in this case, on the second hand the naive approach, that minimizes the entropy of the partition, but at the expense of density.

Figure 1 shows the distribution of composition categories for the pure structural partition. Note that each community is heterogeneous in terms of these composition categories. Friendship relationships in this Facebook sample seem not to be related to professional competencies.

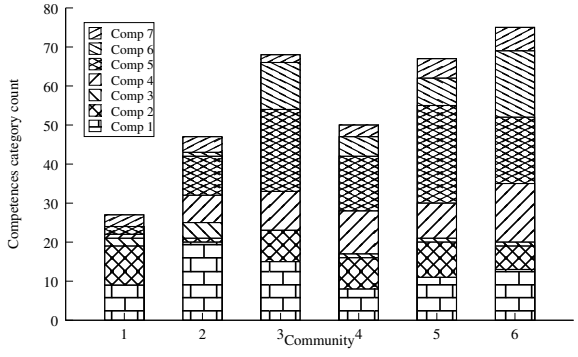

Fig. 1. Distribution of competences for the partition $\mathbf{C}_G$

With the integration of variables we obtain more and smaller groups, but these groups are more homogeneous regarding the profiles, as presented in Figure 2. Some groups represent one or a few categories. But if we focus on the competency 5 (Software engineering), we find that it is well spread in almost each group with an important number of members. This knowledge would have been discarded with pure structural techniques.

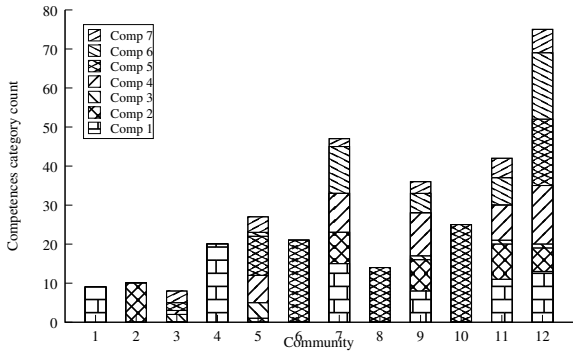

Fig. 2. Distribution of competences for the partition $\mathbf{C}^*_{Variance}$

B. Experiments with the DBLP network

Each actor is described first in terms of her type: highly prolific, prolific and little prolific (according to its publications). Secondly each actor belongs to one of the 99 clusters of topics, given in the dataset, and used in [14].

In this case the structural partition contains 838 groups and the compositional partition contains 53 groups. Therefore the contingency matrix $\mathcal{C}_{DBLP}$ contain 44414 entries. 2496 are non zero. Table VIII shows the summary of the results of the algorithm for the DBLP network.

Partition	Groups	ARI (w.r.t $\mathbf{C}_{F*}$)	Density	Entropy
$\mathbf{C}_G$	838	$3.98 \times 10{-}4$	0.8353	665.6368
$\mathbf{C}^*_{\text{naive}}$	2496	56.4×10^{-4}	0.2079	0
$\mathbf{C}^*_{\text{Variance-I}}$	1662	49.7×10^{-4}	0.2638	6.5743
$\mathbf{C}^*_{\text{Variance-II}}$	893	4.39×10^{-4}	0.8304	523.9579

TABLE VIII
SUMMARY OF RESULTS OF THE ALGORITHM FOR THE DBLP CO-AUTHORSHIP SOCIAL NETWORK

The first row shows the partition created with only the structural information. The ARI indicates that the configuration of this partition is far from the configuration of the composition partition. Additionally this partition has the best density and worst entropy of all the configurations. The second partition, generated using the naive approach, produces 2496 groups, zero entropy and the lowest density. However, this configuration has the highest (possible for this configuration) ARI value, suggesting a more similar configuration to the composition partition.

We use the variance approach to mix the variable types of the social network. The results are on the third row. In this case ARI is not a significant different from the naive approach, which means that the configuration of these partitions are similar. The density of this partition is very low and very close to the naive partition. This can be explained by analyzing

first $\mathbf{C}_G$; about the 60% of groups of the structural partition contains only one and the 20% contains two nodes. Following the variance rule, most of the groups with only two elements are broken into singleton clusters, i.e., a two-nodes structural group will be decomposed into two communities of only one node.

To overcome this issue we use an additional constraint over the variance method consisting in taking only non-zero values to calculate the average and the standard deviation. The results of this modification are reported on the fourth row. In this case we obtain a new configuration with a high density. However it is important to note that the entropy level, a little lower than pure structural partition, is still high. We face here again the trade-off between the density and the entropy. The strategies deployed here to manipulate the rows of the contingency matrix are efficient on Facebook dataset, but not on DBLP. This work does not go very deep in the design of strategies. We feel here that the analysis of the networks should guide the design of the integration process.

VI. Conclusion and future work

We have presented in this paper a novel approach to the community detection problem that integrates the two variables contained in a social network. Each one of these variables is represented as a partition, one from the structure and the other from the composition information. This approach takes advantage of the summarization of the two variables of the social network made with the contingency matrix. This matrix contains the agreements between two partitions issued from different types of information, making them comparable. The rows of the contingency matrix represent the groups of the structural partition while the columns represent the groups of the compositional partition; therefore manipulating the rows in function of the columns yields to a new partition configuration where the structural sub-groups are relevant in the compositional space.

We proposed two ways to divide the structural communities. First a naive method that converts every non-zero entry of the contingency matrix into a new community: these communities are composed of nodes of one type only. Second a method based on the variance of each composition category composing the structural community: a row is split to keep the compositional groups that contribute the most to the variance. This last criterion allows us to decompose the structural partition in terms of the composition variable while keeping a good trade-off between the density and entropy.

Results on real-world networks, when using the variance method, show improvements in both measures, producing partition of well connected and similar nodes. The results could be improved by changing the row manipulation method in such a way the final density can be improved.

One important advantage of this method is the ability to take two partitions already built with existing, suitable methods for each dimensions. This is especially interesting with large scale networks, but also for the analyst who has to try different strategies of analysis in reasonable computional time. This use case illustrates a second advantage which is the opportunity to control the integration phase. Finally this method is extensible.

Future work includes the study of the row division method to take into account the distribution of the composition partition and to select which and how structural groups to divide. Additionally a way to fusion two or more structural groups would allow us to explore different configurations of the structural groups. But before, we have to investigate to understand the difference of performance on our datasets. The comparison with other existing methods on synthetic networks would of course be valuable.

References

[1] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. No. 8 in Structural Analysis in the Social Science, Cambridge University Press, 1994.

[2] J. Neville, M. Adler, and D. D. Jensen, "Clustering relational data using attribute and link information," in *Proceedings of the Workshop on Text Mining and Link Analysis, Eighteenth International Joint Conference on Artificial Intelligence*, (Acapulco, Mexico), 2003.

[3] K. Steinhaeuser and N. Chawla, "Community detection in a large real-world social network," in *Social Computing, Behavioral Modeling, and Prediction* (H. Liu, J. Salerno, and M. Young, eds.), pp. 168–175, Springer US, 2008.

[4] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review. E, Statistical Nonliner and Soft Matter Physics*, vol. 69, p. 026113, Feb 2004.

[5] J. D. Cruz, C. Bothorel, and F. Poulet, "Point of view based clustering of socio-semantic network," in *EGC* (A. Khenchaf and P. Poncelet, eds.), vol. RNTI-E-20 of *Revue des Nouvelles Technologies de l'Information*, pp. 309–310, Hermann-Éditions, 2011.

[6] J. D. Cruz, C. Bothorel, and F. Poulet, "Semantic clustering of social networks using points of view," in *CORIA* (G. Pasi and P. Bellot, eds.), pp. 175–182, Éditions Universitaires d'Avignon, 2011.

[7] T. Kohonen, *Self-Organizing Maps*. Springer, 1997.

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

[9] N. Villa-Vialaneix, M. Olteanu, and C. Cierco-Ayrolles, "Carte auto-organisatrice pour graphes étiquetés.," in *Atelier Fouilles de Grands Graphes (FGG) - EGC'2013*, (Toulouse, France), p. Article numéro 4, Jan. 2013.

[10] D. Combe, C. Largeron, E. Egyed-Zsigmond, and M. Géry, "Getting clusters from structure data and attribute data," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, (Istanbul, Turkey), pp. 731–733, Aug. 2012.

[11] D. Combe, C. Largeron, E. Egyed-Zsigmond, and M. Géry, "Combining relations and text in scientific network clustering," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, (Istanbul, Turkey), pp. 1280–1285, Aug. 2012.

[12] H. Li, Z. Nie, W.-C. Lee, L. Giles, and J.-R. Wen, "Scalable community discovery on textual data with relations," in *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, (New York, NY, USA), pp. 1203–1212, ACM, 2008.

[13] R. Ge, M. Ester, B. J. Gao, Z. Hu, B. Bhattacharya, and B. Ben-Moshe, "Joint cluster analysis of attribute data and relationship data: The connected k-center problem, algorithms and applications," *ACM Trans. Knowl. Discov. Data*, vol. 2, pp. 7:1–7:35, July 2008.

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

[15] U. Brandes, M. Gaetler, and D. Wagner, "Engineering graph clustering: Models and experimental evaluation," *Journal of Experimental Algorithmics*, vol. 12, pp. 1–26, 2008.

[16] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, pp. 193–218, 1985. 10.1007/BF01908075.