IT UNIVERSITY OF COPENHAGEN

A Practical and Critical Look at the Problem of Community Discovery in Multi-layer Networks

by

Obaida Hanteer

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the
Digital Design Department

May 2020

Declaration of Authorship

I, OBAIDA HANTEER, declare that this thesis titled, 'A Practical and Critical Look at the Problem of Community Discovery in Multi-layer Networks' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

"Give me insights, not numbers"

Richard Hamming

IT UNIVERSITY OF COPENHAGEN

Abstract

Digital Design Department

Doctor of Philosophy

by Obaida Hanteer

A core task in social network analysis is to recover communities; that is, to meaningfully partition the network into groups of nodes such that nodes within a group share common behaviors, traits or properties. Driven by the motivation that different types of relationships/interactions among social actors do not exist in isolation but they might depend on each other such that one can lead to the other one, advances on social network analysis suggested that providing meaningful insights about networks should consider these inter-dependencies to avoid misleading or incomplete findings. This leads to the emergence of multi-layer network analysis; that is, an approach where a single network models different types of interactions among a set of actors. This generalization proposed by the multi-layer framework for modelling social networks has introduced new challenges to the community detection problem. There has been a general tendency by researchers to cope with this complexity by extending some of the already existent methods used with mono-relational networks. While the logic behind these methods seems to be sound, I claim that the patterns they identify in multi-layer networks are not thoughtfully investigated yet, which makes their definition of multi-layer communities quite vague and hence providing qualitative interpretation for the outputs becomes not an easy task, especially with real-world data. In my thesis, I investigate community detection from a practical perspective while being equally distant from the different disciplines that researched this topic. I adopt a critical stance towards community detection in multi-layer networks in some cases when the methods used seem to inherit their success and popularity from their original mono-relational implementations rather than their ad-hoc competency in recovering complex patterns in multi-layer networks. I propose different models for multi-layer communities and challenge popular community detection methods to recover these models. I conclude by pointing out to limitations in these methods and the available layer-coupling mechanisms to recover the ground truth communities in multi-layer networks.

IT UNIVERSITY OF COPENHAGEN

Abstrakt

Digital Design Department

Doctor of Philosophy

by Obaida Hanteer

En kerneopgave i analyse af det sociale netværk er at gendanne samfund; det vil sige at meningsfuldt opdele netværket i grupper af noder, så noder inden for en gruppe deler fælles opførsel, træk eller egenskaber. Drevet af motivationen om, at forskellige typer af relationer / interaktioner mellem sociale aktører ikke eksisterer isoleret, men de kan være afhængige af hinanden, så den ene kan føre til den anden, antydede fremskridt med analyse af det sociale netværk, at det at give meningsfuld indsigt om netværk bør overvejes disse indbyrdes afhængigheder for at undgå vildledende eller ufuldstændige fund. Dette fører til fremkomsten af flerlags netværksanalyse; det vil sige en tilgang, hvor et enkelt netværk modellerer forskellige typer interaktioner mellem et sæt aktører. Denne generalisering, der er foreslået af flerlagsrammen til modellering af sociale netværk, har indført nye udfordringer for samfundsdetekteringsproblemet. Der har været en generel tendens fra forskere til at tackle denne kompleksitet ved at udvide nogle af de allerede eksisterende metoder, der bruges med mono-relationelle netværk. Selvom logikken bag disse metoder synes at være forsvarlig, hævder jeg, at de mønstre, de identificerer i multilagsnetværk, ikke er eftertænksomt undersøgt endnu, hvilket gør deres definition af flerskiktssamfund ganske vage og dermed at give kvalitativ fortolkning af outputene ikke bliver en nem opgave, især med data fra den virkelige verden. I min afhandling undersøger jeg samfundsdetektion fra et praktisk perspektiv, mens jeg er lige langt væk fra de forskellige discipliner, der undersøgte dette emne. Jeg indtager en kritisk holdning til detektion af lokalsamfund i flere lags netværk i nogle tilfælde, når de anvendte metoder ser ud til at arve deres succes og popularitet fra deres oprindelige mono-relationelle implementeringer snarere end deres ad hoc-kompetence til at genvinde komplekse mønstre i flerlags-netværk . Jeg foreslår forskellige modeller til flerlagssamfund og udfordrer populære fællesskabsdetektionsmetoder til at gendanne disse modeller. Jeg afslutter med at pege på begrænsninger i disse metoder og de tilgængelige lag-koblingsmekanismer til at gendanne sandhedssamfundene i flere lag i netværket.

Research Context

This research is supported by the VIRT-EU project funded by the European Union's Horizon 2020 research and innovation program under grant agreement No 727040.

VIRT-EU project aimed to analyze and map the ethical practices of European hardware and software entrepreneurs, maker and hacker spaces, and community innovators. The expected outcome from that project has been discussed to be: **(i)** understanding how IoT innovators enact ethics as they design future devices, **(ii)** generate a new framework for Privacy, Ethical and Social Impact Assessment (PESIA) and **(iii)** develop tools to support ethical reflection and self-assessment as part the design and development process for IoT technologies.

The research conducted by VIRT-EU qualitative team provided a seed dataset of European actors who were considered particularly important in the IoT field. Starting from that list, the project had access to Twitter data about actors in that list, their followers and those they follow. This data included the following/follower network, the reply network, the retweet network and the mentioning network. In this context, whether or not the development ethics in IoT shape the formation of online communities seemed to be a very relevant question for the project. With the availability of multiple networks among the same set of actors, a natural choice for a social network analysis tool to answer that question seemed to be multi-layer community discovery. The multi-layer network to analyze in this context is the one that considers all the aforementioned Twitter networks and the dependencies among them. In that regard, there has been many challenging questions, most of which inspired the content of this thesis, which the literature in multi-layer network analysis did not provide clear answers for. First, which multi-layer community discovery to use? The literature in community discovery in multi-layer networks provides a plethora of such methods without any guidance on the differences among these methods . Second, which layers to include in the analysis? is it that the more information we have the more complete the findings are? what if some of these layers are just noise? and is there a way to quantify this noise in multi-layer networks?. Third, what is it actually that we are looking for when we use multi-layer community discovery? In other words, what is the definition of a multi-layer community from a conceptual perspective and does the output of these methods map to that definition, if exists?. In many cases, the complicated mathematical formulations of some of these methods made it quite challenging to give the patterns they recover in multi-layer networks a conceptual definition that can be communicated to the VIRT-EU qualitative team. These challenges have mainly inspired the ideas and the discussions proposed in this thesis.

Acknowledgements

To my primary supervisor – Luca Rossi. If there is a prize given to PhD supervisors, I would definitely award him the "coolest supervisor ever" prize. Despite our different backgrounds, it has always been very easy to talk to him and to collaborate with him. I have definitely learned a lot from being supervised by Luca, and I owe him my flexibility in questioning math, my ex-God, when applied to social contexts.

To my secondary supervisor – Irina Skloveski. The one I sought whenever I needed advice on how to proceed and survive the tough PhD experience. Despite her busy schedule, she always found time to help whenever I needed her, even going as far as to deal with my visa and passport problems.

To the one I owe this great opportunity of doing my PhD in Denmark, my co-author in most of my publications and the biggest influence when it comes to reasoning my research questions – Matteo Magniani. I knew him when I applied for a PhD position in his group and I was not the selected candidate at the end. As much as I was upset at the time, I am very grateful for him now as he strongly recommended me for the position I had at ITU.

To Michele Coscia – a big influence in my research. He was always generous with his time and he revised my papers before submission. During assisting him in the Network Analysis course given at ITU, I have learned a lot about network science and how to teach network science.

To the colleague who made an effort to learn how to greet me in my language. The one who always started her emails with "Marhaban Obaida. How are you doing?". The one who always checked on me to get reassured I was doing well with my PhD – Rachel Douglas-Jones.

To the best Italian hosts and best friends in Rome, Leonardo and Irene, who hosted me for one long stressful month when I decided to write my thesis in the eternal city of Rome. They gave me a lot of love and care, and they were the best suppliers of high-end Italian food which is all a researcher needs to contribute to the field.

To my ex-office mate at ITU, current most considerate roommate and "agapi mou" – Maria Karampela. Coffee/tea times with her were all I needed to feel in peace and kick off the day with great energy.

To the best and most talented hairstylist who truly loves his job and inspires me to love mine too. A walking positive energy in the shape of a human being and a true friend. I can not be more grateful for knowing him – Mike Toro.

To my proof-reader – Keith Lia. The one who had to change every single appearance of the word "multilayer" in my thesis into "multi-layer". The one to blame for any misspelling or unclear rephrasing in my thesis.

To my great friend, Muhammed Allou, who has always been nice, caring, respectful and friendly to me. A true inspiration in his maturity and ability to be independent, and to achieve so much at a very young age. The one who offered me free drinks every time I passed by the bar where he works. Don't we all need such a "True" friend?

To everyone who, when they knew I come from Syria, did not ask me whether or not I am a refugee, whether or not I drink alcohol, whether or not I eat pork, about my opinions in regards to the political situation in Syria and my guesses for how Iran is planning to use its nuclear power.

To Copenhagen – though you have not always been an easy place to adapt to, I have learned a lot and grew up a lot living here, and I owe you the growth, the nice experiences and the amazing people I have met while living here.

Last but not least, I want to give myself some credit for surviving the tough and lonely PhD experience and for being independent enough in a very inter-disciplinary setting. I feel achieved and proud of my work and contribution to the field at the end of this journey.

Contents

List of Figures

Dedicated To Myself

Chapter 1

Introduction to Social Network Analysis

Social network analysis investigates properties, patterns and dynamics in social networks that are represented as nodes (corresponding to social actors) and links between these nodes (corresponding to social relations or interactions among the actors such as friendship, collaboration, or leisure). The detection of cohesive subgroups in social networks, also referred to as “community discovery”, received a great deal of attention in the literature, since these patterns formalize the very important concept of ‘social groups’ used in social sciences, particularly sociology and social psychology. This could be extended to other types of networks that are not necessarily social, for example biological networks, where community discovery helps break down large and complex networks into smaller building blocks, making it easier to understand the structure of the network and provides a better overview of the role or functionality each group plays in it. This chapter provides the reader with a brief introduction to social network analysis in Section 1.1, followed by a definition of the community discovery problem in networks in Section 1.2. Three important community recovery methods are then highlighted in section 1.3, and a discussion about community discovery evaluation is presented in Section 1.4. A general similarity analysis among different community discovery methods is reported in Section 1.5. To shed a light on the importance and relevance of community discovery in the field, the chapter concludes by discussing applications of community discovery in Section 1.6.

1.1 Background in Social Network Analysis (SNA)

SNA is a data analysis strategy that makes use of mathematical tools and models, like graph theory, to investigate structures, patterns and dynamics in social networks [1]. The term *social* comes from the fact that the use of SNA arose as a research topic in sociology and social psychology, going back to the early 20th century where some attempts were made to quantify reciprocity in social networks (i.e., the tendency to reciprocate a relationship/interaction in a system) [2]. The fundamental difference between SNA and other data analysis tools is that SNA involves the intuition that links among social actors are important [3]. Despite the term 'social', the methods and tools developed for social network analysis can be seen abstractly as graph analysis tools. That is to say that while they are useful at providing insights about social phenomena when applied to social networks, they are equally useful in other fields where networks do not necessary capture social relationships among social actors but connections of any type between a set of objects (e.g., protein-protein interaction networks in biology). Unless mentioned otherwise, I keep the focus in this thesis on the application of network analysis to social networks.

The major mathematical model used in SNA is the graph model from graph theory. A graph is a way to model pairwise relationships, interactions or flows of information among objects, usually referred to as nodes or vertices [4]. Depending on what the graph models, these nodes might refer to cities, people, organizations, biological species, airports, etc. The relationships among these nodes are usually represented as links, also called edges. These edges can be directional to model a flow of information in a specific direction, or to model non-reciprocated relationships or interactions. An edge can also be assigned a weight - that is a real value which, depending on what the edge represents in the real-world, may reflect a strength, a capacity or a size when it is positive, or a cost when it is negative. Figure 1.1 illustrates a toy graph constituted of 16 nodes and unidirectional unweighted edges among these nodes. Unless mentioned otherwise, I adopt the terms *actor* throughout this thesis to refer to the actual/physical correspondent of a node in the real-world (e.g., a person or a city), and the term *node* to refer to the mathematical representation of an actor in the graph model. I also adopt the term *relationship* to refer to the actual relationship, interaction or flow of information between two actors in the real-world, and the term *edge* to refer to the mathematical representation of a 'relationship' in the graph model. The terms 'graph' and 'network' will be used interchangeably to refer to the same thing.

As social network analysis adopted the graph model from graph theory to model social relationships, it has also made use of some graph theory concepts to explain social phenomena. For example, a node with a high degree, in graph theory terms (i.e., incident

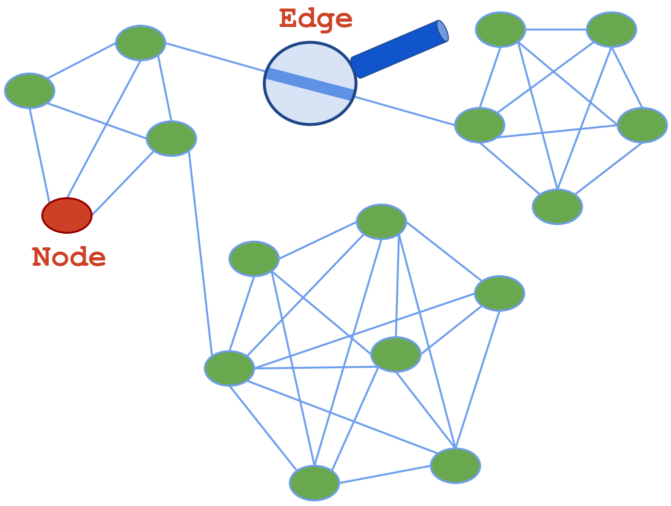


FIGURE 1.1: An illustration of a graph modeling information flows, i.e edges, among 16 nodes

to high number of edges), corresponds to a central, important or an influential actor in the network. Finding the minimum cut, in graph theory terms, corresponds to a high-level partitioning of the network actors into two more homogeneous groups of actors, and repeating that on each subgraph induced from a single partition recursively can reveal different hierarchies of a community structure in the network. Network analysts have also responded to the need of quantifying more complicated social phenomena by providing new metrics. For example, interpreting the most influential/central actor in a network simply as the node with the highest degree in the corresponding graph can be problematic in cases like the one shown in Figure 1.2. As shown in the figure, the most influential node according to the highest degree approach would be node **6** (the red one). However, while node **7** (the blue one) has a significantly lower degree in the graph, it seems to have a more central role in this network as it connects two different components of the network together, and hence a new perspective on quantifying centrality of a node is needed. The result is a new measure of centrality called the 'betweenness centrality' [5]. This concept quantifies the centrality of an actor in a network as the normalized number of shortest paths that go through the corresponding node, normalized by the number of all the shortest paths between all pairs of nodes in the graph. According to this measure, node **7** would be recognized as the most central in the graph. This is to say that social network analysis has evolved as one of the most interdisciplinary topics in the field and most of its basic concepts are the result of collaborative efforts and bouncing conversations among researchers from different disciplines including social science, mathematics, physics and more recently computer science.

Another branch of mathematics that is very present in SNA is algebra. An undirected

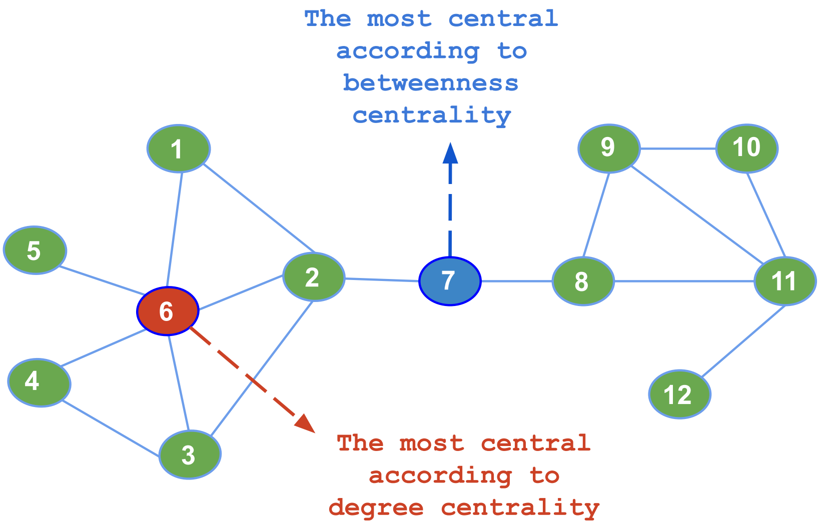
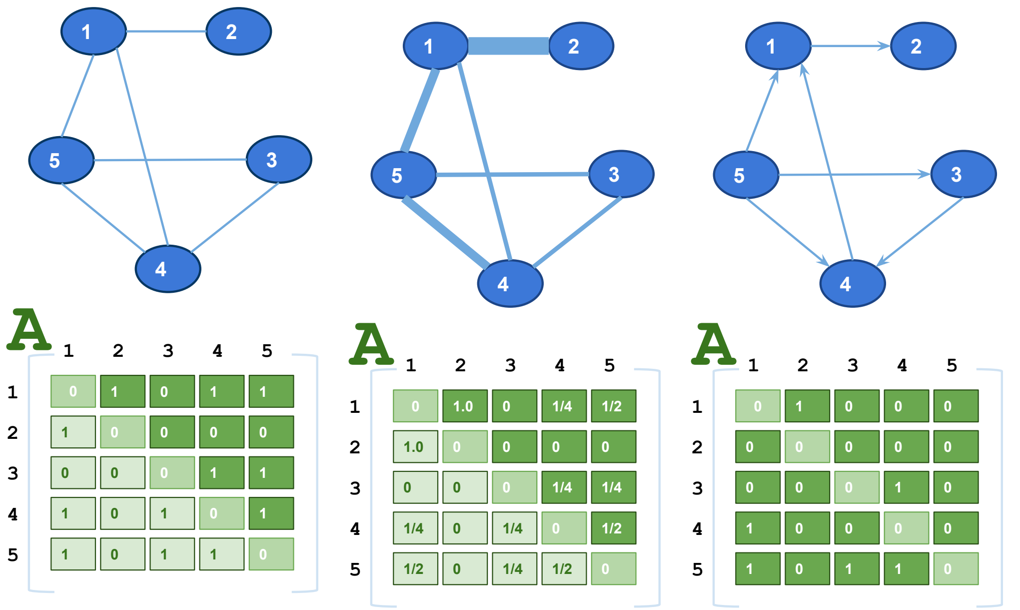


Figure 1.2: A visual illustration for the difference between degree-centrality and betweenness centrality

unweighted graph modeling the relationships among n actors, for example, can be represented as an $n \times n$ two-dimensional symmetric matrix, usually referred to as the *adjacency matrix* A, where the indices of A are actors and the entries are binary values (0s and 1s) referring to the absence and presence of edges respectively. An entry *(i, j)* has the value 1 if there is an edge between the two nodes i and j in the graph, and 0 otherwise. These entries can be substituted by edge weights in the case of a weighted graph. With directed graphs, the corresponding adjacency matrix A is not necessarily symmetric, and a non-zero value in the entry *(i, j)* refers to an edge starting at node i and ending at node j. Figure 1.3 shows different assignments for the adjacency matrix A based on the type of the graph.

Social networks can be analyzed in different scales based on the question of interest – i.e., the micro-scale, the macro-scale and the meso-scale levels [6]. At the micro level, social network analysis begins with an individual (or a small group of individuals) and snowballs as the social relationships are traced [7]. In this level, SNA can answer questions about the social settings of an individual (i.e., ego networks), or questions about small groups of two (dyads) or three individuals (triads). Such questions can be about tendencies toward reciprocity and level of transitivity, just to mention a few examples. On the meso-scale, networks answer questions concerning groups of nodes. Such questions can be whether a certain interaction triggers the formation of communities in a network, or divides the network into a set of disconnected components. On the macro level, network analysis looks at the network as a whole and provides global indicators



(A) Undirected unweighted graph (B) Undirected weighted graph (C) Directed unweighted graph

FIGURE 1.3: Adjacency matrices of three different graphs

about its properties. Figure 1.4 illustrates the relationship among these levels. The focus of this thesis is on the use of social network analysis on the meso-scale level, and more specifically the community detection task.

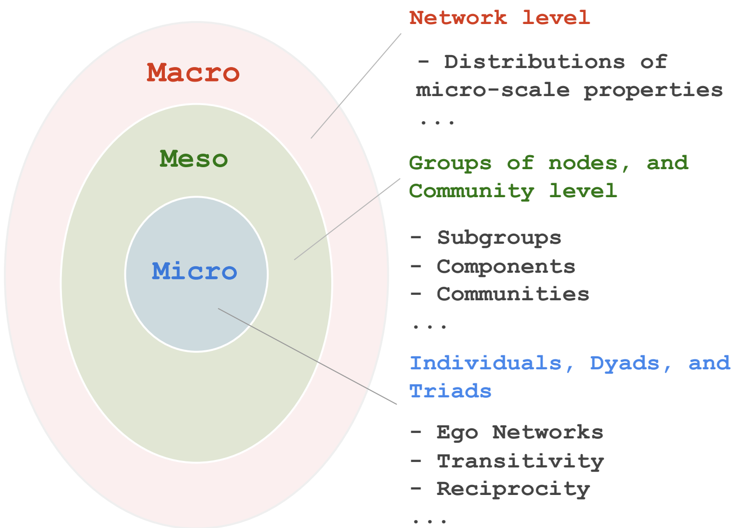


FIGURE 1.4: The three scales of analysis in social network analysis and few examples of the questions that can be asked at each level

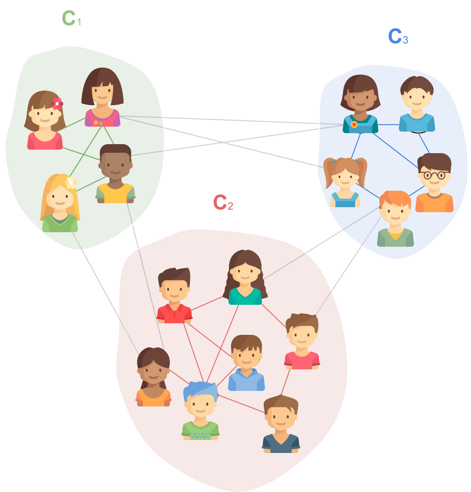
1.2 Community discovery in social networks

The notion of social groups has been one of the most studied topics in social sciences. Friedkin [8] used cohesion in social groups as a variable through which some social dynamics, like reaching a consensus in a group, can be explained. Collins, one of the most notable sociologists, stated in his review of sociological theory that the more tightly a group of individuals are connected, the more they are influenced by the group standards which results in homogeneous beliefs within the group [9]. Therefore, the identification of such groups has always been important in sociology to understand the social forces operating through members of the group and the impact of that on influencing behaviors, values and/or beliefs of social actors. With that being said, it is not surprising that one of the problems a tool that investigates social dynamics in networks is expected to investigate is the recovery of such groups, which in social network terms is usually referred to as "community discovery".

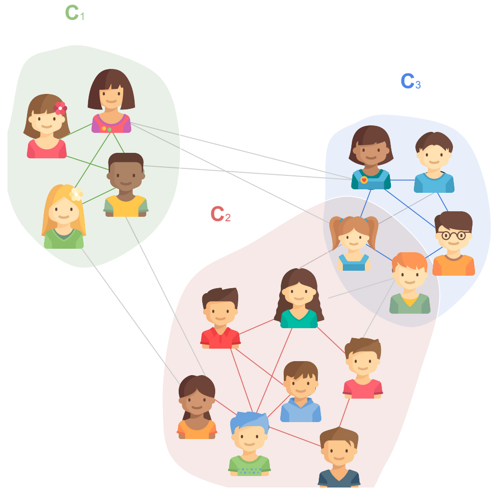
Community discovery is one of the core meso-level tasks in social network analysis. When used in social networks, it recovers network patterns that are believed to formalize the very important concept of 'social groups' in social sciences. The importance of this task extends to recovering meaningful network structures when applied to other types of networks that are not necessarily social, like brain networks, where the recovered communities might refer to different functional areas in the brain. Generally speaking, community discovery helps break down large and complex networks into smaller building blocks so it becomes easier to understand the structure of the network and provides a better overview of the role or the functionality each group plays in the network. As a result, communities, also referred to as clusters, partitions, cohesive groups or meso-scale structures, are groups of nodes that share common traits, properties or behaviors in a given context.

In mathematical terms, a community detection solution is usually represented as a clustering $\mathcal{C}$ where $\mathcal{C} = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3, ... \mathscr{C}_k\}$, and $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_k$ are disjoint or non-disjoint groups of nodes. Figure 1.5 illustrates the difference between those two types of community structures. Unless otherwise mentioned, those two types are referred as partitioning and overlapping clusterings respectively throughout this thesis. I also adopt the term *clustering* to refer to the groups of nodes recovered by a community detection task on a network, and *community* to refer to a single group of nodes in a clustering.

Given a social network whose corresponding graph is G (V, E), where V is a set of nodes and E is the set of edges among them, a community discovery task involves using the information provided by E in order to perform a clustering over V into meaningful groups of nodes. By "meaningful" I mean that there exists an evident qualitative explanation,



(A) Partitioning (disjoint) communities



(B) Overlapping (non-disjoint) communities

FIGURE 1.5: Partitioning vs overlapping communities

in addition to the mathematical one, for what brings the nodes of a recovered clustering together in this particular way.

While the notion of social groups has been extensively studied in social sciences, researchers usually used the term without giving it a precise restrictive definition [10]. On a practical level, this allowed for multiple conceptualizations into social network patterns and that in turn resulted in a plethora of community detection methods for recovering social groups in social networks. Based on my readings in the literature of community detection in social networks, and my experience with different implementations of this task, I chose to mention only a few of these community detection methods, and I refer to their implementations of the algorithms. My choice for these methods comes from their ability to recover implanted clusterings in synthetically generated networks and the availability of scalable implementations that can be efficiently used in large and complex networks. I refer the reader to [11] and [12] for more comprehensive classification of community detection methods from a theoretical and a practical perspectives respectively. Without loss of generality, I keep the focus within this thesis on community detection methods that produce partitioning clusterings.

1.3 Community discovery methods

Three of the different community detection methods stood out for their ability to recover ground truth communities in networks and the availability of scalable and efficient implementations to be used with large and complex networks. These are i) modularity maximization, ii) information flow mapping and iii) statistical inference using block modeling. In this section, I provide the reader with a high-level overview of the logic behind each of these methods. Here I choose to provide the most general mathematical formulations, when needed, to keep the focus on the intuition behind these methods rather than the digging into the interpretation of that with mathematical formulas. Throughout this thesis, the term *method* is used in the context of community detection to refer to the logic, or the mathematical tool adopted by a community detection technique, and the term *algorithm* is used to refer to a possible implementation of that method.

1.3.1 Modularity maximization

In modularity maximization methods, community detection is accomplished by partitioning the graph into sparsely connected groups of densely connected nodes. The main logic behind this is that individuals of a community tend to interact more densely with other individuals belonging to the same community and less often with external individuals in other communities.

Modularity has been firstly proposed by [13] as an index to quantify the quality of communities identified by community detection methods. It takes two parameters, a graph G(V, E) and a clustering $\mathcal{C}$ over V, and calculates the extent to which the distribution of edges with respect to $\mathcal{C}$ is far from being random. By "random" I mean that the distribution of edges within and across groups in $\mathcal{C}$ does not follow a pattern where edges are denser within these groups than across them. This can be calculated as follows:

$$Q = \frac{1}{2|E|} \sum_{\mathscr{C} \in \mathcal{C}} \sum_{(i,j) \in \mathscr{C}} \left[A_{(i,j)} - P_{(i,j)} \right] \tag{1.1}$$

where $|E|$ is the number of edges in the network. The summation is performed only over pairs of nodes that belong to the same community $\mathscr{C} \in \mathcal{C}$. $P_{(i,j)}$ is the probability of having an edge between nodes i and j in a random network that preserves some of the properties and features of G. Values of Q are in the range [-1,1] and the closer it is to 1 the more modular $\mathcal{C}$ in G is.

The value of modularity itself is not informative enough, though, and it becomes more informative when comparing two clusterings over the same network. In fact, modularity has been used in [13] to define which method, between two community detection methods, is the better one. That is, the one that returns more modular clusterings is defined as a better method than the other. Short after that review, modularity has been proposed as an objective function [14], allowing for a new class of community detection algorithms that aimed at finding a clustering over the network such that the modularity is maximized. Figure 1.6 illustrates an ordering of a clustering $\mathcal{C} = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3\}$ from less modular (left-most) to most modular (right-most) as a function of different distributions of the edges in the graph. Note that a clustering is less modular when there exists more edges across its communities and/or less edges within its communities.

Computationally speaking, modularity maximization in graphs is very expensive as it requires trying all the possible clusterings in that graph to return the one with the highest modularity. Even in small networks, maximizing modularity using this greedy approach can be infeasible. Hence, social network practitioners usually use approximation algorithms that can be scalable to large and complex networks. Among the many approximation algorithms existent in the literature for maximizing modularity, the Louvain method [15] [16] is the most popular for its accuracy and performance in recovering communities in large and complex networks.

1.3.2 Information flow mapping

This method translates community discovery in social networks into an information-theoretic problem that aims to compress the description of the flow of information in a network [17]. It uses the path traversed by a random walker in a graph as a proxy for the flow of information in the network. The goal is to provide a minimal descriptive code to that path so the description length, an information theoretic index, is minimal. The intuition here is that important structures in the networks will represent bottlenecks, where the random walker will get stuck for a while and will revisit the same nodes multiple times. Rather than giving a unique code to each node in the graph, minimizing the description length of information flow, that is the path traversed by a random walker, results in giving the unique codes to the important structures in the graph instead, so

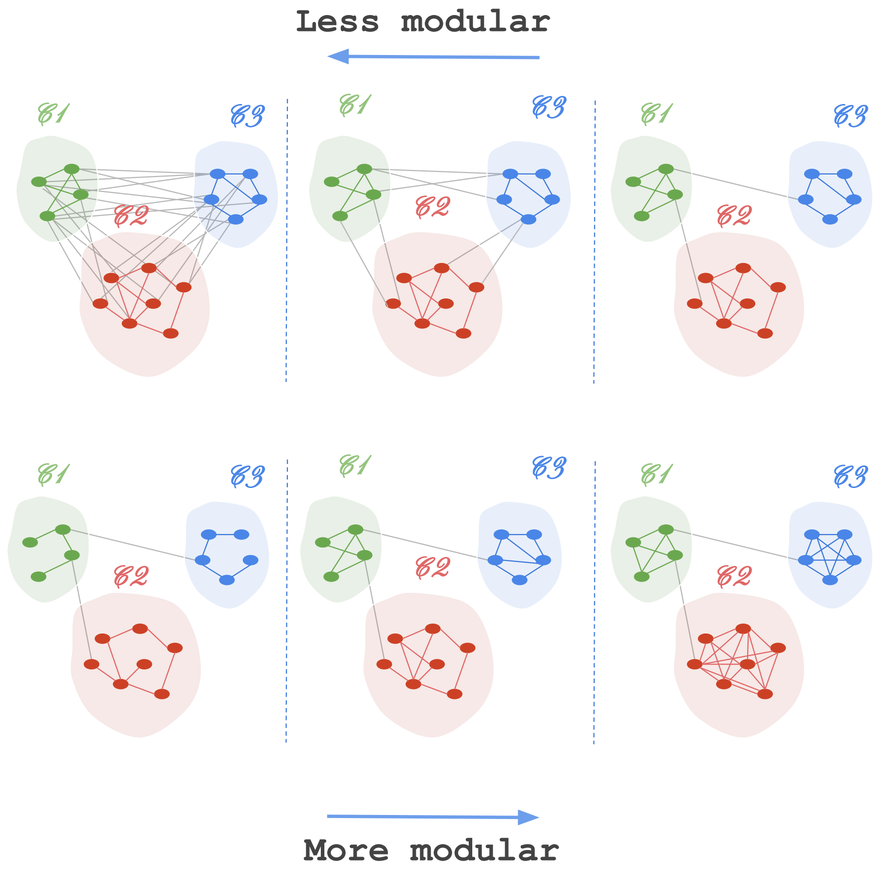


FIGURE 1.6: An ordering of a the clustering $\mathcal{C} = \{\mathscr{C}_1, \mathscr{C}_2, \mathscr{C}_3\}$ from the less modular (to the left) to the more modular (to the right) as a function of different distributions of the edges in the graph

the codes given to the nodes within these structure can be shorter, and can be reused across different structures.

Imagine that you have to explain to a friend the sequence of your last year's movements across different bars you have been to between Paris and Copenhagen. Assume that you went back and forth between these countries and visited the same bars multiple times. There are many options for you to code your path referring to the sequence of bars you have been to, but in order to save your friend's time and listening capabilities, one of the best ways to do that could be to group the important structures in your path together. That is, using a two-level coding where in the first level you refer to important structures in your map (in this case countries) and in the second level you refer to the name of the bar. This way, instead of saying:

"I went from bar$_a$ in Copenhagen to bar$_b$ in Copenhagen then to bar$_c$ in Copenhagen then bar$_a$ in Paris then bar$_b$ in Paris then bar$_a$ in Copenhagen to bar$_b$ in Copenhagen ...etc",

You would rather say:

"I went to Copenhagen then bar_a *,* bar_b *and* bar_c *then left Copenhagen to Paris and went to* bar_a *and* bar_b *then left Paris to Copenhagen and went to* bar_a *and* bar_b *..etc".*

This is actually the main logic behind minimizing the description length of the information flow as it reveals important structures in the network, which in this context correspond to communities. To have a clearer idea, Figure 1.7 illustrates a path traversed by a random walker in a graph, together with two different codings of that path based on different strategies used to name the nodes - one by giving each node a unique code, and the other one by using a two-level coding where the first level gives unique entry/exit codes to important structures in the network and the second level guarantees unique codes for the nodes within each of these structures.

To find an optimal code, the method searches for a clustering $\mathcal{C}$ of k communities that minimises the description length of a random walker. The average description length of a random walker given $\mathcal{C}$ can be calculated using the so called map equation:

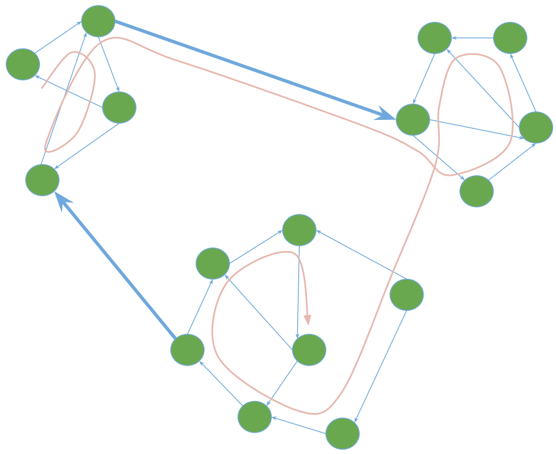
$$L(\mathcal{C}) = \underbrace{q \curvearrowright H(\mathscr{Q})}_{\text{term 1}} + \underbrace{\sum_{C \in \mathcal{C}} p^{C} \circlearrowright H(\mathcal{P}^{C})}_{\text{term 2}} \tag{1.2}$$

The first term of this equation measures the entropy of movement between modules and it is minimal if the clustering $\mathcal{C}$ captures a partitioning where an infinite random walker does the minimal amount of travelling across communities of $\mathcal{C}$. The second term captures the entropy of movement within communities of $\mathcal{C}$ and the longer these communities trap a random walker within them, the more minimal the value of this this term is.

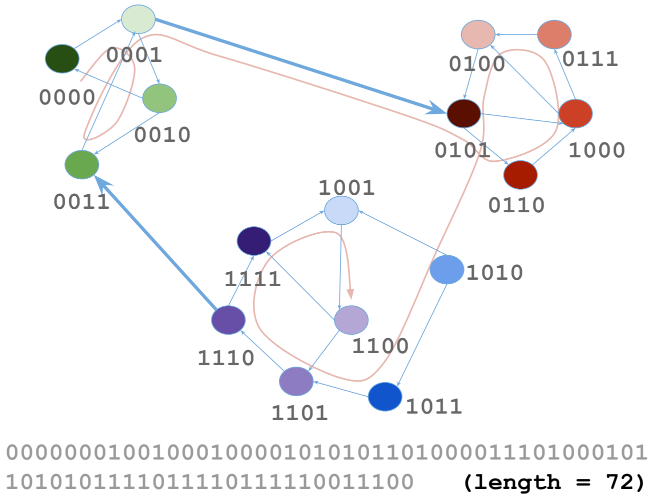
For all but the very small networks, it is infeasible to try all possible clusterings over the network and choose the one that minimizes the description length. An approximation algorithm, Infomap [18], is provided for this method and scales to large and complex networks very well.

1.3.3 Statistical inference and stochastic block models

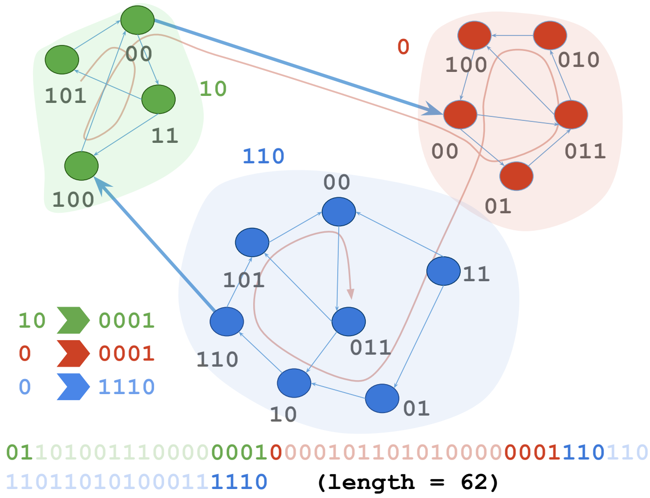
To understand the logic behind statistical inference as a method for recovering communities, we have to first introduce the concept of stochastic block model. A stochastic block model (SBM) is a simple generative process that generates graphs with the notion of communities [19]. Assuming a partition of nodes $\mathcal{C}$ into k groups (i.e., communities), a stochastic block model takes as an input a clustering $\mathcal{C}$ and a k $\times$ k matrix of edge



(A) An illustration of a path traversed by a random walker in a graph



(B) A possible coding given to the nodes of the graph



(C) A possible two-level coding given to the nodes and important structures of the graph

FIGURE 1.7: An illustration of a random walker in the graph and two possible codings for the nodes

counts e, where e_{rs} is the number of edges between two communities r and s (we will refer to this matrix as S), and produces a graph where edges are placed randomly such that the constraints of the edge counts matrix are respected. Figure 1.8 shows three possible outputs for the same input parameters, $\mathcal{C}$ and S, given to a stochastic block model.

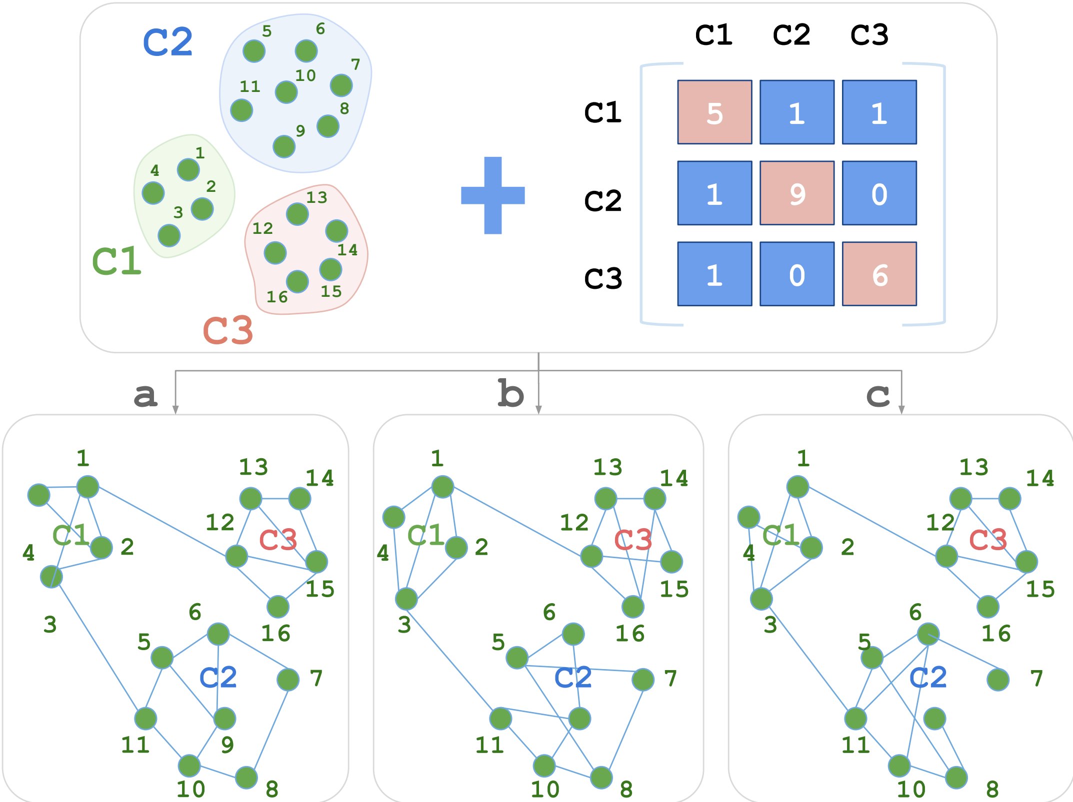


FIGURE 1.8: Three different graphs a , b and c produced by the same stochastic block model (i.e., same edge counts matrix and same clustering $\mathcal{C}$)

Statistical inference converts the community detection in a graph into finding the parameters of the stochastic block model, specifically $\mathcal{C}$, that most probably generated the graph. Given a graph G, there might exist multiple assignments of $\mathcal{C}$ and S given to a stochastic block model to generate G. Figure 1.9, for example, shows only three of all of possible assignments of $\mathcal{C}$ and S that, when given to a stochastic block model, might generate the graph G illustrated in the same figure. However, some of these assignments are better than others where "better" here means "more likely" to be the ones responsible for generating the graph G. Think of the three possible assignments shown in figure 1.9. Clearly the probability that the graph G has been generated by assignment a is higher than the same probability had assignments b and c been used. This is because assignments b and c can generate many other graphs in addition to G, while the number of graphs that can be generated using assignment a is smaller, thus more likely to be the right one.

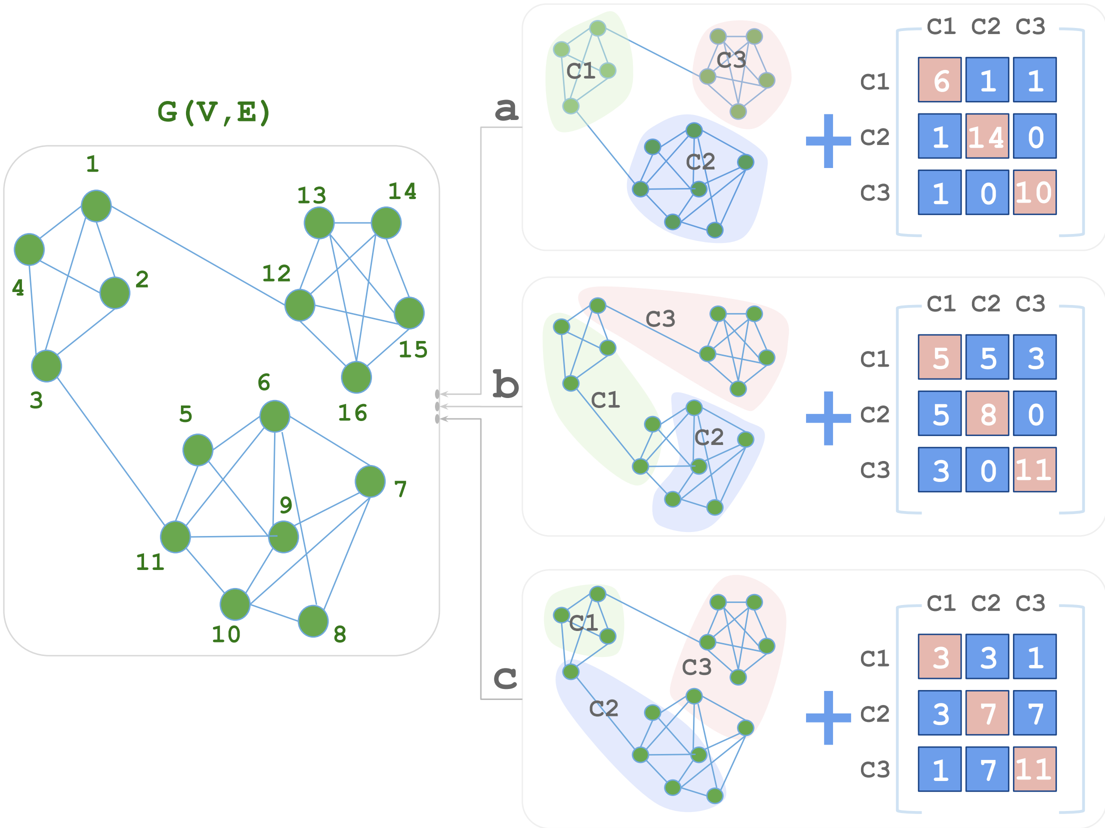


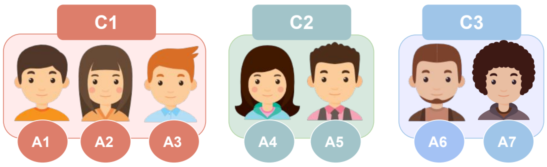
FIGURE 1.9: Three different guesses, a, b and c for a a stochastic block model parameters, $\mathcal{C}$ and S, that might have generated the graph G (on the left)

By finding the parameters of the stochastic block model that most probably generated the graph, namely $\mathcal{C}$ and S, statistical inference recovers the community structure in a network. Since trying all possible parameter assignments can be expensive, some approximation algorithms can be used to infer a solution close to the optimal one. One of these approximations is the Monte Carlo Markov Chain (MCMC) algorithm proposed by [20] which does not require an upfront knowledge about the number of communities in the network. A comparison among different inference methods is proposed by [21]) and an implementation of the reviewed methods is available in [22].

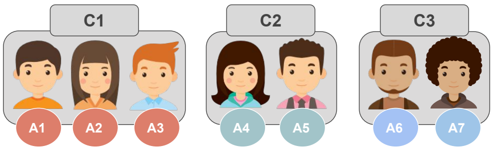
1.4 Evaluation of Community Detection

Given a social network whose corresponding graph is G (V, E), where V is a set of nodes and E is the set of edges among them, a community discovery task involves using the information provided by E in order to perform a clustering over V into meaningful groups of nodes. By "meaningful" I mean that there exists an evident qualitative explanation, in addition to the mathematical one, for what brings the nodes of a recovered clustering together in this way. A common approach to quantify this meaningfulness is by providing another clustering $\mathcal{G}$ over V which supposedly represents the correct answer, and hence is referred to as the ground truth, and then measuring the distance, using some similarity index, between $\mathcal{C}$ and $\mathcal{G}$. The ground truth clustering $\mathcal{G}$ is usually constructed based on some meta-data about the network's actors (for example, their political affiliation in a political network). This practice in quantifying the effectiveness of a community detection method assumes that the closer a clustering $\mathcal{C}$ recovered by a community detection method is to $\mathcal{G}$, the more accurate $\mathcal{C}$ is and the better the community detection method is as a result. This is inherited from the fact that synthetically generated networks that are usually used to evaluate community detection methods are generated in such a way that edges of the network are explicitly placed according to a ground truth clustering, also called implanted clustering in this case. The efficacy of a community detection method is then measured based on its ability to recover these implanted clusterings. Authors in [23], however, provide a good argument that with real-world networks, both the implanted partition (i.e the ground truth) and the actual network generator are typically unknown, which implies that providing objective evaluation for community detection methods performance with real-world networks is not an easy task.

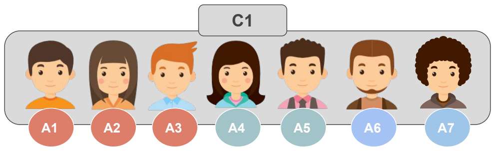
When the chosen evaluation practice is to compare the resulted clustering with a ground truth clustering, multiple normalized similarity indices have been used in the literature to quantify the level of agreement between two clusterings such that the score is 1 when the two clusterings are identical and 0 when the two clusterings are dissimilar. While the meaning of similar seems to be quite intuitive when the similarity score is 1, it is unclear what is perceived as less similar or completely dissimilar (when the score is 0) according to these similarity indices. Figure 1.10, for example, reports the values of three similarity metrics usually used in the evaluation of community discovery, namely Omega index, Normalized Mutual Information (NMI) and Adjusted Mutual Information (AMI). The values are calculated for six different clusterings when compared to the ground truth clustering shown in Figure 1.10a. As the figure shows, the different metrics not only have different values in some cases but also different sensitivities to different types of dissimilarities. We address this issue in our paper, appendix C, which has been published in the proceeding of ASONAM 2019. In that paper, we provide a taxonomy of the similarity indices commonly used for evaluating community detection solutions. We elaborate on the meaning of clusterings dissimilarity and the types of possible dissimilarities that can exist among two clusterings in the context of community detection. We perform extensive evaluation to study the behaviour of the different similarity indices as a function of the dissimilarity type with both disjoint and non-disjoint clusterings. We, based on the outcome of our experiments, provide practitioners with some insights on which similarity indices to use for the task at hand and how to interpret their values.



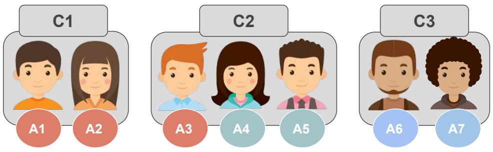
(A) Ground Truth



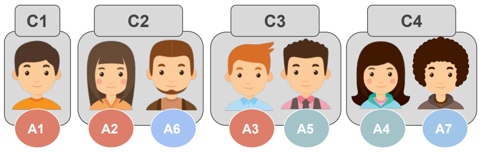
(B) **Omega** = 1 , **NMI** = 1, **AMI** = 1. A disjoint clustering that is identical to the ground truth.



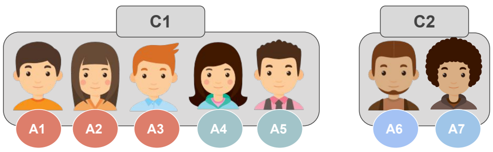
(C) **Omega** = 0 , **NMI** = 0, **AMI** = 0. A disjoint clustering constituted of a single community that is the union of all communities in the ground truth



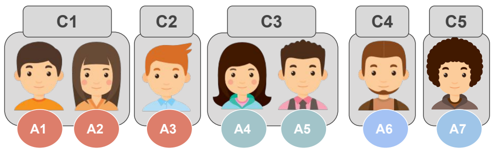
(D) **Omega** = 0.4750 , **NMI** = 0.7472, **AMI** = 0.5702. A disjoint clustering that is similar to the ground truth with only one misplaced node.



(E) **Omega** = -0.2174 , **NMI** = 0.4015, **AMI** = -0.1950. A disjoint clustering resulted by a random reordering of nodes in the ground truth communities.



(F) **Omega** = 0.4425 , **NMI** = 0.7446, **AMI** = 0.4437. A disjoint clustering resulted by merging two of the communities in the ground truth into one community



(G) **Omega** = 0.5039 , **NMI** = 0.8344, **AMI** = 0.3906. A disjoint clustering resulted by splitting two of the communities in the ground truth into two communities each

FIGURE 1.10: Numerical values of NMI, AMI and Omega-Index resulted by comparing the ground truth illustrated in 1.10a and other clusterings constituted of some variations of the ground truth.

1.5 Similarity analysis among different community discovery methods

The availability of different community methods, and consequently algorithms, provides practitioners with different options to choose from based on the available computational resources and the logic they want to adopt for defining a community. A valid question, however, is whether different community detection methods recover similar network patterns (i.e., communities) and if not, how can we choose the one most relevant to our problem. While a precise answer to this question requires a deep investigation of the differences that might arise among these methods based on the research question and the type of network used, here I provide practitioners with a high level distinction between these three different community discovery techniques - both theoretically (Section 1.5.1), and experimentally (Sec 1.5.2). In the former, I map the logic behind each method into a possible interpretation of a community in networks, while in the latter, I test the similarity between these methods as a function of three parameters – namely whether the network is directed or not, whether the network is weighted or not, and the level of noise in the network.

1.5.1 Theoretical analysis

The fact that the patterns identified by community detection methods in networks are usually referred to using the same term (i.e., communities) does not necessarily imply they recover similar patterns. Indeed, the lack of research on mapping the outputs of these methods to possible community definitions conceptually allowed for misleading and fuzzy interpretations for these patterns on a qualitative level. In this section, I analyze the logic behind each of the three community detection methods presented above and map each of them into a possible definition of communities.

Communities produced by modularity maximization are based on the idea of maximizing the number (or the total sum of weights) of edges within a community. In its essence, this definition of communities is static and does not look at the community as a result of possible dynamics in the network. This implicitly ignores the evolutionary aspect of the networks and the fact that edges in a network evolve over time. In addition, it is not clear when the amount of edges (or the total weight) within a group of nodes is 'dense' enough for it to be identified as a community, as this amount is relative to the level of sparsity of the edges across communities. Depending on what an edge between two nodes means, say that there was an interaction between them, all we can claim about a community resulted by modularity maximization is that the relative frequency (or the total weight) of interactions among members of this community is relatively higher than the total interactions of these members with the rest of the network.

Communities produced by Information flow mapping are the result of a dynamic process in the network. Therefore, the definition of a community is not directly based on how dense the edges within communities are in relation to the rest of the network, but rather on how the edges are placed. Here, the directionality of edges plays an important role even if the method works with undirected networks. In order to recover the communities, an information flow mapping method simulates the creation of edges in the network using a random walker that is constrained by the transitioning probabilities induced by the network edges. A community, as a result, is a set of nodes in which the edge creation simulator (i.e., the random walker) gets "stuck in" for a while before leaving. While the sparsely connected groups of densely connected nodes can increase the probability that a random walker would get stuck in each of these groups, this definition of community can be generalized more than that, as the random walker may get stuck by forcing a specific directionality among edges within a group of nodes, for example, instead of forcing a specific density of edges. In [18], the authors show an example of a toy network where modularity maximization and information flow mapping identify two different clusterings in the network.

Communities resulted by statistical inference are resulted by recovering the clustering $\mathcal{C}$ over the network with the highest probability of being responsible for edge creation in the network. This probability is maximized when the number of possible networks that can be generated by $\mathcal{C}$ is smaller. The number of possible networks that can be generated by $\mathcal{C}$ is smaller when edge creation probabilities within communities of $\mathcal{C}$ is higher, thus resulting in communities where the within-community edge creation probabilities are maximized. This method also provides a statistical significance to the meaning of a community as it guarantees, with a certain significance (that can be accepted or rejected), that there is no other solution better than the one provided.

1.5.2 Experimental analysis

To get a general idea about the level of agreement and disagreement among different community detection methods (modularity maximization, information flow maps and statistical inference) in practice, I perform some experiments to study the similarity among patterns identified by these methods on two different types of synthetically generated networks. The first is generated using LFR benchmarks [24], which maintains a power-law degree distribution in the resulted networks, while the second is generated using stochastic block modeling (SBM) [19] without forcing any degree distribution on the output networks. In this section, I refer to these two types of networks as LFR networks and SBM networks respectively. The goal of choosing these two different generative processes is not necessarily to study the effect of the degree distribution on the level of agreement between the different methods, but to consider the different generative models that have been used in the literature to generate networks with implanted communities. While I still invite more research on studying the similarity and dissimilarity among different community methods by considering more types of networks, including real-world networks, I claim that these experiments, as they are designed and presented, are sufficient to get a general idea about the level of agreement among the methods discussed above.

1.5.2.1 Results and observations

Figures 1.11 and 1.12 each report the pairwise similarity together with the accuracy (ability to recover the ground truth communities) among the different community detection methods with the two types of networks, LFR and SBM respectively. The three methods, modularity **M**aximization, statistical **I**nference and information **F**low are referred to as **M**, **I** and **F** respectively in these figures. For each type of network, the patterns were analyzed in four cases: 1) when the network is undirected and unweighted (**UD-UW**), 2) when the network is undirected and weighted (**UD-W**), 3) when the network is directed and unweighted (**D-UW**) and 4) when the network is directed and weighted (**D-W**). In addition, table 1.1 reports the accuracy values achieved by different community detection methods when averaged by experiments on LFR networks, SBM networks, and all networks ($AvgAcc_{LFR}$, $AvgAcc_{SBM}$, and $AvgAcc$ respectively).

Experiments were repeated for different assignments of the mixing parameter μ, which controls the percentage of cross-community edges in the generated network. General observations about the results are summarized in the following:

- Different community detection methods do not always agree in the clusterings they recover from the networks. Moreover, the similarity patterns among them are not necessarily maintained across the different settings of the network's directionality/weight, across different assignments of the mixing parameter μ, or across different network typologies (LFR and SBM). This finding is important as it sheds a light on the importance of understanding the logic behind each method and how that translates in each typology to provide more consistent interpretations of the patterns identified by each method.
- With networks that are generated using LFR benchmarks (Figure 1.11), there is a perfect agreement among the three methods with directed-unweighted networks (D-UW), and their abilities to recover ground truth communities, in this case, is not affected by the level of noise μ in the network. This agreement is relatively maintained with directed-weighted networks (D-W), as well as with undirected-unweighted networks (UD-UW), as long as the level of noise in the network is moderate (less than 0.25), at which point modularity maximization starts to become an outlier. With undirected-weighted networks, statistical inference shows a

	$AvgAcc_{LFR}$	$AvgAcc_{SBM}$	$AvgAcc$
Information flow mapping	0.99	0.6	0.8
Statistical Inference	0.61	0.89	0.7
Modularity maximisation	0.84	0.23	0.5

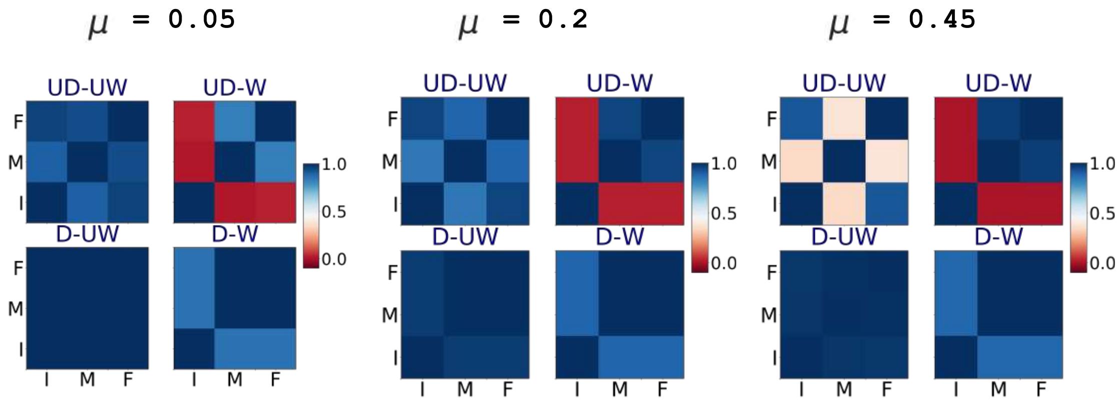
TABLE 1.1: Average accuracy values of different community detection methods where $NormAcc_{LFR}$ is the average accuracy value in all experiments on LFR networks, $NormAcc_{SBM}$ is the average accuracy value in all experiments on SBM networks and $NormAcc$ is the average accuracy value with all experiments

very poor ability in recovering ground truth communities, while modularity maximization and information flow maps demonstrate a similar behavior in recovering the ground truth communities.

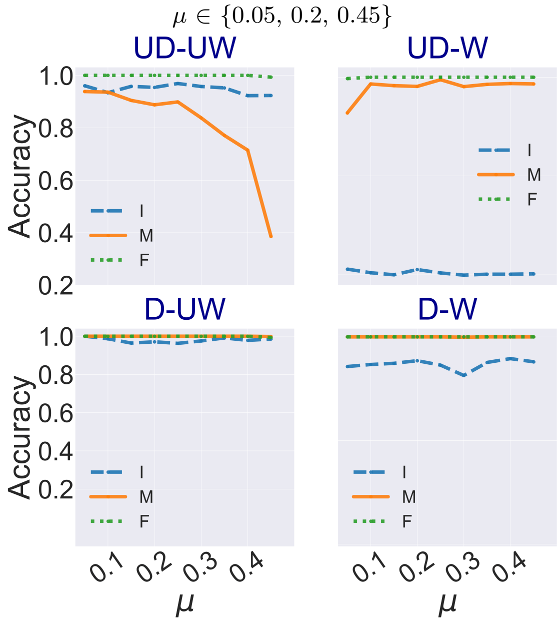
- With networks that are generated using stochastic block modeling SBM, both statistical inference and information flow mapping show a good level of agreement and ability to recover ground truth communities with weighted networks, whether directed or not (figure 1.12). With non-weighted networks that are generated using stochastic block modeling, both modularity maximization and information maps show poor abilities to recover ground truth communities, while statistical inference demonstrates a good accuracy with moderate levels of noise in the network (less than 0.25).
- On average, statistical inference seems to outperform other methods in recovering ground truth communities when the networks are generated using stochastic block modeling, while information flow mapping and modularity maximization methods outperform statistical inference with networks that are generated using LFR benchmarks. Generally speaking, information flow mapping method seems to outperform other methods in its ability to recover ground truth communities in different types of networks and different levels of noise in the network (table 1.1).

1.5.2.2 Experimental settings

Networks used in these experiments are constituted of 1000 nodes. The parameters used for generating LFR networks [24] are: minimum degree = 15, maximum degree = 50 , minimum number of communities = 20, maximum number of communities = 50 and mixing parameter for the weights in weighted networks = 0.1. The process of generating SBM networks takes as an input a clustering $\mathcal{C}$ over the network nodes and a probability matrix which specifies the edge creation probabilities within and across communities of $\mathcal{C}$. In these experiments, $\mathcal{C}$ was chosen to be constituted of 10 communities where the community memberships were sampled from a categorical distribution, and community



(A) Pairwise similarity among community detection methods with three values of the mixing parameter $\mu \in \{0.05, 0.2, 0.45\}$

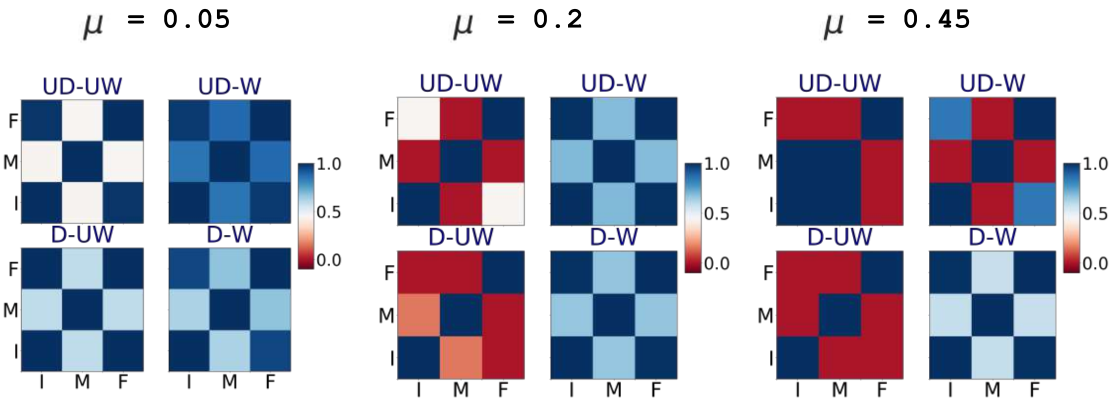


(B) Accuracy of different community detection methods as a function of the mixing parameter μ

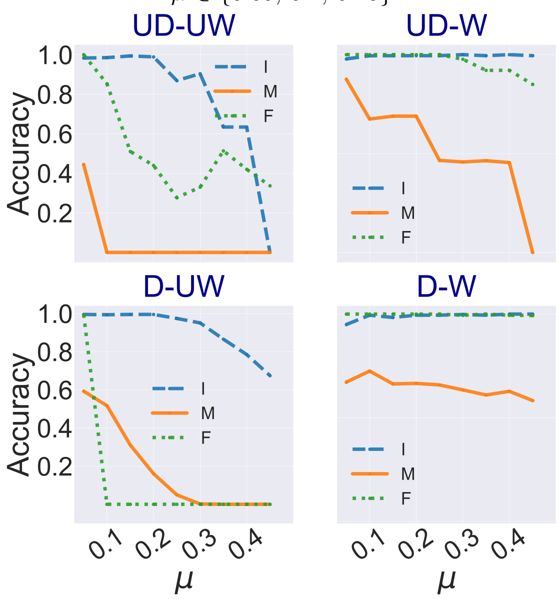
FIGURE 1.11: Accuracy and pairwise similarity among three community detection methods, namely modularity maximization (**M**), information flow maps (**F**) and statistical inference (**I**), with networks generated using the LFR benchmark in four different cases: when the network is undirected and unweighted (**UD-UW**), when the network is undirected and weighted (**UD-W**), when the network is directed and unweighted (**D-UW**) and when the network is directed and weighted (**D-W**)

sizes were sampled from Dirichlet distribution (as used by [25]). Edge probabilities were chosen to be between 0.5 and 0.7 for the within-community edges and μ (the value of the mixing parameter) for the cross-community edges. To generate weighted networks (as the model used does not provide that possibility) , I started by generating non-weighted networks, then weights were placed randomly such that they take higher values when they connect two nodes of the same community and lower values otherwise.

For each of the three methods, a representative algorithm that provides a good approximation in implementing the logic behind the method was chosen. This resulted



(A) Pairwise similarity among community detection methods with three values of the mixing parameter $\mu \in \{0.05, 0.2, 0.45\}$



(B) Accuracy of different community detection methods as a function of the mixing parameter μ

FIGURE 1.12: Accuracy and pairwise similarity among three community detection methods, namely modularity maximization (**M**), information flow maps (**F**) and statistical inference (**I**), with networks generated using stochastic block modeling in four different cases: when the network is undirected and unweighted (**UD-UW**), when the network is undirected and weighted (**UD-W**), when the network is directed and unweighted (**D-UW**) and when the network is directed and weighted (**D-W**)

in choosing Louvain [15], Infomap [18] and MCMC [22] algorithms as representatives of modularity maximization, information flow mapping and statistical inference methods respectively. For calculating the similarity and accuracy values, the adjusted mutual information was used for its sensitivity to capture different types of dissimilarities and its consideration for the by-chance agreement among clusterings as we discussed in a previous study [26] reported in Appendix C.

With weighted networks, modularity maximization translates into maximizing the total sum of weights (rather than the number edges) within communities and minimizing that amount across communities. The same method (i.e., Louvain) can guarantee that if we assign weights to the entries in the adjacency matrix A, rather than 0s/1s for the absences/existences of edges. However, with directed networks, a variant of Louvain that accounts for the direction of edges is used [27]. As to the other algorithms (Infomap and MCMC), the directionality and accounting for edge weights is naturally implemented in them.

1.6 Applications of community discovery

Community detection has been applied in a wide variety of real-world scenarios, and for a systematic review of these applications, I refer the reader to [28]. In the following, I point out a few of these applications to showcase how this tool has been used in practice and to highlight the usefulness of community detection.

- **Bot Detection:** Bots are automated software that can be implanted in social network platforms (Twitter, for example) to perform specific actions (retweeting a specific content, or sending direct messages to other account). A specific type of these bots, referred to as social bots, can play the role of automated social actors in a social network, and hence can be maliciously used to infiltrate human conversations, disseminate fake news, impersonate other social actors, or manipulate the stock market. The analysis of the network topology is a way for the detection of some of these bots (referred to as Sybil accounts), which are defined to be multiple accounts that are controlled by an adversary. As discussed in [29], the Sybil detection problem can be regarded as a community detection problem as these accounts form many peripherally small communities rather than constructing one big community.

- **Analysis of brain graphs:** The human brain can be represented as a graph where nodes represent brain cells or small volumes of tissue, and edges among these nodes represent structural or functional links among the corresponding brain areas. In [30], authors discuss how community detection in brain networks can correspond to areas in the brain that are activated by the performance of specific cognitive and behavioral tasks. Community detection in this area has also revealed changes in the community structure in brain networks of a set of participants while learning a simple motor skill across several days [31].

- **Topic discovery:** Community detection has applications in the are of topic detection in social networks. For example, authors in [32] applied community detection on a YouTube graph where different nodes refer to different videos and a weighted edge between two nodes captures the level of similarity between them. The result is communities that refer to different topics based on the content of the video.

- **Observational political studies:** Community detection in political networks can provide insights on the level to which political ideologies, affiliations or coalitions influence certain dynamics in the network. For example, our analysis of the Twitter interactions among Danish politicians during the parliamentary elections of 2015 has revealed communities that resemble, to a good extent, the main political affiliations in Denmark [33].

- **Link prediction:** Link prediction is a network task that is usually used to predict future links among social actors or missing links during data collection. There are different approaches to assign the probabilities to these missing links and community detection has been used as one of those approaches. The intuition is that missing links among members of the same community have higher probabilities to be missing (or to appear in the future) [34].

- **Recommendation systems and targeted advertising:** In its essence, the task of community detection in social networks is supposed to group like-minded people together in communities. This information can be used to improve targeted advertising and recommender systems for more community-personalized suggestions [30].

In Brief

- Social network analysis investigates properties, patterns and dynamics in social networks that are represented as graphs, where nodes correspond to social actors and edges among these nodes refer to social relations, interactions or flows of information.

- Community discovery in social networks is a very relevant task in the analysis of social networks for its ability to identify patterns that formalize the very important concept of ‘social groups’ used in social sciences.

- The notion of social groups in social sciences is not defined in the literature which resulted in different interpretations of the problem of community discovery in social networks.

- Three methods for community discovery in social networks stood out for their abilities to recover implanted clusterings in synthetically generated networks and their scalability to large and complex networks. These interpretations are conceptualized into the following three community discovery methods: i) modularity maximization, ii) information flow mapping and iii) statistical inference using block modeling.

- Preliminary similarity analysis among patterns identified by the three methods showed that the different community detection methods (modularity maximization, information flow mapping and statistical inference using block modeling), do not always agree in the patterns they recover in networks and their agreement patterns are not necessarily maintained across different types networks – that is, whether the network is directed or not, whether the network is weighted or not, and across different levels of noise in the network.

- Community detection has been applied in a wide variety of areas including but not limited to social bot detection, analysis of brain graphs, topic discovery in social networks, link prediction and recommendation systems.

Chapter 2

Does community discovery in social networks discover Communities?

Despite the term 'community' in community discovery, a brief review of the definition of Community in social sciences suggests that the network patterns identified by community discovery are much more general. Indeed, these patterns are closer in their properties to the notion of 'social groups', rather than the notion of 'Community'. This chapter starts off by providing a review of the notions of 'Community' and 'social groups' from social sciences in Section 2.1. In Section 2.2, the chapter discusses different conceptualizations of community discovery in social networks into specific edge patterns.

2.1 Communities versus social groups

- *"An elephant is like a rope."*, said the one who touched its trunk.
- *"No that is not true."*, replied the one who touched the side, *"an elephant is just like a wall."*
- *"You are both wrong."*, another who touched the leg said, *"an elephant is like a column holding a rooftop."*
- *'Seriously, guys!? Does it require much intelligence to realize the similarity between an elephant and a chapati (roti)?"*, the one who touched the ear said.

And on and on the four blind men kept arguing.

Indeed, many concepts in social sciences are like the 'elephant' in that story, quite hard to find a consensus on a precise definition for in the literature. The notions of 'community' and 'social groups' are apparently few examples of such concepts. Despite the widespread attention the study of these constructs has received in many disciplines, researchers have always used the terms without providing a precise definition. In the following, I review how these concepts - 'community' and 'social group' - were defined respectively in social sciences.

The concept of "community" emerged in classical sociology as a product of the ideological conflict between conservatism and liberalism which took place in the 19th century. During the democratic political revolutions of North America and France, and the shift in economics towards industrialization, the concept of community emerged as a way to refer to ideologies of the past and the present. At the beginning, it was not used as an analytic concept, meaning that what community is was not clear. Instead it was used as a normative concept to refer to what a community should be [35].

Generally speaking, this concept has been developed in social sciences in two directions. The first sees a community from the perspective of the relationship type – the sense of identity, the common traits, or the spirit among a group of people. The second puts emphasis on the geographical sense of community, that of a particular territory, to refer to a local social system or a variety of social relations among a group of people in a particular bounded area [36]. According to Ferdinand Tonnies [37], a social community resembles the natural community of a living organism. It involves an underlying consensus based on many factors including kinship, common residence, and friendship. Tonnies makes a distinction between the social status ascribed by birth and that ascribed by personal achievements, education, and properties. To him, being part of a 'community' is a status ascribed by birth, and therefore a community is homogeneous.

A review that goes back to 1955 found no less than 94 definitions of community with only one common denominator: all the definitions deal with people [38]. Another review by Collin [39] pointed out important differences in the dynamics of a rural versus urban communities. The social interactions in rural communities are more familial – they involve intimate face-to-face relationships, they have minimal specializations or division of labor and they are permanent, strong and durable. In these communities there exists an intense cohesion and a deep commitment to shared values. In urban settings, however, the societal interactions can be casual, superficial and short-lived.

Clearly, a consensus about one definition of communities is non-existent. However, they all express a level of commonality in traits, behaviors, values, ideologies, or geographical space among members of a group. Here I quote a few definitions of "community" as they were reviewed by [35].

- "A community is a set of interrelationships among social institutions in a locality."
- "A community is, first, a place, and second, a configuration as a way of life, both as to how people do things and to what they want, to say, their institutions and goals."
- "Community refers to a structure of relationships through which a localized population provides its daily requirements."
- "A community is a collection of people who share a common territory and meet their basic physical and social needs through daily interaction with one another."
- "A community is a social group with a common territorial base; those in the group share interests and have a sense of belonging to the group."
- "A community is a body of people living in the same locality. Alternatively, a sense of identity and belonging shared among people living in the same locality, or the set of social relations found in a particular bounded area."

Even though "*everybody knows what it means*", as claimed by Freeman [40], the concept of 'social groups' is yet another heavily researched and not precisely defined concept in sociology. As discussed by [41], a social group can be defined as two or more people who perceive themselves to be members of the same social category. It reflects a certain level of social or psychological interdependence among members of the group. This interdependence can be for the attainment of goals, the validation of values and attitudes, or the satisfaction of needs. According to [42], a social group is a set of two or more persons who are interacting with each other in a way such that one gets influenced by and/or influences the other one. In [43], a social group is defined as the group of people who identify themselves as part of the group. Some researchers in social psychology has defined the social group to be a set of people interacting with each other based on common motives and goals, an accepted division of roles in the group, accepted norms and values with respect to issues that matter to the group, and an accepted code for praise/punishment when values of the group are respected or violated [44]. Examples of social groups can be: couples, families, cliques, gangs and communities.

The revolution of technology and the availability of various social communication means have made social interactions no longer dependent on geographical proximity. Instead, they can literally be with anyone and anywhere [45]. This allowed for the formation of social groups that make use of online social platforms in order to substitute the absence of a common geographical territory. Indeed, the openness of the term 'community' and the lack of agreement among sociologists on how this definition should be updated allowed researchers to consider such online groups as a special type of communities [46]

. These are communities that use online means to interact with one another, and hence can be called “online communities”, while the previous notion of community began to be referred to as “offline community”. The term “online community” is yet another notion that suffers from the lack of consensus among researchers on its precise definition. However, the intuition is always the same – an aggregation of individuals who interact around a shared interest, where the interaction is fully, or at least partially, supported or mediated by technology (or both), and guided by some protocols or norms [47]. The main distinction between online communities and offline ones, therefore, is the geographical reach. Today, an online community may have members from Denmark, the US, Japan, Australia, and Syria. In fact, members can be from anywhere in the world as long as they have access to the Internet.

While the list of possible definitions for the concepts of ‘social group’ , ‘offline community’ and ‘online community’ can be easily expanded, it is not hard from what is discussed above to spot a few distinctions among them. A first impression one can get by comparing these notions is that the concept of ‘social group’ is much more general and relaxed compared to the other concepts – so no geographical constraints, restrictions on the communications means or some sense of belonging are forced among members to identify as a social group. In addition, despite the fact that the notion of community did not explicitly force any restrictions on the number of members in its definitions, it is implied somehow that a group of two people can not identify as a community, while two persons can still form a social group.

2.2 Conceptualizations of community discovery in social networks

It comes as no surprise that the relatively fuzzy concepts of ‘community’ and ‘social groups’ in social sciences would result in multiple conceptualizations of these concepts into specific patterns in social networks. This starts from the variety of social-network terms that were given to such constructs (like communities, cohesive groups, cohesive subgroups, meso-scale structures, modules, and modular structures), and ends with as many community discovery methods as the number of social network analysts working in the field. Even though it is more commonly known as community discovery, I claim that this task in social networks does not aim to recover structures that map directly to the ‘community’ concept from sociology, but rather to the more general concept of ‘social groups’ which can be seen as a relaxation of the ‘community’ concept with regards to its size, geographical constraints, communication means and the sense of belonging. While I still use the terms ‘community discovery’ and ‘communities’ in this thesis for consistency, I assume that the task of community discovery in social networks is not restricted to communities as defined in social sciences but can also be seen as a task to identify social groups.

Despite the plethora of conceptualizations of social groups in networks, there are four main properties that influenced the formalization of this concept into social network patters according to [10]. These properties are:

1) Mutuality of ties, which requires that all pairs of members in a group to choose each other.

2) The closeness or reachability of members of the group. This does not necessarily force each pair of the group members to be directly connected, but at least reachable to each other.

3) The frequency of ties among members of the group. This requires each member of the group to have many ties with other members within the group.

4) The relative frequency of ties with members within the group with respect to ties with members outside of the group. This requires groups to be relatively cohesive when compared to the rest of the network.

In the following, I discuss the mapping of each of the previous properties into network patterns. Most of the theoretical content about these properties, but not their mappings, in this section is based on the book by Stanley Wasserman and Katherine Faust, "Social Network Analysis: methods and applications" [10].

2.2.1 Social groups based on mutuality of ties

Mutuality of ties in a social group maps to a very well-established and precisely defined concept in graph theory – the clique. A clique is a maximal complete subgraph of three or more nodes, where each pair of nodes is connected via a non-directional relationship. In figure 2.1, the group of nodes 1, 2, and 3 form a clique of size 3 because each pair of nodes in this group are connected. Many community discovery methods have the idea of 'clique' in their essence. Clique percolation method, for example, conceptualizes community discovery in social networks into finding groups of nodes that consist of highly overlapping cliques [48]. More clearly, given a parameter k, the method tries to find groups of nodes such that each group consists of cliques of size k that overlap with each other in k-1 nodes. For example, in figure 2.1, there are two cliques of size 3 (one constituted of nodes 1, 2 and 3, and another one constituted of nodes 1, 3 and 4) which overlap together in two nodes (1 and 3). If k is set to 3 in the clique percolation method, the group of nodes 1, 2, 3 and 4 will be identified as a single community.

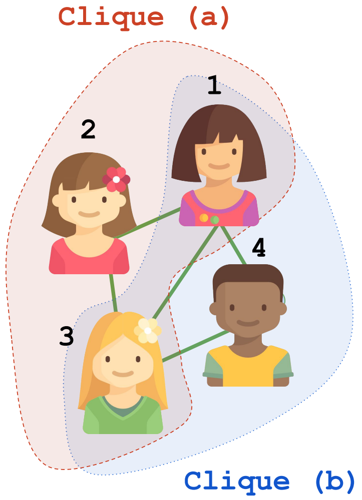


FIGURE 2.1: An illustration of two overlapping cliques of size 3

2.2.2 Social groups based closeness and reachability

In graph theory, there might exist many paths between two nodes. The shortest of these paths in terms of the number of nodes it must travel through is referred to, as the name suggests, the 'shortest path' between those two nodes. Reachability or closeness among members of a social group can be quantified in graph theory terms as the length of the shortest path in the graph between those two members. Groups that are based on the idea of reachability can be seen as a relaxation of the very strict definition of cliques which require each pair of members in the group to be directly connected. Rather than forcing a direct connection, reachability groups require each pair of nodes in the group to be closely reachable by requiring the shortest path length between each pair of members in the group to be small. In this context, small can be chosen as a threshold that the shortest path length among each pair of nodes does not exceed. Figure 2.2 illustrates a group of 4 nodes that are closely reachable to each other with a threshold of 2.

One community detection method that seems to be based on the idea of reachability is the information flow mapping method mentioned in Section 1.3.2. This method benefits from the duality between recovering important structure in the network, and minimizing the code of a random walker traversing the network. The code of such random walker is minimal for a clustering that divide the network into groups of nodes that are highly reachable by other nodes belonging to their group and hardly reachable by nodes from other groups. Another method can be the one proposed by [49]. In their algorithm, the authors assume that nodes with high betweenness centrality in the network represent

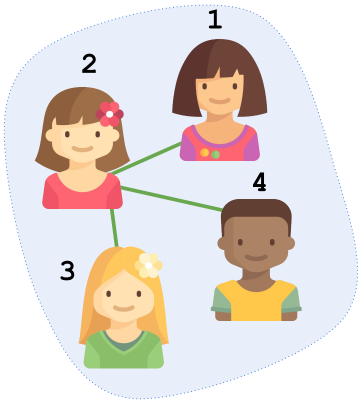


FIGURE 2.2: A group of 4 nodes closely reachable with a threshold of 2

bridges in the network that, when removed, can reveal boundaries of a community structure. Therefore, the algorithm proceeds by calculating the betweenness centrality of the nodes of the graph, removing edges of the nodes with the highest betweenness centrality, and re-calculating the betweenness centrality of the nodes until there are no edges in the graph. Each round in this interactive process can identify a level of division in the network into communities.

2.2.3 Social groups based on frequency of connections with group members

Groups that are identified based on the within-group frequency can be seen as another relaxation of the strict concept of 'cliques'. Rather than forcing a direct link between each pair of members, this approach assumes a minimum number of links between each member and the rest of the group members. This minimum can be defined as a threshold k which, when it is equal to the number of members in the group, results in a clique. A group of nodes that satisfies the condition that each member of the group is connected directly to at least k other members of the group is called k-plex. Figure 2.3 illustrate a k-plex of 5 nodes with $k = 2$.

Intuitively, the higher the value of k is in a k-plex, the more cohesive the group can be. Based on that idea, a class of community discovery methods consider the problem to be equivalent to the maximum k-plex problem in graphs [50]. In simple terms, the maximum k-plex problem guarantees partitioning the graph into k-plexes with the maximum possible k such that there is no clustering with a higher value for k.

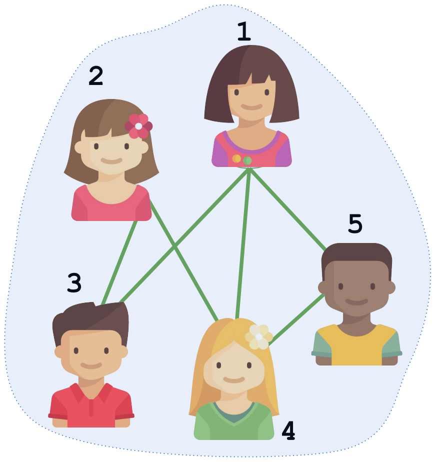


FIGURE 2.3: An illustration of a k-plex among 5 members with k=2

2.2.4 Social groups based on relative frequency

Based on the idea that some forms of social groups can express a tendency to interact more with members of their group than with outsiders, this approach identifies the social group in terms of its relative frequency of ties within the group as compared to the frequency of interactions with members outside of their group. The higher this relative frequency is, the more cohesive the group. Indeed, this perception of what a social group is seems to be the most common way of defining a community social network. As discussed in Section 1.3.1, modularity maximization community discovery methods aim to find such groups in networks. Another method that seems to implicitly find similar patterns is the statistical inference method mentioned in Section 1.3.3.

In Brief

- While the concepts of 'community' and 'social group' have plenty of not completely aligned definitions in social sciences, they are also clearly distinguishable. The former often puts emphasis on geographical commonality, communication means (in the case of online communities), and the sense of identity and belonging. The latter is more general and puts emphasis on common influences, interests, and interdependence on each other.
- Despite the name community discovery, the patterns recovered by this task in social networks are closer in their nature and properties to the 'social group' concept than to the 'community' concept in social science.

- There are plenty of conceptualizations of social groups in social networks. Four properties have influenced the formalization of these conceptualizations, namely mutuality of ties, closeness and reachability of members within the group, frequency of ties with members of the group, and relative frequency of ties within the group compared to that with members out of the group.

- Communities resulted by the information flow mapping method are based on closeness and reachability across nodes of a group. Modularity maximization and statistical inference, on the other hand, recover patterns based on the relative frequency of ties among members within the group relative to ties with members outside the group.

.

Chapter 3

The Multi-Layer Network Framework

Multi-layer networks aim to model inter-dependencies among different relationships in a social context, in addition to the relationship ties themselves among the social actors. This chapter starts off with an introduction to the multi-layer network framework in Section 3.1. In Section 3.2, the definition of community detection in multi-layer networks is pointed out, followed by a discussion of some community detection methods in multi-layer networks in Section 3.3. I continue in Section 3.4 by proposing some important multi-layer community models, followed by a commentary on our systematic review of community detection methods in multi-layer networks in Section 3.5. Finally, the chapter presents two use cases for the use of multi-layer community detection in the real-world in Section 3.6.

3.1 The multi-layer network framework

The graph model has provided a useful mathematical tool to model certain social relationships among social actors. This allowed computational methods developed for analysing graph data to provide useful insights about social phenomena that might arise from certain social relationships. Social relationships, however, do not exist in isolation and they influence each other such that one may depend on, or lead to, another one. Being a friend on Facebook with someone, for example, is more probable to happen if they already had some off-line activity beforehand (like going to the gym together or meeting in an event). Studies on Twitter user behaviour showed some level of dependency between retweeting behaviour and mentioning behaviour, meaning that users who have been mentioned by other users showed a higher tendency to re-tweet their posts

[51]. Considering only one type of relationship in the study of complex networks risks either defining a world where different kinds of relationships are equivalent, or overlooking the invisible relationships emerging from the interactions among different types of relationships [52]. It has become indeed clear in the last few years that characterizing these inter-dependencies is essential for understanding most complex networks - including economical networks, brain networks, cellular networks in addition to social networks [53]. Given that the graph model as is can not leverage this information, an extension of graphs into multi-layer graphs has been proposed.

Multi-layer networks, also referred to as layered, multivariate, multidimensional, multi-slice, or multi-relational networks, are a generalization of the notion of the graph to model different aspects of a connection between some actors, such as different types of relationships, or different points in time when a relationship happens [25]. Each relationship is characterized by a *layer*, i.e., single-graph, and two nodes in a layer are connected via an *intra-layer edge* if the corresponding actors are connected through the relationship modeled by that layer. *Inter-layer edges* are used to couple nodes across different layers, and they are called *diagonal* if they connect only nodes of the same actor. Figure 3.1 provides an illustration of a toy single-aspect two-layer network modeling connections among 15 actors.

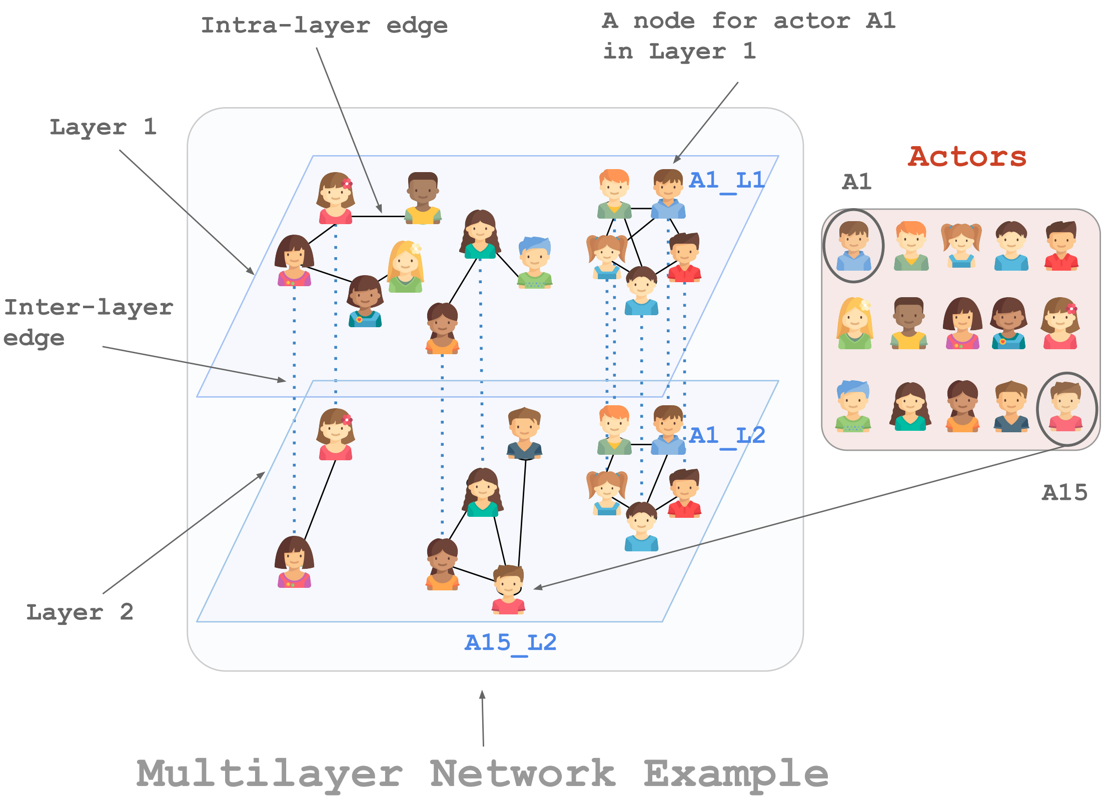


FIGURE 3.1: A layered representation of a single-aspect two-layer network modelling connections among 15 actors.

The highest abstraction in the multi-layer model is the *aspect* [54]. Aspects represent the high-level features that we want to model between actors, such as time, relationships, and so forth. The second level of abstraction is the *layer* which decomposes an aspect into multiple states, categories, types, points in time, etc. Figure 3.2 illustrates a two-aspect multi-layer network where the studied aspects are relationship and time. The relationship aspect is decomposed into two layers, namely friends and work. The time aspect decomposes each of the relationship aspect layers into three layers, one for each of the years 2018, 2019 and 2020. The resulted multi-layer network is constituted of 6 layers as shown in the figure.

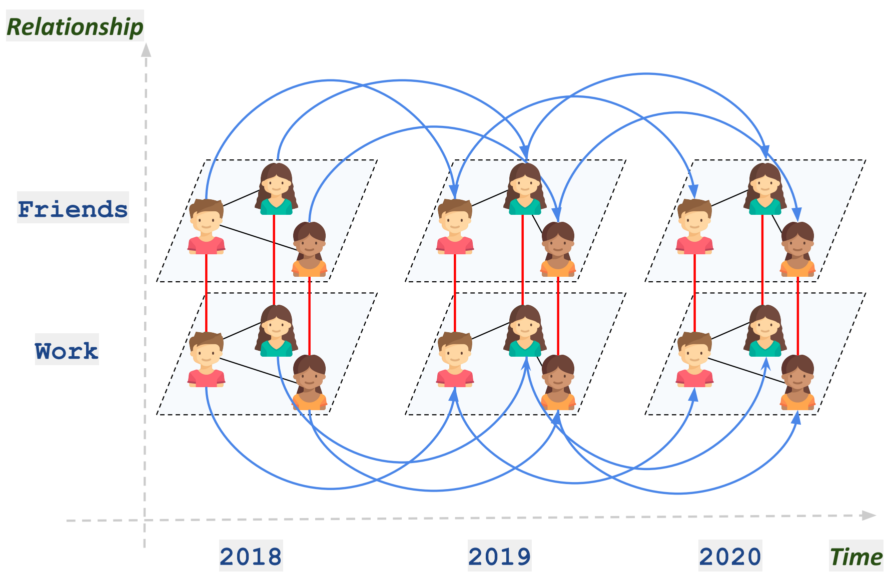


FIGURE 3.2: A visual illustration for a two-aspect multi-layer network. The aspects are relationship and time. The relationship aspect is decomposed into two layers, namely work and friends, and the time aspect decomposes each of the relationship layers into three layers, one for each of the years 2018, 2019 and 2020.

All inter-layer edges in figure 3.2 are *diagonal* as they only connect nodes of the same actor. The inter-layer edges connecting nodes across different relationships (the red ones) are said to be *categorical* as there is no specific order for the layers that these edges connect, and hence they are also usually undirected. Inter-layer edges connecting nodes across different times (the blue ones), however, are said to be *ordinal* as they connect layers that have a specific order forced by time, and are therefore directed in order to capture the successive order of a relationship.

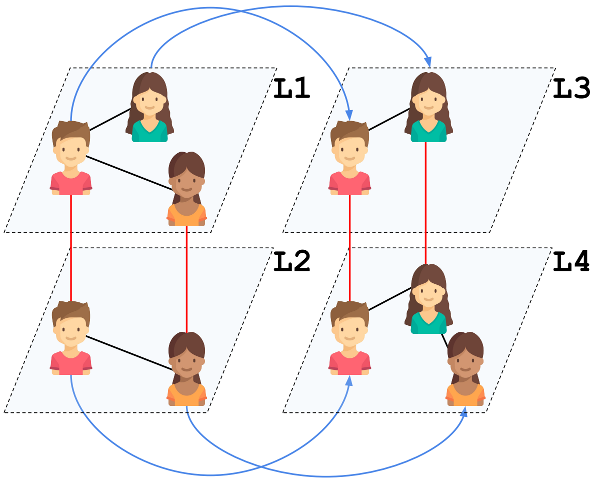
In algebraic terms, multi-layer networks have been represented in several ways in the literature. Assuming a multi-layer network M $(V, \mathcal{L}, \mathcal{V}_M, \mathcal{E}_M)$ with $n = |V|$ actors and $l = |\mathcal{L}|$ layers, where $\mathcal{V}_M$ is the set of nodes and $\mathcal{E}_M \subseteq \mathcal{V}_M \times \mathcal{V}_M$ is the set of edges. Here I refer to three different representations, namely, i) the multiple matrices representation used by [55], ii) the supra adjacency matrix representation used by [56] and iii) the adjacency tensor representation used by [25]. In the following, I provide a definition for each representation:

The multiple matrices representation: In this representation, each layer s is represented as a $n \times n$ adjacency matrix A_s. To represent the inter-layer edges, $\frac{l*(l-1)}{2}$ matrices, referred to as the coupling matrices, are used so that the inter-layer edges between each pair of layers are captured. A single coupling matrix $C_{s,r}$ between two layers s and r, is a $n \times n$ matrix, and an entry (i,j) which is not 0 is used to couple node i from layer s to node j from layer r. If inter-layer edges are diagonal, entries of the coupling matrices will be 0 in all cases except for those in the main diagonal. Figure 3.3 provides an illustration of this type of representation on a toy multi-layer network example.

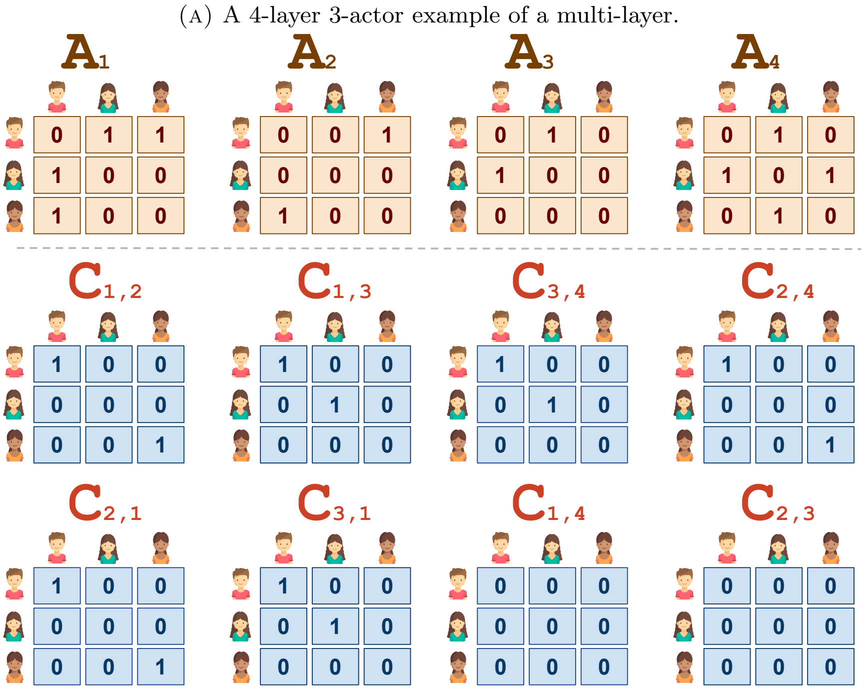
The supra adjacency matrix: This representation integrates matrices of the previous representation into one matrix called the supra adjacency matrix $\mathscr{A}$. Nodes are given unique identifiers so that the existence of an actor in one layer is represented by a universally unique node identifier rather than the combination (actor, layer), as was the case in the previous representation. This results in a $(n*l) \times (n*l)$ matrix. This matrix is comprised of $l \times l$ blocks where the diagonal blocks are the adjacency matrices of the different layers, and an off-diagonal block (s,r) is the coupling matrix between layers s and r. Figure 3.4 illustrates the supra adjacency matrix of the same network used in the previous representation.

The adjacency tensor representation: A tensor, in simple terms, is an Algebraic generalisation to represent multi-dimensional arrays. Multi-layer network information can be represented using an adjacency tensor $\mathcal{A}$ that has four dimensions - two of them are used to refer to the actors, while the other two are used to refer to the layers. An entry $\mathcal{A}_{i,j}^{s,r}$ is equal to 1 (or a given weight) if there is an edge between actor i in layer s and actor j in layer r, otherwise it is equal to 0. Figure 3.5 illustrates the adjacency tensor of the same network used in the previous representation.

The key difference between the multi-layer graph model and other previous graph generalisations that consider assigning properties to edges/nodes of the graph is the introduction of inter-layer edges - that is, the fact the connection patterns in one layer might depend on another layer. Multi-graph model, for example [57], which precedes the multi-layer one, allows for multiple parallel edges between two nodes in a graph where each edge can be assigned a different property (a relationship type or a timestamp for



(A) A 4-layer 3-actor example of a multi-layer.



(B) A representation of the multi-layer network illustrated in 3.3a using multiple adjacency matrices A_1, A_2, A_3, and A_4 for layers L_1, L_2, L_3, L_4 respectively, and coupling matrices $C_{i,j}$ for each pair (i,j) of layers.

FIGURE 3.3: Representing a multi-layer network using multiple matrices

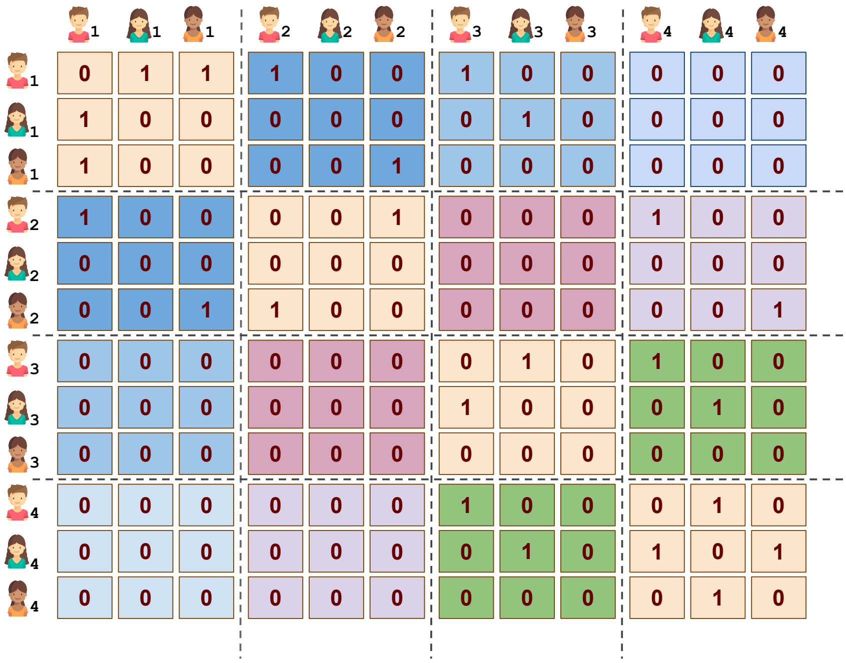


FIGURE 3.4: Representing the multi-layer network illustrated in figure 3.3a using a $(n * l) \times (n * l)$ supra adjacency matrix. Nodes are given unique identifiers so that the existence of an actor in a layer is represented by a universally unique node identifier rather than the combination (actor, layer) like in the previous representation.

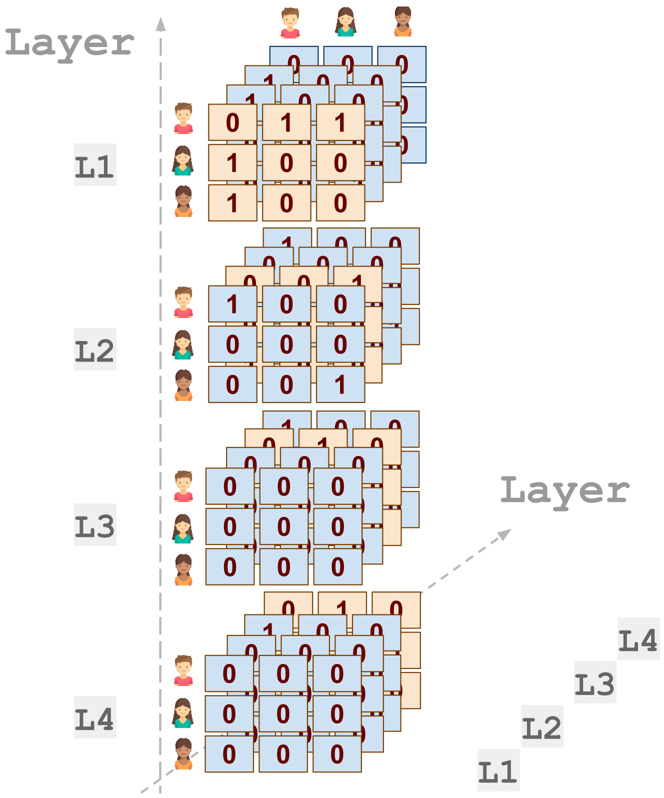


FIGURE 3.5: Representing the multi-layer network illustrated in figure 3.3a using an adjacency 4-d tensor where two dimensions are used to refer to the layers and the other two are used to refer to the actors.

example). However, it does not allow for modelling inter-dependencies among these nodes across different properties.

Two special applications of the multi-layer model that received a great deal of attention in the literature are *multiplex* networks–that is, networks that model different types of relationships among a set of actors [58], and *time-dependent* (or temporal) networks–that is, networks in which the interactions among actors change over time [59]. When the multi-layer model is used to model the former, layers correspond to different types of relationships and are thus categorical, while in the latter, layers represent different time-points of a relationship, and thus are ordinal. Multiplex networks have been used in a variety of different applications including, to mention a few, ecological networks (which model different types of interactions among species) [60], and social networks (where different layers can be used to model different types of interactions provided by a social media platform or different relationships across different social media platforms) [61]. Temporal networks have been used to model brain network dynamics (where layers capture the interaction among different areas in the brain over time) [62], and citation networks (where layers refer to different time points in which the citation happened) [63]. Unless mentioned otherwise, I keep the focus within this thesis on those two cases of multi-layer networks. I refer to graphs that model a single relationship among a set of actors as *mono-layer* networks, and to community discovery task in such networks as mono-layer community discovery, and to community resulted by such task as mono-layer community. Similarly I call a *multi-layer* community discovery the task of clustering nodes of a multi-layer network into communities, and to a community resulted by such clustering as a *multi-layer* community.

3.2 Multi-layer community detection

As discussed before, the term 'community' has been conceptualized in various ways in mono-layer networks. The conceptualization of that in multi-layer networks, however, seems to be missing in most of the research that proposed new multi-layer community discovery methods. For example, it is common to accept the conceptualization of mono-layer communities based on the relative frequency as discussed in Section 2.2.4, but no extension of this is provided in the literature for multi-layer networks. On a higher level, the motivation for using multi-layer community discovery seems to be sound. Nonetheless, it is not always clear what patterns in multi-layer networks is a multi-layer community discovery method able to reveal, and in which way that maps to qualitatively meaningful communities. Surprisingly enough, the literature about community discovery in multi-layer networks provides a plethora of multi-layer community discovery methods.

I claim that the reason for this gap between the lack of ad-hoc conceptualizations for multi-layer communities and the abundance of methods that recover what is claimed to be multi-layer communities is two-fold. First, multi-layer community discovery as a tool emerged as a response to the generalization of graphs into multi-layer graphs, and not to the generalization of communities into multi-layer communities. Second, most multi-layer community detection methods are mathematical extensions of mono-layer community discovery methods, which made it possible to provide such methods without having to investigate a precise extension of their original conceptualizations.

From a structural perspective, authors in [25] argue that there are two primary differences between a multi-layer community and a mono-layer one. First, as the name suggests, a multi-layer community expands over multiple layers. Second, edges of a multi-layer community in one layer depend on, or lead to, edges in another layer. For example, in a 2-layer multiplex network that models different Twitter interactions among a set of Twitter users Following and Retweeting, it might be the case that the retweet edges among a set of users are largely dependent on whether these users follow each other or not, which might be responsible for the existence of multi-layer communities that expand over the two layers. In the following, I discuss the extensions of the three mono-layer community discovery methods discussed in Section 1.3 into multi-layer community discovery methods.

Based on the definition of a multi-layer community proposed above, a multi-layer community detection task is not only expected to use the intra-layer information to provide a meaningful grouping of the nodes in their layers, but to also use the dependencies among edge patterns in different layers to provide a meaningful aggregation of them into multi-layer communities as illustrated in Figure 3.6.

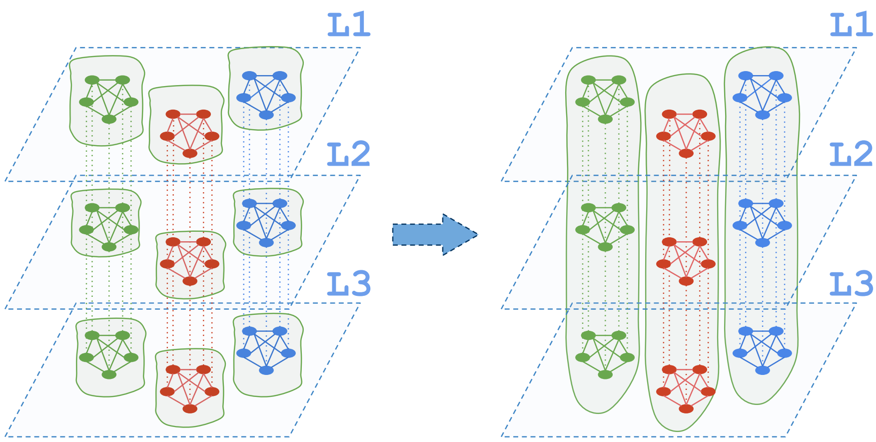


Figure 3.6: An aggregation of dependent edge patterns across different layers of a multi-layer network into multi-layer communities

3.3 Multi-layer community detection methods

As mentioned above, the attention multi-layer community discovery received from social network analysis has produced a plethora of methods to recover the so called multi-layer communities. In this section, I will refer to only three of these methods, which are extensions of the methods discussed in 1.3 to multi-layer networks. The original methods are modularity maximisation, information flow mapping, and statistical inference. I refer to their extensions to multi-layer methods as multi-layer modularity maximisation, multi-layer information flow mapping, and multi-layer statistical inference respectively. On one hand, the choice to discuss these three methods in this section stems from the popularity and success their original mono-layer versions previously had. On the other hand, I think of these choices as random picks from multi-layer community discovery methods to investigate their mathematical formulations and how multi-layer communities are defined accordingly. Later, in Section 3.5, I refer to our more comprehensive review and similarity analysis of community discovery in multi-layer networks.

3.3.1 Multi-layer modularity maximization

A generalization of the modularity formula 1.1 has been proposed by [55] for it to apply to multi-layer networks. In this generalisation, the authors introduced a new parameter, the coupling strength ω, to the modularity equation. The coupling strength is a weight assigned to the inter-layer edges connecting nodes of an actor across different layers. On the layer level, it has been interpreted as the closeness among different layers [56]. On the actor level, it has been claimed that this parameter reflects information about the extent to which an actor tends to have the same behaviour across layers (corresponding to high values for ω), or a different behaviour in different layers (corresponding to low values for ω) [64]. With this introduction of ω, the modularity score does not only reward density/sparsity of intra-layer edges within/across the communities but also rewards the existence of coupling edges (i.e., inter-layer edges) within communities, and this reward is proportional to the value given to ω. This means that if two nodes n_{ix}, n_{iy} which refer to the same actor i, and hence are coupled, happen to appear in the same community, this contributes positively to the multi-layer modularity score with an amount proportional to ω_{xy} - that is, the coupling strength assigned to the coupling edge between n_{ix}, n_{iy}. Given a multi-layer M (V, $\mathcal{L}$, $\mathcal{V}_M$, $\mathcal{E}_M$) that is represented using multiple matrices (as shown in section 3.1), the multi-layer modularity of a multi-layer clustering $\mathcal{C} = \{\mathscr{C}_1$, $\mathscr{C}_2$, $\mathscr{C}_3$, ... $\mathscr{C}_k\}$ in M can be calculated as follows:

$$Q = \frac{1}{2\mu}\underbrace{\Big(\sum_{\mathscr{C}\in\mathcal{C}}\sum_{(i_s,j_s)\in\mathscr{C}}\Big[A^s_{(i,j)} - P_{(i,j)}\Big]\Big)}_{\text{part 1}} + \frac{1}{2\mu}\underbrace{\Big(\sum_{\mathscr{C}\in\mathcal{C}}\sum_{(i_s,j_r)\in\mathscr{C}}\Big[C^{s,r}_{i,j} * \delta(i_s, j_r)\Big]\Big)}_{\text{part 2}} \tag{3.1}$$

Where A^s is the adjacency matrix of layer s. $C^{s,r}$ is the coupling matrix that captures inter-layer edges between layers s, r so the entry $C^{s,r}_{i,j}$ corresponds to the value of the coupling strength $\omega_{i,j}$ between nodes i and j from layers s and r respectively. $\delta(i_s, j_r)$ is a Kronecker delta, and its value equals 1 if nodes i_s and j_r refer to the same actor, and 0 otherwise. μ is a normalisation factor.

The first part of equation 3.1 is the same used to calculate mono-layer modularity in equation 1.1. This part alone reaches its maximum when nodes in each layer are grouped according to their optimal mono-layer modularity. The second part of this equation is the added multi-layer ingredient to the modularity score. This part alone is maximised when the mono-layer optimised groupings are cross-merged across the layers such that all the overlapping mono-layer groupings appear together in the same multi-layer community. Accordingly, there are two forces that drive the partitioning in multi-layer networks using multi-layer modularity maximisation. The first tries to keep the node in its optimal mono-layer modularity grouping, and the second tries to group the node together with other nodes that refer to the same actor.

With that being said, the result of maximising modularity in multi-layer networks is not necessarily a clustering that groups all the nodes of an actor in one community, and at the same time does not guarantee that nodes fall in their optimal mono-layer groupings. Generally speaking, assigning low values to ω uniformly across layers produces communities such that nodes in each layer where the community expands are grouped according to their optimal mono-layer modularity. Large values given to ω will allow the gain achieved by the second part of equation 3.1 to compensate for the penalty in the first part resulted by not grouping nodes according to their optimal mono-layer modularity grouping. This allows for multi-layer communities where nodes do not necessarily fall in their optimal mono-layer grouping. In both cases, a multi-layer community resulted by maximising modularity corresponds to a community of actors where subsets of these actors interacted densely with each other but sparsely with the rest of the network in a subset of layers, and these subsets of actors overlap with each other. This means that a multi-layer community that expands over multiple layers does not necessarily correspond to a set of actors that were all involved in a community behaviour in all the layers where the community expands. For more details about the assumptions behind multi-layer modularity and the impact of that on the resulted communities, I refer the reader to a preprint of our paper "Unspoken Assumptions Behind Multi-layer Modularity" in appendix B. This article is under review by the Scientific Reports journal editors. In that article, I show how the ability of multi-layer modularity maximization based community discovery to recover ground truth communities is conditioned by a specific coupling of the nodes across different layers.

3.3.2 Multi-layer information flow mapping

The Information flow mapping method mentioned in Section 1.3.2 has been extended by [65] so it can be used to reveal multi-layer communities. In this extension, the goal is still to provide a two-level coding in order to minimise the description length of a random walker traversing the multi-layer network. On the surface, the map equation 1.2 remains the same; however, the random walker corresponds to a multi-layer dynamic as it can move between layers proportional to the weights (i.e., the coupling strength) assigned to the inter-layer links. The movements between nodes within the same layer are said to be Markovian as the position of a random walker at time t depends only on its previous position at time $t-1$. Nonetheless, movements across layers are non-Markovian as the fact that a random walker is at layer L_i does not necessarily depend on which layer it was in before. In the absence of empirical inter-layer weights, a relax rate r is used (similarly to the teleportation rate parameter used in PageRank algorithm adopted by Google to rank web pages [66]) to allow the random walker to move across layers. In a given step, the random walker stays in the same layer with a probability equal to (1-r), or moves to another layer with a probability r.

Similarly to the original method, this approach suggests a natural concept of communities in multi-layer networks as groups of nodes that capture flows within and across layers for a relatively long time. The only difference between this extension and the original method is that nodes belonging to the same actor will be assigned the same code when they appear in the same community. This implies that placing nodes that correspond to the same actor in the same community contributes positively (i.e., it is favored) to the minimisation process.

As discussed by the authors, assigning low values for r overstates the constraints set by each layer, which might result in multi-layer communities such that nodes in each layer fall in their optimal layer grouping (the one resulted by executing the method on each layer independently). On the other hand, assigning high values for the relaxation rate r washes out the constraints set by the edges of each layer for the random walker, which might favour having nodes of the same actor in the same community rather than keeping nodes in each layer in their optimal grouping. The way this parameter impacts the aggregation of mono-layer groupings seems to resemble the effect of ω in multi-layer modularity.

The intuition behind this method is that there might be multiple interests, relationships, interactions ..etc (depending on what the multi-layer network models) that bring a social group closely together. When that is the case, the random walker is expected to spend longer time within that group in the layers referring to these relationships. That leads to identifying this group as a multi-layer community that expands over these layers. When edge patterns across layers are not aligned, the resulted multi-layer communities by this method are similar, conceptually, to those discussed in the previous section (Sec 3.2).

3.3.3 Multi-layer statistical inference

An extension of community discovery using statistical inference for edge-valued, multiplex and time-dependent networks has been provided by [67]. In their extension, the authors argue that most community detection methods fail to address two substantial points: i) whether the multi-layer representation of the network is indeed important, and ii) how to separate actual structure in multi-layer networks from stochastic fluctuations.

Similar to the logic behind the original method, the goal here is to find the stochastic block model SBM that is the most probable to be the generative model for the network, in this case being the multi-layer network. In order to select a stochastic block model that both satisfies that and avoids over-fitting, the authors assume that a multi-layer network can be generated using two alternatives which they call "edge covariates" and "independent layers" respectively. The former assumes that a multi-layer network can be generated by first generating the collapsed graph and then assigning the layer membership of each edge randomly from a distribution. The latter, however, assumes that layers can be generated independently from each other. Since empirical data might follow a specific degree distribution dynamic in addition to the community dynamic, an additional consideration is given to whether the stochastic block model should consider that or not. This results in 4 different variants of stochastic block models for multi-layer networks, namely, non-degree-corrected edge covariates SBM, degree-corrected edge covariate SBM, non-degree-corrected independent layers SBM and degree-corrected independent layers SBM. The proper way to select between the alternatives is to perform model selection based on statistical significance and select one of the 4 alternatives based on sufficient evidence available in the data.

To decide whether the layered representation is important or not, the authors consider layers to be informative of the network structure if their incorporation into the stochastic block model yields a more detailed description of the data when compared to a model that is only based on the collapsed structure of the network. This can be done by considering a null model where the edges are distributed among the layers in a manner that is entirely independent of the group structure, and is constrained only by the total number of edges in each layer. We can then compare the description length of this null model with any of the other layered variants, and decide if there is enough evidence to justify the incorporation of layers that are correlated with the group structure. Since the layer variants consider groupings across layers to be relatively aligned, communities resulted by this method are constituted of actors who interacted more significantly with each other in all layers (in each one separately or in total depending on the chosen layered variant) than they did with the rest of the network.

3.4 Different models for multi-layer communities

Despite this variety of existing generative models for multi-layer communities in multi-layer networks [25, 67–75], they focus on a limited amount of structures for multi-layer communities. Indeed, many of these models produce synthetic multi-layer communities that are not straightforwardly mappable to real-world scenarios in multi-layer networks. Here I hypothesize some real-world scenarios for multi-layer communities and interpret them into different multi-layer community models. These models are possible to exist in multiplex and time-dependent networks under different conditions. While I do not claim that this list of models is exhaustive to all possible communities that can exist in multi-layer networks, I claim that they provide a good categorization of types of multi-layer communities for future work to investigate further generalizations. I refer to the nodes of an actor a in different layers with a non-zero degree in their layers as the *active nodes* of a. For a multi-layer community $\mathscr{C}$ that expands over multiple layers, I refer to the set of nodes of $\mathscr{C}$ in a single layer L as the induced nodes of $\mathscr{C}$ in L, and to the set of actors resulted by mapping each of the induced nodes of $\mathscr{C}$ in a layer L to their actors as the induced actors of $\mathscr{C}$ in L. These models are:

- **[M1] Pillar communities**: We call a multi-layer community $\mathscr{C}$ a pillar community if there exists a set of actors $\mathcal{A} = \{a_1, a_2, ...a_k\}$ such that $\mathscr{C}$ is constituted only of the active nodes of all the actors in $\mathcal{A}$ in all layers of the multi-layer network. Pillar communities result from a very high dependency in the connectivity patterns across all layers of the multi-layer network, which results in an aligned grouping of nodes across all layer.
- **[M2] Semi-pillar layer-adjacent communities**: We call a multi-layer community $\mathscr{C}$ a semi-pillar layer-adjacent community if there exists a set of actors $\mathcal{A} =$

$\{a_1, a_2, ...a_k\}$ such that $\mathscr{C}$ is constituted only of the active nodes of all the actors in $\mathcal{A}$ in a subset of layers of the multi-layer network, and these layers are adjacent to each other. Semi-pillar layer-adjacent communities usually evolve in time-dependent networks where layers refer to specific time-windows. In these networks, a set of actors might engage in the same community for a limited time and then engage in other groups in subsequent time-windows. This might result in semi-pillar communities that expand over an subset of the layers that are adjacent to each other.

[M3] **Semi-pillar non-layer-adjacent communities**: Similar to [M2], the difference with semi-pillar non-layer-adjacent communities is that the layers where the community expands are not adjacent to each other. These communities might evolve in multiplex networks where layers are categorical. These communities might also evolve in time-dependent networks if a group of actors engage in a community for a couple of consecutive time-windows, then engage in other groups at some windows, then engage again in the same grouping.

[M4] **Partially overlapping communities**: A multi-layer community $\mathscr{C}$ that expands over multiple layers is partially overlapping if the sets of the induced actors of $\mathscr{C}$ in each layer where the community expands, partially, but not completely, overlap. These communities evolve in cases when the community membership of a set of actors in one layer L_1 influences the community membership of only a subset of these actors in another layer L_2, while the membership of the rest of these actors in L_2 does not necessarily depend on their membership in L_1. Think of an example where the network is a three-layer multiplex network modelling twitter interactions (following, retweeting and replying) among a set of actors. It might be the case that the community membership of a set of actors in the ‘following’ layer influences the community membership of only a subset of these actors in the ‘retweet’ or ‘reply’ layers (i.e., user a_1 retweets user a_2 because they follow each other) while the community membership of the rest of these actors on these layers does not really depend on the ‘following’ layer.

[M5] **Hierarchical communities**: A multi-layer community $\mathscr{C}$ that expands over multiple layers is hierarchical if there is a hierarchy among the sets of the induced actors of $\mathscr{C}$ in the layers where it expands. A hierarchical community happens when the grouping of a set of actors in a layer L_1 still depends on the community membership of these actors in another layer L_2 but additional non-layer specific dependencies might result on different divisions of this grouping across the layers, which breaks the perfect cross-layer group alignment that happens in the pillar model. Think of the 3-layer multiplex modelling Twitter interactions mentioned above. A grouping of a set of actors in the 'retweeting' layer might still depend on whether they follow each other or not (i.e., user a_1 retweets user a_2 only if they follow each other), but some other user-specific properties (political affiliation for example) might result on multiple divisions of their groupings across the two layers.

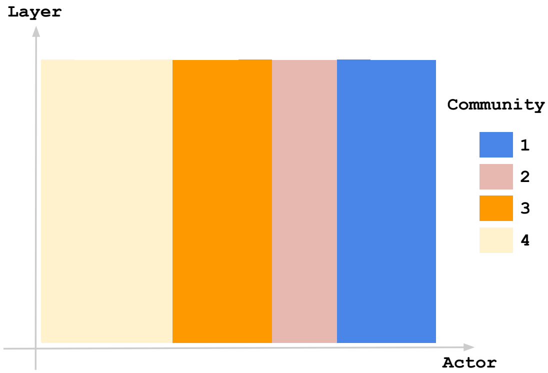
Figure 3.7 provides a colored illustration of the distribution of community memberships in each of the models described above. Different colors refer to different community memberships. *NS* in figures 3.7b and 3.7c means that the community membership of these actors in the corresponding layers is not specified in this illustration as it is independent from the community membership in other layers.

3.5 Review of community detection in multi-layer networks

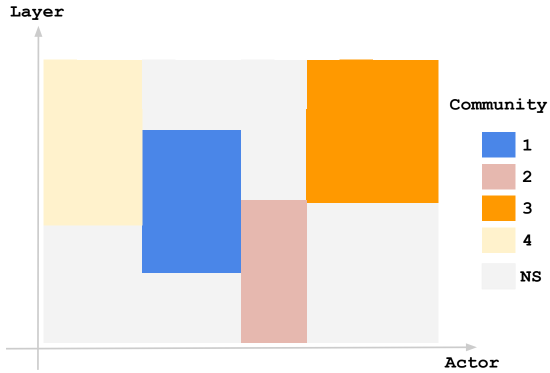
Here I refer to our review of community detection methods in a multiplex networks reported in Appendix A. In this article, which has been submitted to Computing Surveys journal (CSUR) and is still under review, we provide the following contributions to the literature:

- On a theoretical level, it provides the reader with a viable taxonomy of multiplex community detection methods based on the way they handle the multiplexity problem and the mathematical tools used to identify multiplex communities.
- On a practical level, it reports the results of an extensive evaluation and a comparative analysis among the most relevant state-of-the-art methods trying to answer three main questions: **a)** to what extent are the evaluated methods able to detect ground truth communities, **b)** to what extent do different methods identify similar community structures, and **c)** to what extent are the evaluated methods scalable.
- It elaborates on the definition of multiplex communities on a structural level by providing eight different types of multiplex communities that can exist in network, and then tries to answer which of these structures is poorly considered by the available methods.

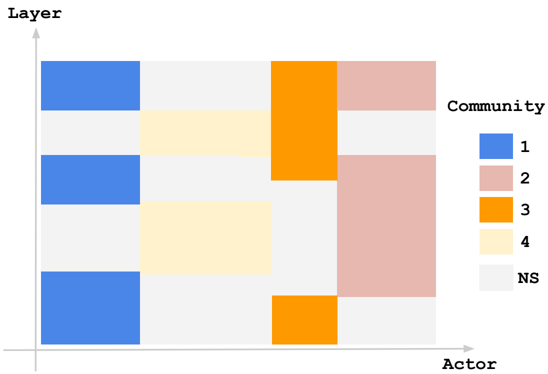
Our main finding in this survey is that "pillar" communities, where all nodes representing the same entity on different layers/relations are grouped within the same community, can be well detected with the methods we have available, with information flow mapping and modularity maximization achieving very good results. Nevertheless, the more we move



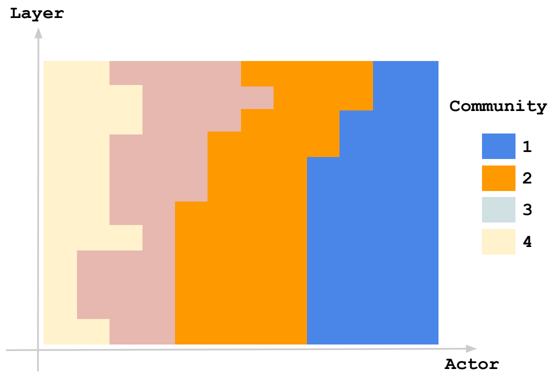
(A) Pillar communities



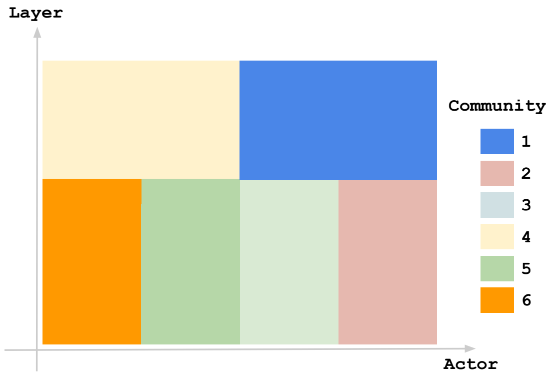
(B) Semi-pillar layer-adjacent communities



(C) Semi-pillar non-layer adjacent communities



(D) Partially overlapping communities



(E) Hierarchical communities

FIGURE 3.7: Different models of multi-layer communities, illustrated using colors referring to the actors' community memberships across different layers. *NS* in figures 3.7b and 3.7c means that the community membership of these nodes is not specified in this illustration as it is independent from the community membership in other layers.

away from that ideal model of multiplex community structure, the more the accuracy drops and the differences between various algorithms become apparent. Based on our review of the literature and our activities in the field, we think that this article is timely and opportune as it will both help existing research as well as generate future research.

3.6 Applications of multi-layer community detection

The multi-layer network model has been applied to investigate many real-world complex systems. Here I refer to two of our contributions in that direction, reported in Appendices D and E. The first is titled "From Interaction to Participation: The Role of the Imagined Audience in Social Media Community Detection and an Application to Political Communication on Twitter" and was published in the proceedings of ASONAM 2018. The second is titled "An Innovative Way to Model Twitter Topic-Driven Interactions Using Multiplex Networks" and was published in a workshop proceedings of the 13th International AAAI Conference on Web and Social Media. Both papers test the proposed models on a dataset of Twitter content generated by Danish politicians during the month leading to the 2015 Danish parliamentary elections. I will refer to this dataset as the DkPol dataset, and it can be found at the following link [https://github.com/obaidaITU/dkpol]. In the following, I briefly explain both contributions in Section 3.6.1 and Section 3.6.2 respectively.

3.6.1 Modeling Interactions and Participation on Twitter

A common way to model the multiple types of relationships supported by a social media platform is to use multiplex networks where different layers refer to different interactions provided by that platform (for example, replying and retweeting). As discussed in Section 2.1, members of a social group are expected to share similar interests in addition to their frequent interactions over time. With that idea in mind, we add a layer representing the users' interests. This layer, which we call the Topical Audience Model (TAM), aims at modeling the shared interests among users based on their participation in public discussions. We build that layer in two steps. In the first step, the discussions among the users of interest are modeled as a multiplex of n layers, where n is the number of topical discussions to be considered in the model. In the context of this paper, we use the explicit hashtag as a proxy for the topic of the shared conversation as suggested by [76]. Each discussion adds a layer to that multiplex and is modeled as a single clique that ties all the users who were part of the discussion – that is, those who included the same hashtag in their messages. In the second phase, we compute a single network by applying a weighted flattening [77]. A threshold θ can be used here to filter out edges with less then a specific weight. We perform community discovery on the DkPol dataset in two cases, one by considering only the retweets among the politicians, and the other is by considering retweet, reply and the shared interests layers we propose. Figure 3.8 provide an illustration for the proportion of political coalitions (Red Block and Blue Block) within the communities detected on both cases. As the figure shows, the first case (only the retweet) produced 9 communities that are more homogeneous in their composition of political coalitions, while in the case of the multiplex we have 10 communities some of which are less homogeneous.

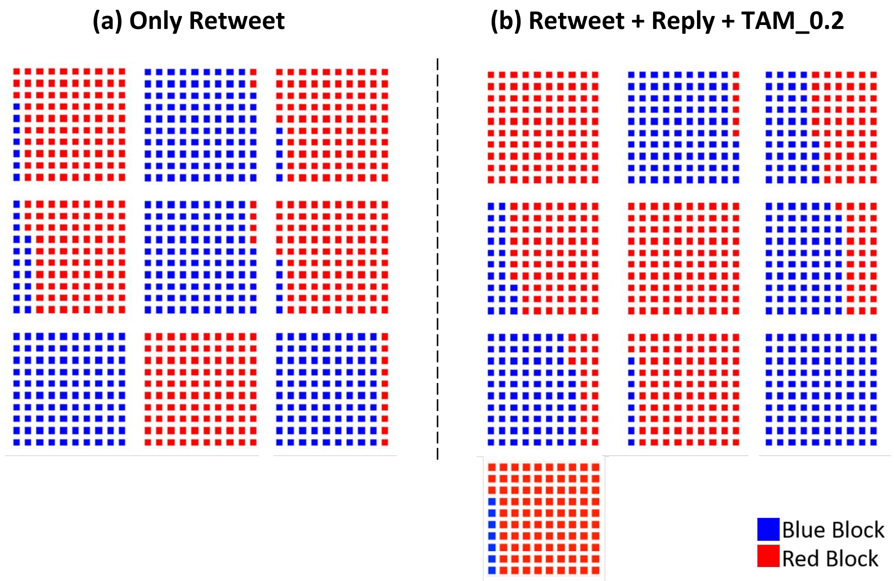
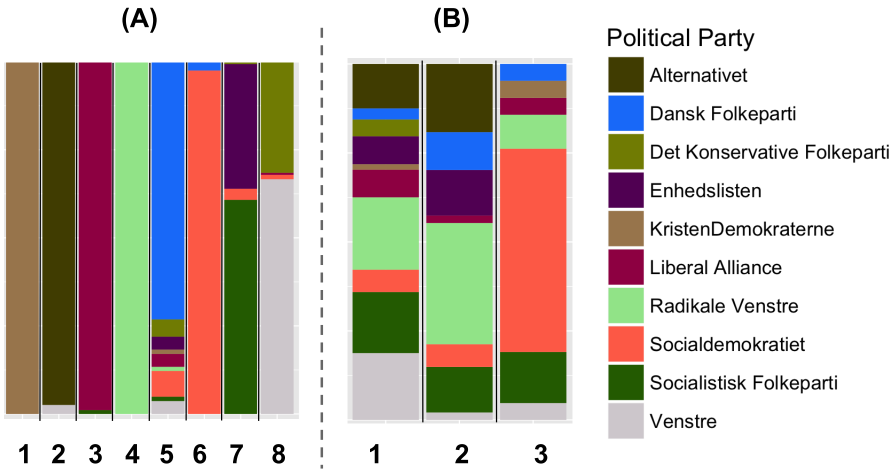


FIGURE 3.8: Proportion of political coalitions (Red Block and Blue Block) within the communities detected on both a) only the retweet network b) the multiplex network including retweets, replies and the TAM with $\theta = 0.2$

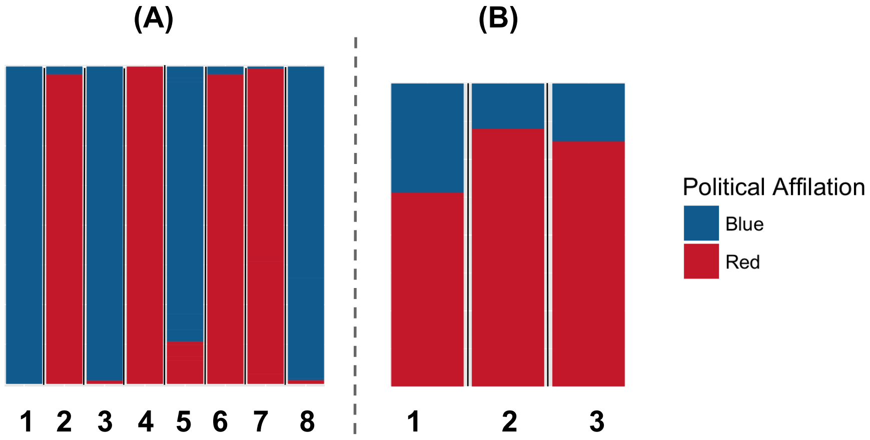
.

3.6.2 Modeling topic-driven interactions on Twitter

In this work we propose a way to model topic-based implicit interactions among Twitter users. This model relies on grouping Twitter hashtags into themes/topics, and then using the multiplex network model to construct a multiplex network (we call it the thematic multiplex), where each layer corresponds to a topic or theme, and users within a layer are connected if and only if they have used the same hashtag. By testing our model on the DkPol dataset, we show that applying multi-layer community detection on the thematic multiplex can reveal different types of communities that differ from those observed when modeling Twitter interactions. Figure 3.9 illustrates the difference between communities resulting from two different multiplex networks.



(A) Coloring based on political parties



(B) Coloring based on political coalitions

FIGURE 3.9: The resulted communities by applying multi-layer community detection on two different multiplex networks over the DkPol dataset: **(A)** the multiplex consisting of the three layers (following/follower, retweet and reply), and **(B)** the thematic multiplex. Each bar refers to a different community, while the colors in each bar (i.e., community) refer to the composition of each community with respect to the political affiliation (3.9a) or the political coalition (3.9b) of the members constituting it.

We use the thematic multiplex to perform a longitudinal analysis of the communities formed based on the topical interests. That is achieved by dividing the data into four time windows of length equal to one week, then constructing a thematic multiplex out of the data within each time window. We perform multi-layer community detection on the resulting multiplex networks separately, and we study the frequency of each topic/theme in each community over the month leading to the elections. Figure 3.10 illustrates the relationship between the topics and the thematic communities resulted by applying discovery detection on the 4 thematic multiplex networks.

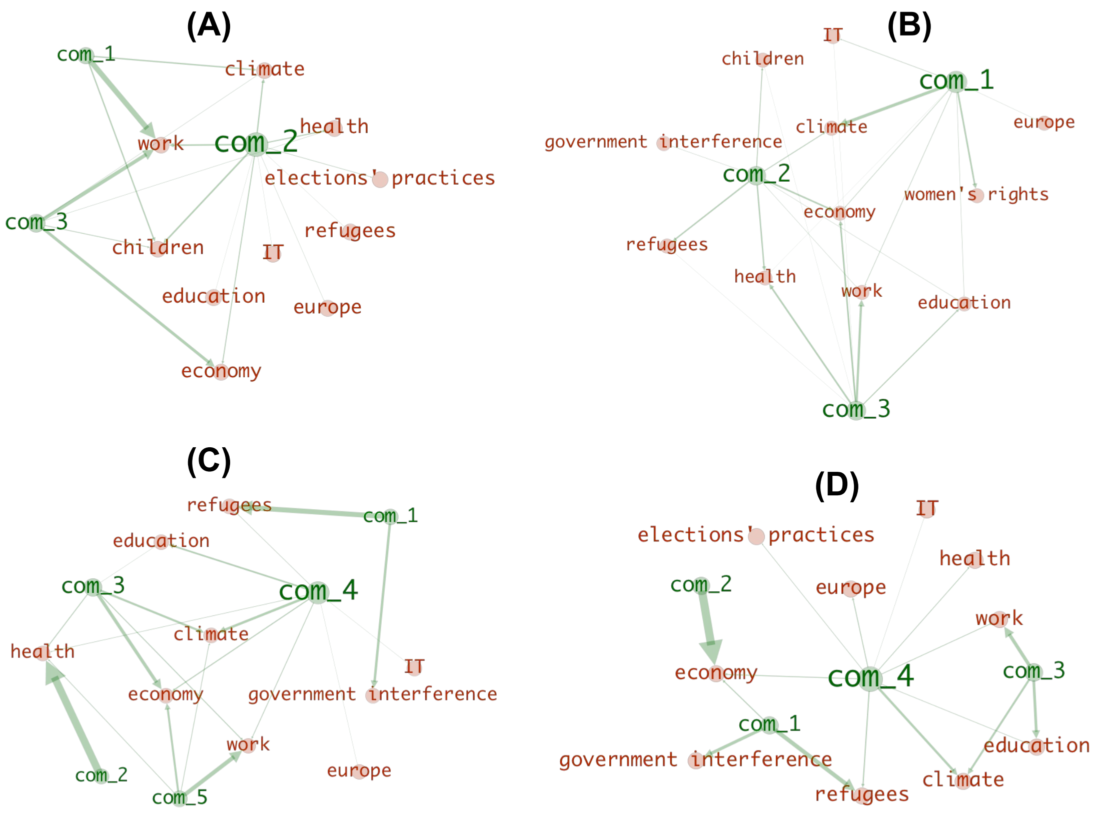


FIGURE 3.10: The relationship between the topics and the thematic communities resulted by applying community detection on 4 thematic multiplex networks that captured the Twitter interactions of Danish politicians during the month leading the parliamentary elections of 2015 (one per week). **(A)** week 1, **(B)** week 2, **(C)** week 3, **(D)** week 4
.

In Brief

- The multi-layer network model is a generalization of the notion of the graph to model different aspects of a connection between some actors, such as different types of relationships, or different points in time when a relationship happens.

- The key difference between the multi-layer graph model and other previous graph generalisations that consider assigning properties to edges/nodes of the graph is the introduction of inter-layer edges - that is, the fact the connection patterns in one layer might depend on another layer.

- Two special applications of the multi-layer model that received a great deal of attention in the literature are modeling *multiplex* networks–that is, networks that capture different types of relationships among a set of actors, and modeling *time-dependent* (or temporal) networks– that is, networks in which the interactions among actors change over time.

- A multi-layer community detection task in multi-layer networks is not only expected to use the intra-layer information to provide a meaningful grouping of the nodes in their layers, but to also use the dependencies among edge patterns in different layers to provide a meaningful aggregation of them into multi-layer communities.

- Despite the plethora of multi-layer community discovery methods, it not always clear what patterns in multi-layer networks is a multi-layer they reveal, and in which way these patterns map to qualitatively meaningful communities.

- Pillar communities, where all nodes representing the same entity on different layers/relations are grouped within the same community, can be well detected with the methods we have available, with information flow mapping and modularity maximization achieving very good results. Nevertheless, the more we move away from that ideal model of multiplex community structure (i.e., the pillar one) the more the accuracy drops and the differences between various algorithms become apparent.

Chapter 4

Discussion and Future work

In the light of what has been presented in this thesis, this chapter provides a discussion about community discovery in social networks in general and in multi-layer networks in particular in Section 4.1. Based on that, some suggestions to generate future research are provided in Section 4.2.

4.1 Discussion

This thesis provides a critical and practical look at the problem community discovery in social networks in general, and in multi-layer networks in particular. It reviews the similarity metrics used in the literature for evaluating community discovery solutions, analyses their sensitivities to different types of dissimilarities among clusterings and gives useful guidelines on which metric to use for the task at hand (Section 1.4 and Appendix C). It provides a theoretical and experimental similarity analysis among three popular mono-layer community discovery methods, namely, modularity maximization, information flow mapping and statistical inference using stochastic block models (Sections 1.5.1 and 1.5.2). The similarity has been studied as a function of three different parameters, whether or not the network is directed, whether or not the network is weighted and the level of noise in the network. To be equally distanced from the different disciplines that studied the notion of community in social contexts, the thesis provides a review of the definitions of 'Community' and 'Social Group' in social sciences (Section 2.1) and point out to different conceptualizations for community discovery in social networks and their mapping to some community discovery methods (Section 2.2). It discusses community discovery in multi-layer networks and problematizes the absence of conceptualizations for multi-layer communities into specific and well-defined multi-layer network patterns (Section 3.2). It unfolds the mathematical formulations of three popular multi-layer community discovery methods, namely, multi-layer modularity maximization, multi-layer information mapping and multi-layer statistical inference to understand their implied definition of multi-layer communities (Section 3.3). Being one of the extensively used methods for multi-layer community discovery, the thesis investigates hidden assumptions behind multi-layer modularity maximization in Appendix B. By hypothesizing some real-world scenarios for multi-layer communities, the thesis proposes different multi-layer community models that can be used to test and compare the performance of multi-layer community detection methods (Section 3.4). Because multiplex networks have received a good deal of attention for their application in many disciplines, a systematic review and an extensive similarity analysis of community discovery methods in multiplex networks is performed in this thesis (Section 3.5 and Appendix A). The thesis benefits from the multi-layer framework and its applications in multiplex networks to suggest an innovative model for topic-based interactions on Twitter (Section 3.6.2 and Appendix E). It also shows a possible incorporation of users' participation in public discussions on social media into a multiplex network that models direct interactions among users in addition to their topical interests (Section 3.6.1 and Appendix D).

As shown throughout the thesis, the task of community discovery in social networks is not completely parameterless. On the contrary, there are plenty of decisions to make throughout that process and each of these decisions may largely affect the recovered patterns and the way they can be interpreted. As shown in Figure 4.1, a community discovery task typically goes through four phases, namely, **(i)** the modeling phase, **(ii)** the community discovery phase, **(iii)**, the evaluation phase, and **(iv)** the interpretation phase. In each of these phases, there are plenty of available choices in the literature without clear guidance on why choosing one and neglecting the others is preferred. On one hand, this sheds a light on the importance of investigating a clear and precise reasoning behind each choice and the impact that has on what can be claimed about the recovered patterns qualitatively. On the other hand, one might question, to which extent this availability of choices without a clear guidance affects the reliability of community discovery as a tool for testing hypothesis?. In my personal humble opinion, the availability of such choices at each step provided, to some extent, a tunable black box, the community discovery tool, which, in many cases, can be tuned to support certain hypothesis and reject others.

A possible choice in the modeling phase is the multi-layer model. This model is motivated by the fact that different relationships/interactions in complex systems are interdependent on each other and neglecting these dependencies can yield to misleading or incomplete findings. The question, however, is whether the multi-layer model really

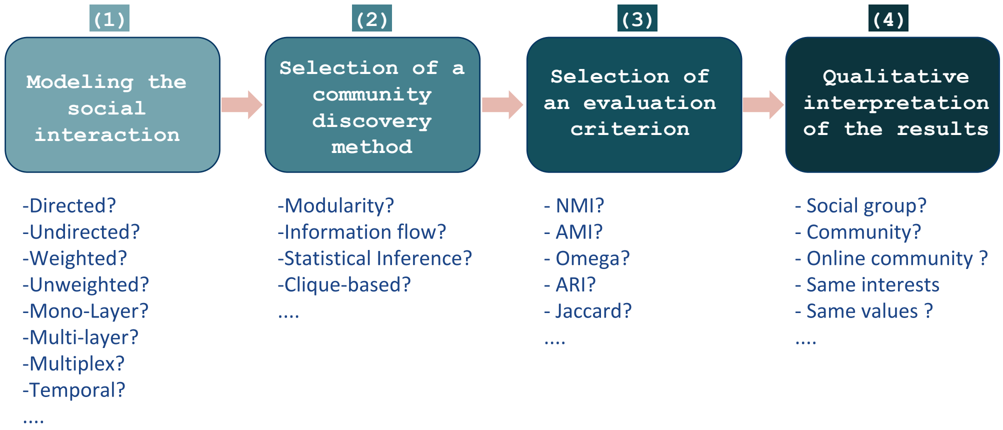


FIGURE 4.1: A pipeline of the main phases a community discovery task typically goes through

leverages information about these dependencies as it promises, and/or whether multi-layer network analysis tools are designed to take into consideration different levels of dependencies across layers other than the uniform full dependency (i.e., the one that results the pillar community structure). For example, assume a multiplex network constituted of two layers L_1, L_2 that are not completely dependent on each other but with a dependency p equals to 0.5 (meaning that the existence of an edge between two actors in one layer might cause an edge between them in the other layer with a probability equals to 0.5). This means that part of the edges in L_1 are caused by edges in L_2 and the rest are independent, and vice versa. For an accurate recovery for the ground truth communities in this case, a multi-layer community discovery task should be able to differentiate between the two types of edges. Here I ask whether the inter-layer edges in the multi-layer model are meant to leverage such information about the dependencies and if yes whether multi-layer community discovery treats inter-layer edges as such. This is to say that it is of great importance for future research to investigate the meaning of inter-layer edges in the multi-layer model, what information they convey about the modeled system and whether that is aligned with how different multi-layer network analysis tools in general and community discovery in particular process them.

As shown in our review of multiplex community discovery, the more we move away from the pillar community model, the more the accuracy drops and the differences between various algorithms become apparent. Indeed, the lack of clear conceptualizations of multi-layer communities into multi-layer edge patterns and lack of research on which layers to include in a multi-layer network for community discovery tasks makes it hard to tell whether this disagreement comes from different interpretations for multi-layer communities, or different responses to the existence of layers that do not positively contribute to the community structure.

Multi-layer community discovery has provided a tool to reveal community structures that can not recovered by analyzing the layers separately or in a collapsed graph. While research on synthetically generated multi-layer networks provided multiple examples to support that claim, there is no enough evidence yet that patterns recovered by multi-layer community discovery in real-world data correspond to qualitatively meaningful clustering of the nodes. Most of the research on real-world multi-layer networks provided a ground truth clustering and showed that the layered representation is better compared to the other variants in its ability to recover patterns that are closer to the ground truth. Apart from the fact the ground truth assumption itself can be questionable as discussed by [23], this approach does not consider the by-chance factor that might favor the layered representation only because its ability to recover the ground truth communities is by chance.

4.2 Future work

The multi-layer model has provided a useful tool to model a variety of complex systems. This allowed the task of community discovery to reveal network patterns that can not be revealed by analyzing the layers separately or integrating them in a collapsed graph. In the light of what is discussed in the previous section, I invite more research in the following areas

Layer selection in multi-layer networks: While the layered representation may reveal community structures that can not be revealed should the relationships had been analyzed separately or integrated in a collapsed graph, it can also obscure the community structure in some cases where the layered representation is not necessary as discussed by [67]. The work proposed by the authors provides a significance test to help deciding whether the layered representation is preferred over the collapsed one or not. More research in this direction can be done to select the layers that contribute positively to a community structure (not necessarily a pillar one) before rejecting the layered representation assumption.

Conceptualizing multi-layer communities to multi-layer edge patterns: More importantly than proposing new multi-layer community discovery methods, it is very important to conceptualize what a multi-layer community is first in terms of multi-layer edges patterns (as done in section 2.2 for mono-layer communities). This step is trivial to be able to provide consistent qualitative interpretations for the outputs of multi-layer community discovery methods.

Customized coupling strategies The fact that most multi-layer community discovery methods reward the co-occurrence of two connected nodes of an actor in one community makes the coupling type chosen to couple nodes across different layers of great importance. Two types of coupling strategies have been used in the literature, namely, *complete* coupling, which is usually used with multiplex networks and it couples each pair of nodes of an actor together, and *adjacent* coupling that is used with time-dependent networks and it couples each node with its replica in the previous layer. While this way of coupling seems to be intuitive, when the community structure is different from the pillar model, these uniform coupling strategies can force communities to expand over multiple non-dependent layers and the resulted communities would be an artifact of the model rather than actual multi-layer communities. More research on proposing more complex non-uniform coupling strategies is needed so it prevents the multi-layer model from forcing/favoring specific community structures.

Appendix A

Community Detection In Multiplex Networks

Community Detection in Multiplex Networks

OBAIDA HANTEER, IT University of Copenhagen
ROBERTO INTERDONATO, CIRAD, UMR TETIS
MATTEO MAGNANI, Uppsala University
ANDREA TAGARELLI, University of Calabria
LUCA ROSSI, IT University of Copenhagen

A multiplex network models different modes of interaction among same-type entities. A great deal of attention has been devoted to the community detection problem in multiplex networks, that is, revealing meaningful patterns of node groupings into communities by considering different types of interactions among them. In this article, we provide the reader with a taxonomy on the community detection algorithms in multiplex networks. We characterize the different algorithms based on various properties and we discuss the type of communities detected by each method. We then provide an extensive evaluation of the reviewed methods according to different criteria trying to answer three main questions - to what extent the evaluated methods are able to detect ground truth communities, to what extent different methods produce similar community structures and to what extent the evaluated methods are scalable. The ultimate goal of this survey is to drive scholars and practitioners in choosing the right method for the data and the task at hand.

CCS Concepts: • **General and reference → Surveys and overviews**; • **Theory of computation → Social networks**;

Additional Key Words and Phrases: Community detection, multiplex networks, multiplex community detection

ACM Reference format:
Obaida Hanteer, Roberto Interdonato, Matteo Magnani, Andrea Tagarelli, and Luca Rossi. 2019. Community Detection in Multiplex Networks. *ACM Comput. Surv.* 42, , Article (January 2019), 39 pages.
https://doi.org/

1 INTRODUCTION

. Multiplex network analysis emerged as a promising approach to investigate complex systems in the real world. A multiplex network is a compact network model used to represent multiple modes of interaction or different relationships among entities of the same type (e.g. people). These models have been used to study a large variety of complex systems across different disciplines ranging from living organisms and human societies to transportation systems and critical infrastructures. For example, a reasonably effective description of the full protein-protein interactom[1] in the human body requires, for some organisms, up to seven distinct modes of interaction among thousands of protein molecules [12]. Another example is in air transportation systems when modeling the

[1]an interactom is the totality of protein–protein interactions that happen in a cell, an organism or a specific biological context

0360-0300/2019/1-ART $15.00
https://doi.org/

connections between airports through direct flights; here, the different commercial airlines can be seen as different modes of connections among airports [9].

Figure 1 shows a typical layered representation of a multiplex network, where each of the layers corresponds to a type of interaction, and nodes in different layers can be associated to the same actor, e.g. the same person or the same airport. Here, we adopt the term *actor* from the field of social network analysis, where multiplex networks have been first applied, and the term *layer* from recent generalizations of the original multiplex model [13]. Let $A = \{a_1, a_2, \ldots, a_m\}$ be a set of *actors* and $L = \{l_1, l_2, \ldots, l_n\}$ be a set of *layers*, where each layer represents a different type of relationship among actors in A. Layers are not required to contain all actors. The existence of an actor in a layer is represented as a unique node in that layer. When an actor is existent in two or more layers, this is represented as distinct nodes (one per layer), with inter-layer edges that connect them to denote that they belong to the same actor. A *multiplex network* is defined as a graph $G = (V, E)$ where $V \subseteq A \times L$ is the set of nodes and E is a set of intra-layer edges connecting nodes on the same layer.

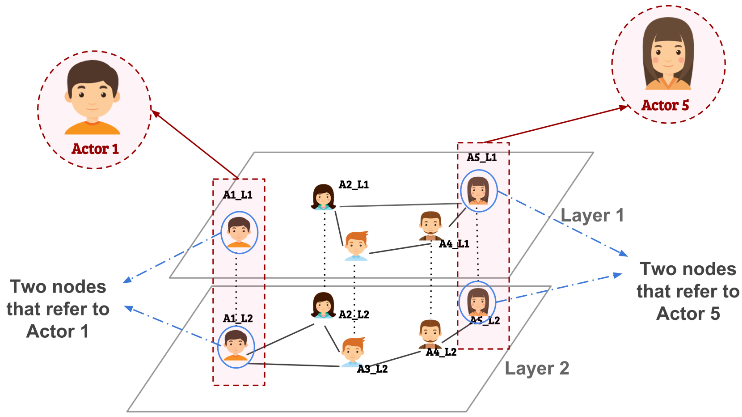


Fig. 1. A toy-example of a multiplex network with two types of interaction among five actors. This is represented as five nodes replicated in two layers. The two nodes representing the same actor (e.g. the same person) are linked by a dotted line

. A core task in social network analysis is to identify and understand communities, also known as clusters or cohesive groups; that is, to explain why groups of entities (actors) belong together based on the explicit ties among them and/or the implicit ties induced by some similarity measures given some attributes of these entities. Since members of a community tend to generally share common properties, revealing the community structure in a network can provide a better understanding of the overall functioning of the network. Community detection methods on simple graphs are not sufficient to deal with the complexity of the multiplex model. Thus, multiplex community detection was introduced to leverage various interaction modes among the studied entities when grouping them into communities and give more qualitatively interpretable results.

. Given the emergence of multiplex networks and the importance of the community detection task in network analysis, several community detection algorithms for multiplex networks have been recently proposed, based on different definitions of community and different computational approaches. This article provides a systematic review and experimental comparison of existing methods to simplify the choice and the set up of the most appropriate algorithm for the task at hand. At the same time, our review will highlight weaknesses and strengths of specific methods and of the current state-of-the-art as a whole.

. Recent works have provided a partial overview of the field. [22] proposed some criteria to compare multi-layered community detection algorithms, but without providing experimental evaluation. Similarly, [6] highlighted the conceptual differences among different clustering methods over attributed graphs, including edge-labeled graphs that can be used to represent multiplex networks, but only provided a taxonomy of the different algorithms without any experimental analysis. [28] instead performed a pairwise comparison of the different clusterings produced by some existing algorithms.

. In this work we provide a theoretical as well as experimental comparison of existing multiplex community detection methods, including several algorithms and approaches not covered by previous studies. Multiplex community detection algorithms are organized into a taxonomy and characterized by various properties making it easier for practitioners to decide which algorithm to use for the data and task at hand. We also analyze the accuracy of the different methods with respect to a given ground-truth on both synthetic and real-world networks and we study their scalability in terms of the size of the multiplex vertically (i.e., number of layers) and horizontally (i.e. number of actors).

. The focus of this survey is on algorithms explicitly designed to discover communities in multiplex networks through the analysis of the network structure. Several community detection algorithms have been proposed that deal with multi-relational network models which, however, are not compliant with the multiplex model we are taking into account. Notable examples include algorithms designed to deal with networks that allow different types of actors, e.g., Heterogeneous Information Networks [37–39, 45] and Bipartite Networks [2, 17]. Since we focus on methods exploiting networks' topology, graph clustering on attributed multi-layer networks [5, 27, 32, 34, 35, 43, 46] will not be included in our analysis. For a survey on attributed graph clustering we refer the reader to [6].

. The rest of this work is organized as follows. In Section 2 we introduce a taxonomy of existing multiplex community detection methods. Section 3 discusses different types and properties of multiplex communities. Section 4 presents the experimental settings and the evaluation datasets used in our experiments for which the results are provided in Section 5. We summarize our main findings and indicate usage guidelines emerged from our experiments in Section 6. Finally, Section 7 concludes the survey.

2 A TAXONOMY OF THE REVIEWED ALGORITHMS

In this section we provide a taxonomy of multiplex community detection methods. We classify the main existing algorithms according to this taxonomy. Figure 2 and Table 1 show an overview of the related methods.

. Our taxonomy is constituted of three levels of comparison among multiplex community detection methods. The top-level distinction answers whether the method is *global* or *local*. Global methods are designed to discover all possible communities in a network, thus requiring knowledge on the whole network structure. Conversely, local methods (also known as *node-centric*) are *query-dependent*, i.e., they are designed to discover the (local) community of a set of query nodes as an input.

. The second level of distinction concerns how the method solves the multiplexity problem. Three main approaches, illustrated in Figure 3, have been used to solve this problem. The first approach, *flattening*, consists of simplifying the multiplex network into a simple graph by merging its layers, then applying a traditional (i.e., designed for simple graph) community detection algorithm. The second approach, *layer-by-layer*, consists of processing each layer of the multiplex network

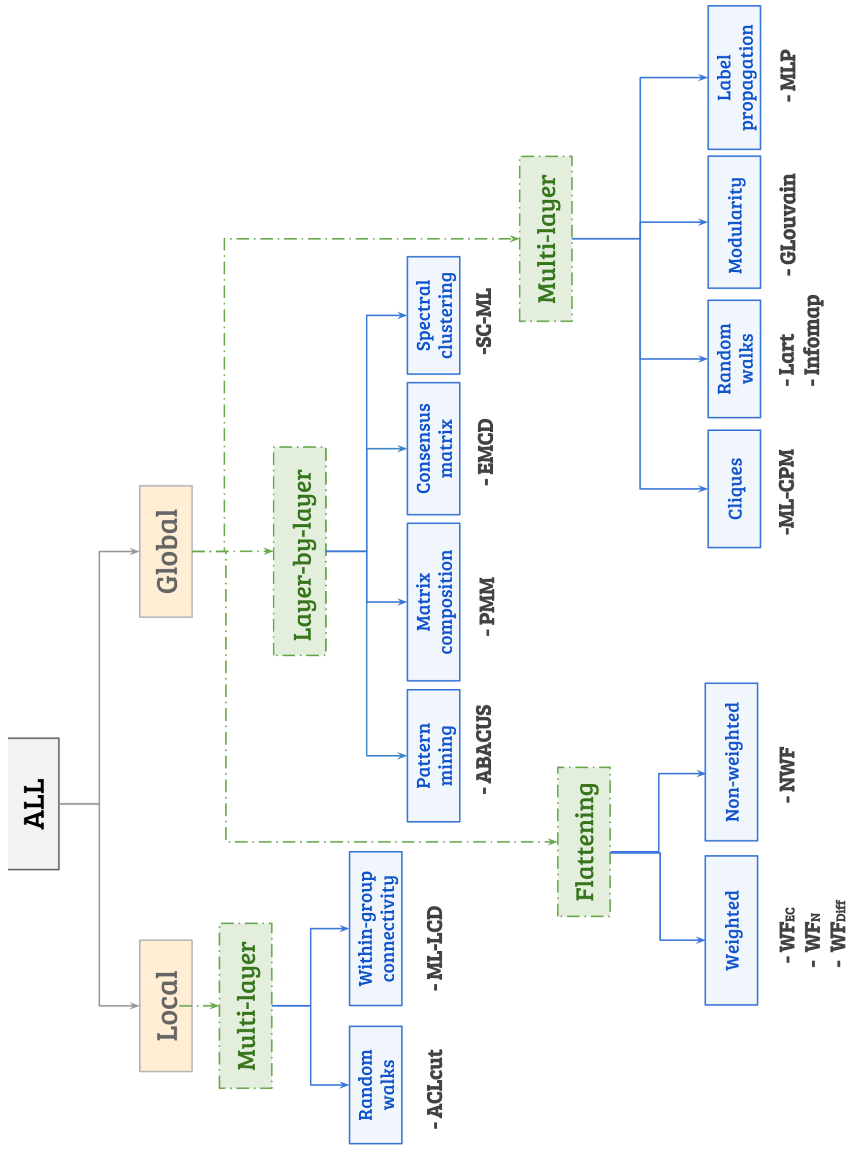


Fig. 2. A taxonomy of multiplex community detection algorithms

separately, then aggregating the resulting solutions. The third class of algorithms, *multi-layer*, operates directly on the multiplex network model.

. The third level in our taxonomy corresponds to a fine-grained distinction that defines the mathematical tools used to identify the multiplex communities. *Flattening* branch distinguishes between *non-weighted flattening* and *weighted flattening*. The latter reflects some structural properties of the multiplex network in the form of weights assigned to the output simple-graph edges, and then uses a traditional community detection method for weighted graphs. *Layer-by-layer* has four branches: *pattern mining*, *matrix composition*, *consensus matrix* and *spectral clustering*.

Table 1. Multiplex community detection algorithms covered in this survey

Algorithm	Notation	Reference
Non-Weighted Flattening	NWF	[3]
Weighted Flattening (Edge Count)	WF_EC	[3]
Weighted Flattening (Neighbourhood)	WF_N	[3]
Weighted Flattening (Differential)	WF_Diff	[23]
Cluster-Based Similarity Partitioning Algorithm	CSPA	[42]
Canonical Correlations Analysis	CCA	[42]
Frequent pattern mining-based community discovery	ABACUS	[4]
Subspace Analysis on Grassmann Manifolds	SC-ML	[14]
Ensemble-based Multi-layer Community Detection	EMCD	[40]
Principal Modularity Maximization	PMM	[41]
Generalized Louvain	GLouvain	[21]
Locally Adaptive Random Transitions	LART	[24]
Multi Layer Clique Percolation Method	ML-CPM	[1]
Modular Flows on Multilayer Networks	Infomap	[11, 15]
Multi Dimentional Label Propagation	MLP	[7]
Multilayer local community detection	ML-LCD	[19]
Andersen-Chung-Lang cut	ACLcut	[20]

The former detects communities in each layer separately using a simple-graph community detection, then makes use of pattern mining algorithms to aggregate the resulting communities. The matrix-composition-based methods identify structural features from each layer of the network via modularity analysis, and then integrate them to identify community structure among actors. The consensus-matrix-based methods combine multiple solutions over the various layers to infer a single community structure that is representative of the set of layer-specific community structures. Finally, the spectral-clustering-based methods combine the characteristics of individual graph layers to form a low dimensional representation of the original data, then makes use of spectral clustering to identify the multiplex communities. *Multi-layer* includes *clique*-based methods, which exploit the concept of multi-layer clique to identify multiplex communities, *random walk*-based methods, which introduce a multi-layer random walker that can traverse interlayer edges, *modularity*-based methods, which define a multi-layer modularity function and optimize it to produce the community structure solution, *label propagation* methods, which utilize a multi-layer affinity measure among actors given their connections on different layers and then use a labeling method for the actors controlled by these affinity scores, and *within-group connectivity* for local methods, which define a multi-layer within-group connectivity function for the multiplex community and try to maximize that function.

3 DEFINITION OF MULTIPLEX COMMUNITY

The output of a community detection algorithm for multiplex networks is a set of communities $C = \{C_1, C_2, \ldots, C_k\}$ such that each community contains a non-empty subset of V. We will also use *cluster* as a synonym of community, when it is clear from the context that we are referring to the set of nodes assigned to a community, not to the subgraph underlying a community; an analogous remark applies to *clustering* as a set of communities. Figure 4 illustrates different possible types of clusterings on a multiplex network.

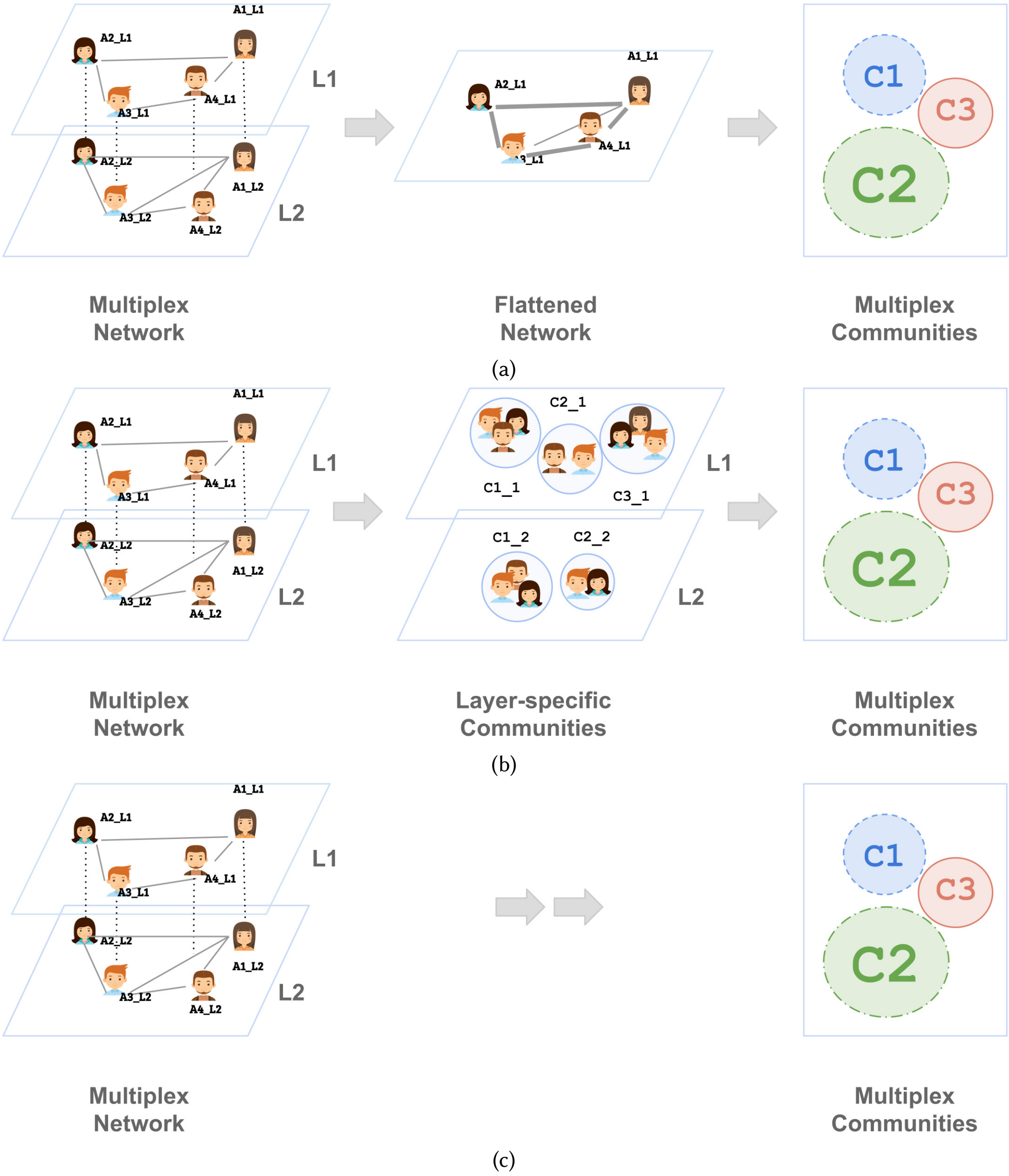
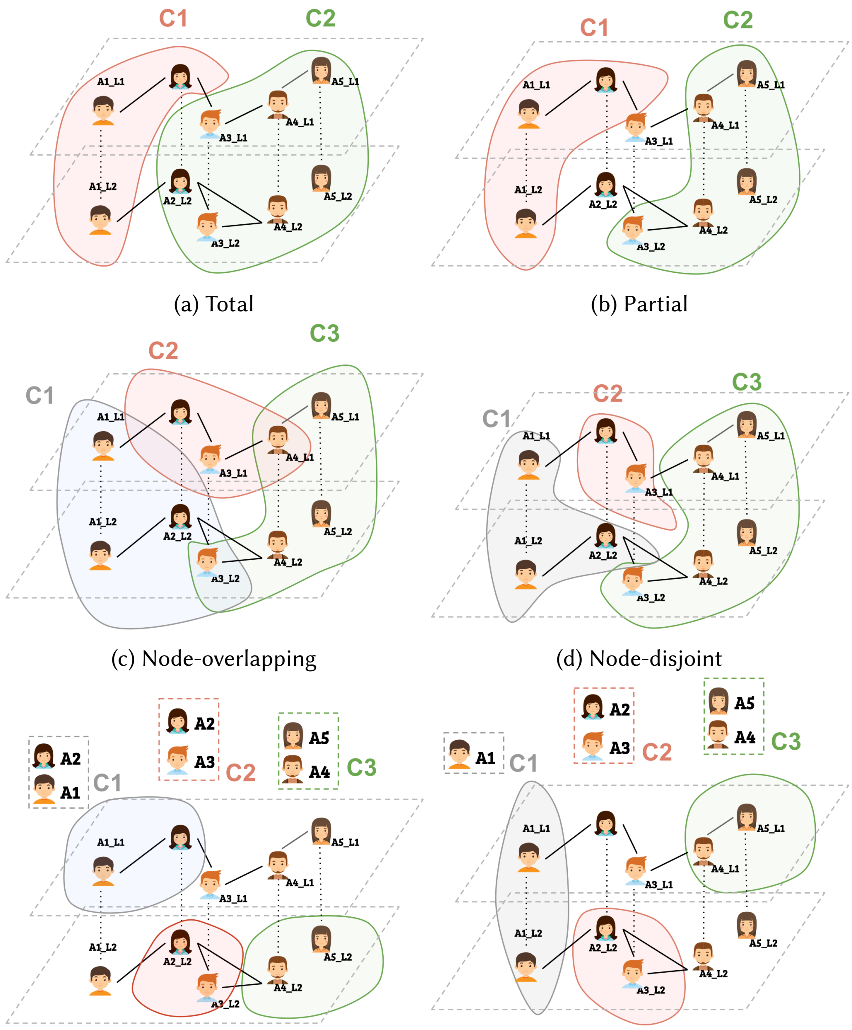


Fig. 3. Three main global approaches to discover communities in multiplex networks. **a)** Flattening approach, **b)** layer by layer approach, and **c)** multi-layer approach

. A clustering C is **total** if every node in V belongs to at least one community, and it is **partial** otherwise. We also call a clustering **node-overlapping** if there is at least a node that belongs to more than one cluster, otherwise the clustering is called **node-disjoint**. Analogously, if there is at least an actor belonging to more than one cluster we call the clustering **actor-overlapping**, otherwise it is called **actor-disjoint**. Notice that a node-overlapping clustering is also actor-overlapping,



(a) Total (b) Partial

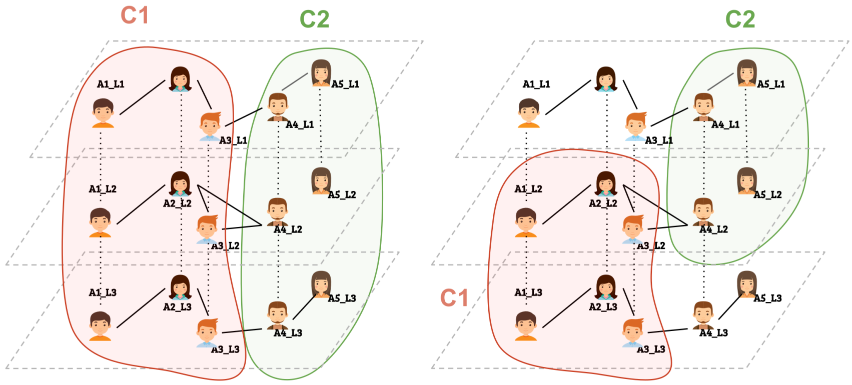
(c) Node-overlapping (d) Node-disjoint

(e) Actor-overlapping (f) Actor-disjoint

Fig. 4. Different types of clustering on a multiplex network

while an actor-overlapping clustering may or may not be node-overlapping. Table 2 reports the type of clustering produced by each reviewed method. In addition, a multiplex community is called **pillar** if all the nodes of each actor included in the community are covered by that community and **semi-pillar** if the majority (but not all) of the nodes of each actor included in a community belong to that community (Figure 5).

. In their survey work, [22] discussed a classification framework based on a set of desired properties for multilayer community detection methods. These properties are: multiple layer applicability, consideration of each layer's importance, flexible layer participation (i.e., every community can have a different coverage of the layers' structure), no-layer-locality assumption (e.g., independence from initialization steps biased by a particular layer), independence from the



(a) Pillar communities (b) Semi-pillar communities

Fig. 5. Pillar and semi-pillar multiplex community structures

Table 2. Types of clustering featured by the reviewed methods. (*) indicates that the answer depends on the single-layer clustering algorithm used by the method.

Algorithm	Actors	Nodes	Coverage
NWF	*	*	*
WF_N	*	*	*
WF_Diff	*	*	*
CSPA	Disjoint	Disjoint	Total
CCA	Disjoint	Disjoint	Total
ABACUS	Overlapping	Disjoint	Partial
SC-ML	Disjoint	Disjoint	Total
EMCD	Disjoint	Disjoint	Total
PMM	Disjoint	Disjoint	Total
GLouvain	Overlapping	Disjoint	Total
LART	Overlapping	Disjoint	Total
ML-CPM	Overlapping	Overlapping	Partial
Infomap	Overlapping	Overlapping	Total
MLP	Disjoint	Disjoint	Total

order of layers, algorithm insensitivity, and overlapping layers (e.g., two or more communities can share substructures over different layers).

. We observe that the first of the above listed properties (i.e., multiple layer applicability) should be true for all methods, therefore we do not elaborate on this further. By contrast, the second property (i.e., consideration of each layer's importance) deserves a clarification. Also, both properties about independence from node/layer order express a non-deterministic behavior of the community detection method, therefore we will treat them as a single property. The insensitivity property (i.e., independence or robustness against main tunable input parameters) is instead specialized into two properties: one referring to whether the number of communities is automatically derived or is an input parameter, and one about whether an output multiplex community may contain a subset of the layer-relations (instead of just being an induced subgraph of the community node set).

In light of the above considerations, we next define four properties in addition to the characteristics previously discussed in Section 3. Table 3 organizes the reviewed methods according to the four properties.

- **P1: Layer relevance weight distribution.** Some methods may take into consideration each layer's importance, thus allowing the assignment of different weights to the layers in order to control their contribution to the computation of the multiplex community structure. Such layer weights could be learned based on the layer characteristics, or could be user-provided based on a-priori knowledge (e.g., user preferences).
- **P2: Determinism.** This refers to whether a method has a deterministic behavior, i.e., its output is independent from the order of examination of the nodes and/or layers.
- **P3: Auto-detection of the number of communities.** Some methods expect the number of communities to be decided ahead of time while other methods can automatically define the number of communities.
- **P4: Community structure inference.** The default is that the structure of the communities corresponds to the multiplex subgraph induced by the communities' node-sets, i.e., the internal and external links of the communities coincide with all links from the multiplex graph. Nevertheless, a method might detect communities such that the set of layers is only partially exploited to infer the community structure solution, e.g., in order to optimize the multilayer-modularity of the solution [40].

Table 3. Algorithmic properties featured by the reviewed methods. (*) indicates that the answer depends on the single-layer clustering algorithm used by the method. (-) indicates that this feature is not measurable

Algorithm	P1	P2	P3	P4
NWF	×	*	*	*
WF_N	×	*	*	*
WF_EC	×	*	*	*
WF_Diff	✓	*	*	*
CSPA	×	✓	×	✓
CCA	×	✓	✓	✓
ABACUS	×	*	✓	×
SC-ML	✓	✓	×	×
EMCD	✓	✓	×	✓
PMM	×	×	✓	×
GLouvain	✓	×	✓	×
LART	×	✓	✓	×
ML-CPM	×	✓	✓	×
InfoMap	×	✓	✓	×
MLP	×	×	✓	×
ML-LCD	✓	✓	-	×
ACLcut	×	✓	-	×

4 EXPERIMENTAL EVALUATION

We devised an experimental evaluation to pursue two main goals in comparing the various methods: one relating to the quality of the produced communities, the other to efficiency aspects. More specifically, our experiments were carried out to answer the following research questions:

Q1 To what extent do the evaluated methods produce similar community structures?
Q2 To what extent are the evaluated methods able to detect ground truth communities?
Q3 To what extent are the evaluated methods scalable?

We used three types of datasets: (i) a selection of real datasets widely used in the literature, whereby two of which are associated with available ground-truth, (ii) *benchmark datasets* generated using the mLFR framework [8], and (iii) *synthetic datasets* generated after forcing specific community structures (Figure 6). General information about the used multiplex networks including the mean and standard deviation over the layers for density, degree, average path length and clustering coefficients are reported in Table 4. More detailed information about the evaluation datasets used in the experiments are provided the Appendix.

Table 4. **Summary of structural characteristics of the evaluation networks**: number of layers (**#l**), number of actors (**#a**), number of nodes (**#n**), number of edges (**#e**), and mean/std over the layers of density (**a_den**), node degree (**a_deg**), average path length (**a_p_len**), and clustering coefficient (**ccoef**)

	Network	#l	#a	#n	#e	a_den	a_deg	a_p_len	ccoef
1	Airports	37	417	2034	3588	0.0603±0.0194	3.53±7.22	2.25±0.34	0.07±0.08
2	Aucs	5	61	224	620	0.124±0.0725	5.54±4.12	2.43±0.73	0.43±0.1
3	Dkpol	3	493	839	20226	0.068±0.0827	48.21±68.1	3.43±1.32	0.24±0.26
4	Rattus	6	2640	3263	3956	0.052±0.0697	2.42±15.35	2.75±2.22	0.03±0.08

(a) Real datasets

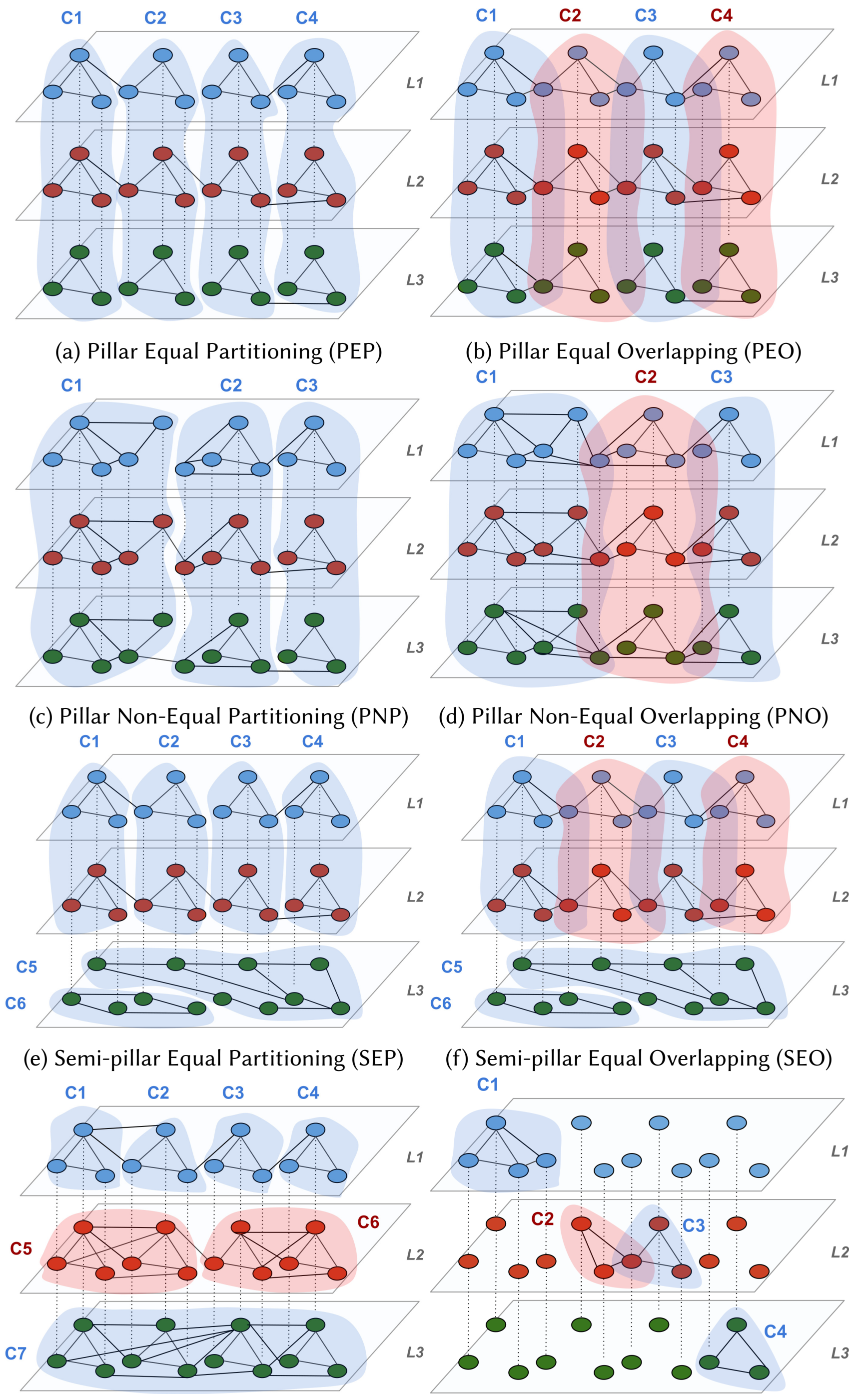
	Network	#l	#a	#n	#e	a_den	a_deg	a_p_len	ccoef
1	PEP	3	100	300	1305	0.07 ±0	7.28 ±0.1	3.33 ±0.06	0.53 ±0.01
2	PEO	3	100	300	1891	0.11 ±0	11.11 ±0.39	2.54 ±0.06	0.47 ±0.03
3	PNP	3	100	300	2051	0.12 ±0	12.31 ±0.27	2.93 ±0.05	0.62 ±0.01
4	PNO	3	100	300	2724	0.17 ±0	16.63 ±0.31	2.31 ±0.07	0.59 ±0.01
5	SEP	3	100	300	1238	0.07 ±0.01	6.93 ±0.57	3.37 ±0.26	0.53 ±0.03
6	SEO	3	100	300	1592	0.09 ±0.02	9.17 ±1.96	2.91 ±0.36	0.49 ±0.04
7	NNM	3	100	300	1108	0.07 ±0.05	6.85 ±5.35	2.42 ±0.48	0.42 ±0.36
8	NHN	3	100	300	2525	0.16 ±0.09	15.39 ±8.61	2.63 ±0.53	0.61 ±0.07

(b) Synthetic datasets with a controlled community structure

Two main stages of evaluation were devised: one for *global* methods (Sect. 5.1), whose output is a set of communities covering the whole network, and one for *local* methods (Sect. 5.2), whose output is a single community centered around a node (or set of nodes). Due to their structural differences, these two tracks had to be evaluated separately and by means of different criteria.

Selection of the methods: For *flattening* and *layer-by-layer* categories, we chose to carry out our experiments only on representative methods for each class. The reason behind this is that these approaches still try to map the problem of multiplex community discovery to either an aggregation of single-layer solutions or a community discovery on an aggregated representation of the multiplex to simple graph like representations. More specifically, we made the subsequent choices:

- Concerning *Flattening* approaches, to stay within the scope of this survey, we tried to focus mainly on the flattening as a concept rather than on the choice of the underlying single-layer



(a) Pillar Equal Partitioning (PEP)
(b) Pillar Equal Overlapping (PEO)
(c) Pillar Non-Equal Partitioning (PNP)
(d) Pillar Non-Equal Overlapping (PNO)
(e) Semi-pillar Equal Partitioning (SEP)
(f) Semi-pillar Equal Overlapping (SEO)
(g) Non-pillar Hierarchical Non-Equal (NHN)
(h) Non-Pillar Non-Equal Mixed (NNM)

Fig. 6. An illustration of 8 synthetic multiplex networks generated for different possible multiplex community structures

algorithm. Thus, we then include two flattening methods [3], namely *non-weighted flattening (NWF)*, and *edge-count weighted flattening (WF_EC)*. We used label-propagation [33] and InfoMap [11, 15] for non-weighted and weighted community detection, respectively.
- As for *layer-by-layer* approaches, we chose ABACUS [4] which is one of the earliest methods in this class; for the aggregation step, we used label-propagation as suggested by the authors.

Conversely, we preferred to focus on *multi-layer* approaches, covering the most relevant state-of-the-art methods in literature.

Parameter setting. We used the default values proposed by the original works. In some specific cases, where the parameter setting is known to have a higher bias on the results (i.e., GLouvain, ML-CPM, ABACUS, ACLcut, ML-LCD, and Infomap), we made the following choices for the input parameters:

- GLouvain: we defined two settings, $\mathsf{GLouvain}_{(hOm)}$ to denote high weight assigned to the inter-layer edges ($\omega = 1$), and $\mathsf{GLouvain}_{(lOm)}$ to refer to low value for the inter-layer edge weight ($\omega = 0.1$). The motivation is that high values for ω favor the identification of pillar communities and may prevent the identification of actor-overlapping communities that the algorithm can retrieve with a low ω.
- ML-CPM: two main parameters could bias the results and have a huge impact on the computational complexity of the algorithm, namely, the minimum number of actors that form a multi-layer clique (min-m), and the minimum number of layers to be considered when counting the multi-layer cliques (min-n). To be more inclusive, we defined two settings for these parameters, $\mathsf{ML\text{-}CPM}_{(31)}$ with (min-m=3, min-n=1) which allows single-layer communities but could be computationally very expensive with large networks, and $\mathsf{ML\text{-}CPM}_{(42)}$ with (min-m=4, min-n=2) which is less expensive computationally, but forces the communities to be expanded over at least two layers.
- ABACUS: two main parameters have effect on filtering out possible multiplex communities during the aggregation phase, namely, the minimum number of actors in a community (min-m) and the minimum number of single-layer communities in which the actors must have been grouped together (min-k). We use this algorithm with two settings, one $\mathsf{ABACUS}_{(31)}$ with (min-m=3, min-k=1) and $\mathsf{ABACUS}_{(42)}$ with (min-k=4, min-k=2) which filters out the communities that are not expanded over multiple layers.
- Infomap: this can be used to find both overlapping and non-overlapping communities. Consequently, we included it twice in our experiments, i.e., forcing a non-overlapping community discovery $\mathsf{Infomap}_{(no)}$, and accepting overlapping communities $\mathsf{Infomap}_{(o)}$.
- ACLcut: two settings were used. One with a classical random walker $\mathsf{ACLcut}_{(c)}$, and another with a relaxed random walker $\mathsf{ACLcut}_{(r)}$.
- ML-LCD: settings referred to the three different definitions to optimize the *LC* function during the selection of nodes to join a local community, namely, $\mathsf{ML\text{-}LCD}_{(lwsim)}$, for the layer-weighted similarity based *LC*, $\mathsf{ML\text{-}LCD}_{(wlsim)}$ for the within-layer similarity based *LC*, and $\mathsf{ML\text{-}LCD}_{(clsim)}$ for the cross-layer similarity based *LC*.

Assessment criteria. In order to measure pairwise similarity between two global community structures, we use the Omega-index which is a well known measure [10] that can be applied to situations where both, one, or neither of the clusterings being compared is non-disjoint (i.e., overlapping) [31]. It does so by simply averaging the number of agreements on both clusterings and then adjusting that by the expected agreement between both clusterings in case they were generated at random. An agreement is when a pair of nodes are grouped together for the same number of times j in both clusterings. The values of j starts from 0, meaning that if a pair does not appear together in either clustering, this still counts as an agreement. Given two clusterings C_1, C_2, the similarity between them using Omega index is given by

$$\text{Omega } (C_1,C_2) = \frac{\text{Observed } (C_1,C_2) - \text{Expected } (C_1,C_2)}{1 - \text{Expected } (C_1,C_2)} \tag{1}$$

$$\text{Observed } (C_1,C_2) = \frac{1}{N}\sum_{j=0}^{l} A_j \tag{2}$$

$$\text{Expected } (C_1,C_2) = \frac{1}{N^2}\sum_{j=0}^{l} N_{(j,1)} * N_{(j,2)} \tag{3}$$

Where Observed (C_1,C_2) refers to the observed agreement represented by the average number of agreements between C_1 and C_2, l is the maximum number of times a pair appears together in both C_1 and C_2 at the same time, N is the total number of possible pairs, A_j is the number of pairs that are grouped together j times in both clusterings, and $N_{(j,1)}$, $N_{(j,2)}$ is the number of pairs that have been grouped together j times in C_1, C_2 respectively. Theoretically, omega-index values are in the range [-1,1]. However, in practice, Omega-index returns 1 for two identical clusterings, and values close to 0 when one of the two input clusterings is a totally random reordering of the other one.

The reason why we choose the Omega-index for this purpose is that it is, by definition, a valid measure when one, both or none of the two clusterings is non-disjoint as we discuss in a previous study for community evaluation metrics [18]. In addition, Omega-index is an adjusted similarity measure that accounts for the by-chance agreements that might still exist between any two random clusterings over the same node-set.

For measuring similarity between two local communities s_1, s_2, we use Jaccard coefficient as:

$$JC = \frac{N(s_1, s_2)}{N(s_1) + N(s_2) - N(s_1, s_2)} \tag{4}$$

where $N(s_1)$ refers to the number of actors in solution s_1 and $N(s_1, s_2)$ refers to the number of common actors between two solutions s_1, s_2. The values of Jaccard coefficient lie in the range [0,1] where 1 means perfect similarity and 0 means perfect dissimilarity.

In order to measure the accuracy of the solutions obtained by global methods with respect to a ground truth (Section 5.1.3), we resort again to Omega-index. Note that for the real-world networks we used, the provided ground-truth is a complete node-partitioning, i.e., every node in the multiplex is assigned to only one community. Since this would introduce a bias in the accuracy obtained by some methods (i.e., overlapping methods and methods which allow nodes with no community memberships), we computed two accuracy scores for each method: *full* accuracy (**F**), which refers to the similarity between the original community structure produced by a method and a ground truth, and *intersection* accuracy (**I**), in which all nodes that are not assigned to at least one community have been removed from both the community structure and the ground truth.

The accuracy of local community detection methods (Section 5.2.2)) has been evaluated by comparing pairwise similarities (using Jaccard-index) between a given actor (i.e., seed node) and the ground truth community it belongs to. The average Jaccard-index over all actors is then used as the final accuracy score.

5 RESULTS

In this section we will present the experimental results of our comparative evaluation. Results of the comparative evaluation of global methods are reported in Section 5.1, while results related to the evaluation of local methods are reported in Section 5.2.

5.1 Global Methods

In this section we will report the experimental results of the comparative evaluation of global multiplex community detection methods. The section is structured as follows: Section 5.1.1 reports on the main structural properties of the community structures detected by the evaluated methods in different datasets. Section 5.1.2 discusses the results of the pairwise comparison between different methods. Section 5.1.3 presents the results of the accuracy analysis, while finally Section 5.1.4 focuses on scalability.

5.1.1 Basic descriptive statistics. As the first step of our comparative analysis, we analyzed the structural properties of the different community structures identified by the evaluated methods. Table 5 reports on the statistics concerning the community structures obtained on the smallest (Aucs) and largest (Airports) of the real-world multiplex networks taken into account; statistics for the other real-world networks are reported in the Appendix. In Table 5, we denote with $\#c$ the number of communities, with $sc1$ the size of the largest community, with $sc2/sc1$ the ratio between the size of the largest community to the second largest, with $\%n$ the percentage of nodes assigned to at least one community, with $\%p$ the percentage of pillars, with $\%ao$ the percentage of actors in more than one community, with $\%no$ the percentage of nodes in more than one community and with $\%s$ the percentage of singleton communities.

These statistics should be observed bearing in mind that some of them are not significant in all cases, i.e., depending on the properties of each algorithm reported in Table 3. For instance, flattening methods will always return pillar communities, while some algorithms are forced to assign each node to at least one community. Properties also act as an indirect bias, e.g., the number of communities detected by methods allowing *overlapping* solutions is likely to be higher than that of algorithms producing disjoint communities (in terms of nodes and/or actors).

While we show results averaged over 10 runs, in order not to clutter the results visualization, we do not show standard deviation. We do this since it does not apply to deterministic methods (i.e., it is always zero). Moreover, even for the non-determistic ones, it does not bring any relevant information to the discussion.

It can be observed how LART generates a number of communities which is higher than that of most other methods on all real networks. However a large percentage of these communities appear to be singletons, indicating that this algorithm mostly fails in aggregating nodes into communities. Other algorithms which appear to generate a relatively high number of communities regardless of the network structure are MLP and $\text{ABACUS}_{(31)}$. Interestingly, while MLP still identifies a significant number of singletons, the number of singletons for $\text{ABACUS}_{(31)}$ is always zero. As regards to the size of the largest community, higher values correspond to WF_EC and Infomap (both variants) on all networks, while LART, ABACUS and ML-CPM (except on Arxiv) tend to show smaller values. Concerning $sc2/sc1$, we can observe how the values tend to be all relatively high for the smallest (Aucs) and largest (Airports) network, indicating that in these cases the larger communities for each identified community structure have comparable sizes. Interestingly, strong imbalances can arise in the community structures identified for medium-sized networks: values both close to zero and close to one can be observed for Airports, Dkpol and Rattus, however no particular trends can be identified in the different methods' behavior (except for a tendency towards higher values for ABACUS and ML-CPM).

The percentages of $\%n$, $\%p$, $\%ao$ and $\%no$ mostly depend on the properties of each method.

- With regards to the percentage $\%n$ of nodes assigned to at least one community, as we discussed in Section 3, certain methods[1] are forced to provide a community assignment for

[1] NWF, WF_ECGLouvain (both variants), LART, Infomap (both variants)

each node: in these cases the value of $\%n$ will be always 1.00. As regards the other methods, we can observe how ML-CMP$_{(42)}$ and ABACUS$_{(42)}$ are unable to detect community assignments for a majority of nodes on almost all networks: same observation applies to MLP on smaller networks (i.e., Airports and Aucs).
- The percentage $\%p$ of pillars, both flattening methods always return pillar communities (that is trivial, since the information about layers is lost during the flattening process). Both variants of Infomap also always return pillar communities, and, in most cases, both variants of GLouvain (with some exception for Dkpol and Aucs).
- The percentage of actors ($\%ao$) and nodes ($\%no$) mainly depends on the properties of the specific methods whether they allow overlapping (on the node level or the actor level) or not.
- The percentage of singleton communities $\%s$ appears to be extremely high in the case of LART, as discussed earlier, and significant (but reasonable) in the cases of NWF and MLP.

5.1.2 Pairwise comparison analysis. In order to answer **Q1** (i.e., "To what extent do the evaluated methods produce similar community structures?", cf. Section 4), we set up an evaluation stage based on pairwise comparison between the selected methods, i.e., in order to determine the similarity between the community structures produced by each couple of methods on each network.

We structured the analysis on different *sub-stages*, depending on the nature of the method and of the network. More specifically:[2]

- Figures 7–8 report on the results of pairwise analysis among node-partitioning methods on real-world and synthetic networks, respectively;
- Figures 9– 10 report on the results of pairwise analysis among node-overlapping methods on real-world and synthetic networks, respectively.

All results are organized as heatmaps reporting the Omega-index values for the pairwise similarities.[3] We show Omega-index values in the main paper for a matter of homogeneity, i.e., since NMI cannot be applied to overlapping solutions. We provide NMI values for the node-partitioning methods on real-world and synthetic networks in the Appendix.

Figure 7 reports results regarding node-partitioning methods using real-world networks. On the smallest network (Aucs), we can observe significant similarities between the community structures identified by Infomap$_{(no)}$ and the two variants of GLouvain (with higher values for GLouvain$_{(hOm)}$); some significant similarities also arise between Infomap$_{(no)}$ and MLP. As for Airports, significant similarities can only be identified between the two variants of GLouvain - possibly indicating that a high number of layers can emphasize differences between the methods. On Dkpol, we observe a perfect correspondence between Infomap$_{(no)}$ and WF_EC, that can be explained with the fact that Infomap$_{(no)}$ is the algorithm used to detect communities on the flattened network (cf. Section 4). Similarity between the two variants of GLouvain is confirmed also in this case. A different behavior can be observed on Rattus, where major similarities arise among the two variants of GLouvain, NWF and MLP.

A completely different scenario can be observed for synthetic networks (Figure 8), which are produced with controlled community structures. Similarities are generally much higher than the ones observed for real-world networks, indicating that *artificial* communities are less noisy and more recognizable than real ones, as will be confirmed by the accuracy analysis in Section 5.1.3, resulting in similar outputs from different methods.

[2] The executions were stopped if not terminated within 24 hours. These cases are left blank in the reported heatmaps.
[3] We do not show results for the Arxiv network, since computation of Omega-index was too computationally expensive in this case.

Table 5. Statistics about the community structures obtained on the Aucs (upper table) and Airports (lower table) networks (results averaged over 10 runs). We denote with **#c** the number of communities, with **sc1** the size of the largest community, with **sc2/sc1** the ratio between the size of the largest community to the second largest, with **%n** the percentage of nodes assigned to at least one community, with **%p** the percentage of pillars, with **%ao** the percentage of actors in more than one community, with %no the percentage of nodes in more than one community and with **%s** the percentage of singleton communities

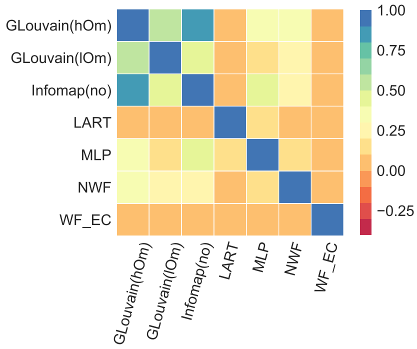
method	$\#c$	$sc1$	$sc2/sc1$	$\%n$	$\%p$	$\%ao$	$\%no$	$\%s$
NWF	10.60	94.20	0.54	1.00	1.00	0.00	0.00	0.42
WF_EC	6.00	53.00	0.96	1.00	1.00	0.00	0.00	0.00
MLP	19.60	27.80	0.63	0.50	0.16	0.00	0.00	0.41
GLouvain$_{(hOm)}$	5.00	60.80	0.87	1.00	0.99	0.00	0.00	0.00
GLouvain$_{(lOm)}$	5.09	59.60	0.89	1.00	0.53	0.29	0.00	0.00
LART	45.00	33.00	0.66	1.00	1.00	0.00	0.00	0.91
Infomap$_{(no)}$	6.00	59.00	0.89	1.00	1.00	0.00	0.00	0.00
Infomap$_{(o)}$	21.00	69.00	0.86	1.00	1.00	0.70	0.74	0.00
ML-CPM$_{(31)}$	41.00	59.00	0.61	0.83	0.03	0.95	0.76	0.00
ML-CPM$_{(42)}$	11.00	18.00	0.88	0.37	0.15	0.67	0.40	0.00
ABACUS$_{(31)}$	50.60	20.70	0.87	0.79	0.03	0.93	0.65	0.00
ABACUS$_{(42)}$	16.10	18.89	0.88	0.44	0.07	0.58	0.50	0.00

Aucs

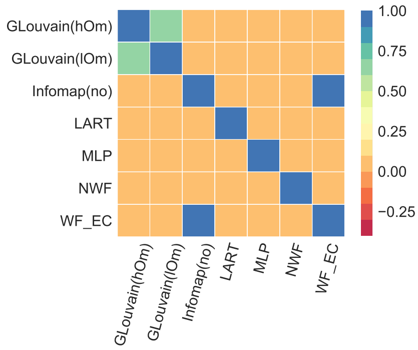
method	#c	sc1	sc2/sc1	%n	%p	%ao	%no	%s
NWF	75.40	1482.20	0.07	1.00	1.00	0.00	0.00	0.47
WF_EC	10.00	1846.00	0.04	1.00	1.00	0.00	0.00	0.00
MLP	88.40	30.60	0.72	0.18	0.41	0.00	0.00	0.29
GLouvain$_{(hOm)}$	9.50	1052.40	0.43	1.00	1.00	0.00	0.00	0.00
GLouvain$_{(lOm)}$	9.69	1027.90	0.42	1.00	0.99	0.00	0.00	0.00
LART	1822.00	213.00	0.00	1.00	0.09	0.59	0.00	0.99
Infomap$_{(no)}$	10.00	1855.00	0.03	1.00	1.00	0.00	0.00	0.00
Infomap$_{(o)}$	35.00	1471.00	0.77	1.00	1.00	0.63	0.92	0.00
ML-CPM$_{(31)}$	62.00	93.00	0.72	0.44	0.08	0.53	0.13	0.00
ML-CPM$_{(42)}$	3.00	8.00	1.00	0.01	0.00	0.00	0.00	0.00
ABACUS$_{(31)}$	20250.80	92.10	0.95	0.88	0.00	0.63	0.78	0.00
ABACUS$_{(42)}$	11955.50	89.30	0.95	0.66	0.00	0.69	0.84	0.00

Airports

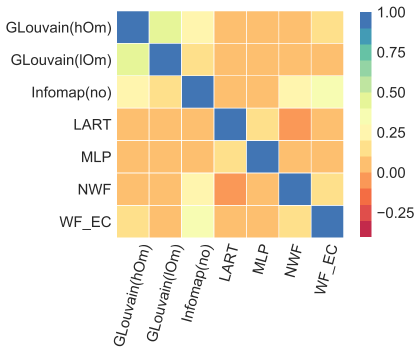
Generally high similarities (i.e., Omega-index above 0.75) can be observed for PEP, PNP and NHN multiplex networks. The variants of GLouvain, Infomap$_{(no)}$ and Lart produce identical solutions for PEP and identical or nearly identical for PNP. On NHN, Infomap$_{(no)}$, LART and MLP produce identical communities, while similarities of these methods w.r.t. GLouvain are lower but still significant. The situation is much different when taking into account semi-pillar community structures (SEP), where the only identity is between Infomap$_{(no)}$ and GLouvain$_{(hOm)}$. Some significant similarities can be observed between MLP, WF_EC and the solution produced by Infomap$_{(no)}$ and GLouvain$_{(hOm)}$, while other values are less significant.



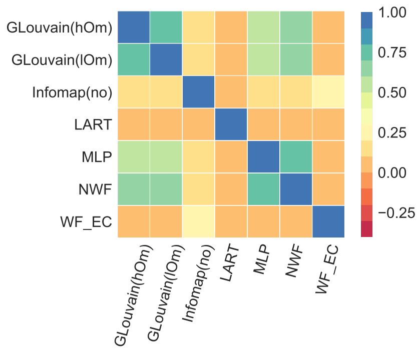
(a) Aucs



(b) Dkpol



(c) Airports



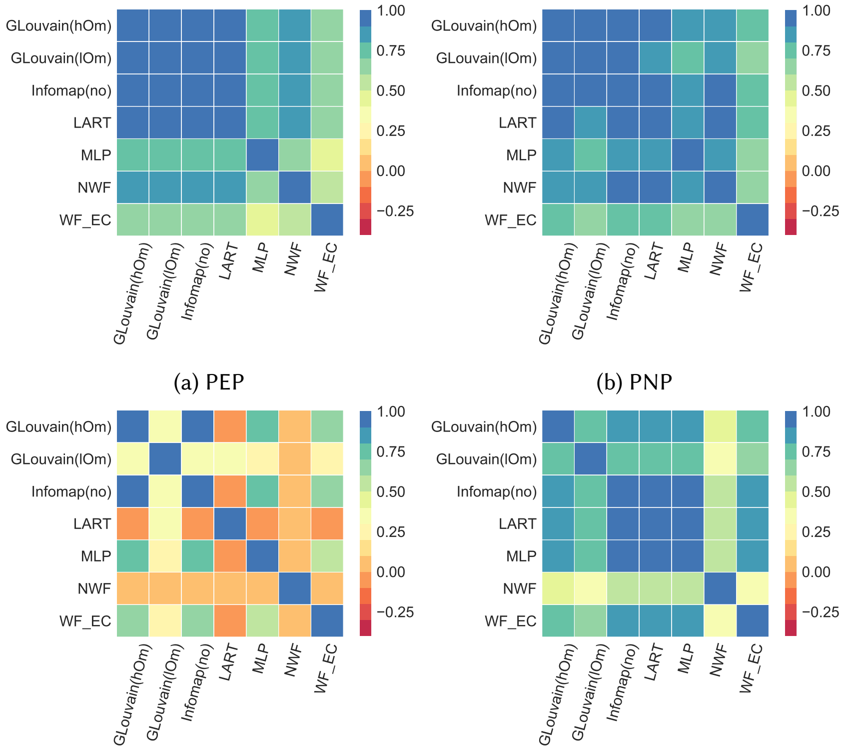
(d) Rattus

Fig. 7. Pairwise comparison among node-partitioning community detection methods using Omega-index, on real-world networks

With regards to node-overlapping methods, pairwise similarities were relatively low on both real-world networks (Figure 9) and synthetic ones (Figure 10). The only significant similarities can be observed on synthetic networks, more specifically between ML-CPM$_{(42)}$ and ABACUS$_{(42)}$ (on SEO and NNM) and between Infomap$_{(o)}$ and ABACUS$_{(42)}$ on PNO.

Summing up, we observed that node-partitioning methods may produce similar community structures on specific cases (i.e., depending on the methods and the target network), suggesting that, when multiple community memberships are not allowed, some communities will often be unambiguously recognized in the network topology. Conversely, multiple community memberships allowed by overlapping methods end up in extremely variate solutions, i.e., relatively low similarities are observed regardless of the selected network and pair of methods.

5.1.3 Accuracy analysis. With the aim of answering **Q2** (i.e., "To what extent are the evaluated methods able to detect ground truth communities?", cf. Section 4), we perform here an extensive quantitative analysis about the accuracy obtained by each method with respect to ground truth communities. For real-world networks, only two of them have an available ground truth - specifically Aucs (i.e., affiliations to research groups) and Dkpol (i.e., affiliation to political parties). All synthetic networks naturally come with controlled ground truth communities. Please recall that we compute *full* (**F**) and *intersection* (**I**) accuracy in order to take into account the bias coming from different



(a) PEP (b) PNP

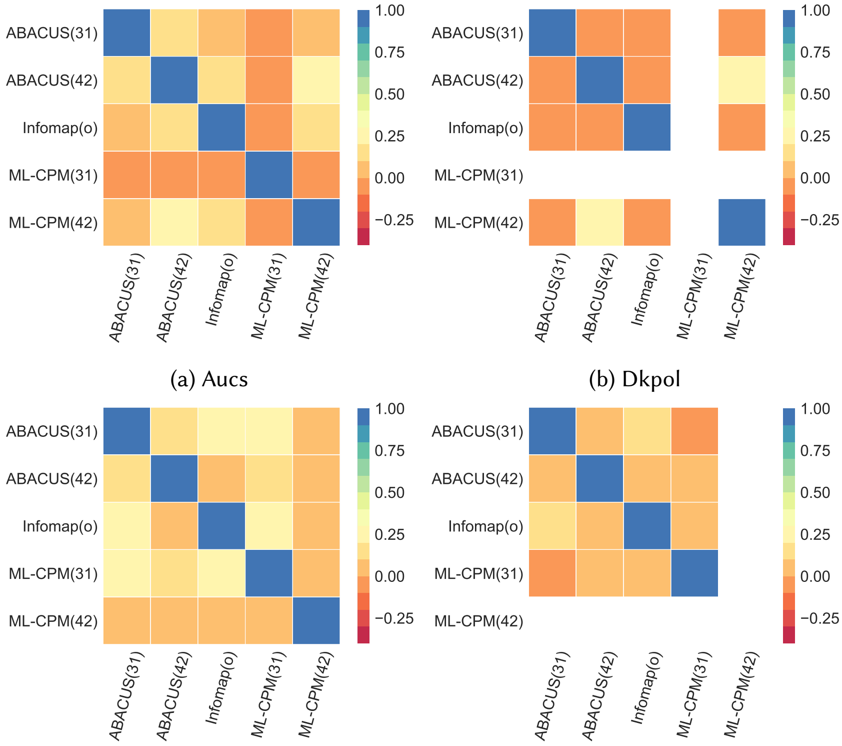
(c) SEP (d) NHN

Fig. 8. Pairwise comparison among node-partitioning community detection methods using Omega-index, on synthetic networks with node-partitioning communities

constraints in the ground truth and the obtained community structure (i.e., overlapping and unassigned nodes, cf. Section 4).

Figure 11 reports on the accuracy obtained by the evaluated methods on real-world networks. It can be observed how accuracy values are relatively low on both networks for all methods, i.e., with Omega-index always below 0.8 and often below 0.5. On Aucs, the best performing method is Infomap$_{(no)}$ (0.61), followed by GLouvain$_{(hOm)}$ (0.56), while noting how some methods reach a near zero accuracy (i.e., WF_EC, ML-CPM$_{(31)}$). It is interesting to observe how, even on a small network like Aucs, the average accuracy is extremely low (0.29), indicating that the size of the network is not necessarily a factor that influences the methods' performance in terms of accuracy. The results are even more variable on Dkpol, where most methods show zero [4] or near zero values. An exception to this are the two variants of GLouvain, both of which reach accuracies of 0.68 (GLouvain$_{(hOm)}$) and 0.53 (GLouvain$_{(lOm)}$) respectively. Moreover, the relatively high values obtained by ML-CPM$_{(42)}$ (0.74) and ABACUS$_{(42)}$ (0.48) for the **I** accuracy largely depends on the fact that they are calculated on an extremely small sample of the nodeset, i.e., most nodes do not have a community assigned, and **F** accuracies are near zero in both cases.

[4]Zero values are a result of either identifying a clustering constituted of only one giant component (i.e with NW_EC and Infomap$_{(no)}$) or the long execution time that exeeded 24 hours (i.e with ML-CPM$_{(31)}$)



(a) Aucs (b) Dkpol

(c) Airports (d) Rattus

Fig. 9. Pairwise comparison among node-overlapping community detection methods using Omega-index, on real-world networks

Looking at the results of the node-partitioning methods on the synthetic networks (Figure 12), it can be noted how accuracies are high (i.e., 1 or close to 1) for most methods on networks with pillar community structures (PEP and PNP) - with Infomap, LART and GLouvain performing slightly better than other methods. A drop in accuracy values can be observed in networks with semi-pillar (SEP) and non-pillar (NHN) community structures, with maximum values around 0.7. However, Infomap, LART and GLouvain are still the best performing methods, followed by MLP (while both flattening approaches show lower accuracies).

Different behaviors can be observed when we take into account the node-overlapping methods on synthetic networks (Figure 13). Accuracy values are generally much lower than in the node-partitioning case, on all four networks. It is interesting to note how Infomap is the best performing method on networks with pillar community structure (0.68 on PEO and 0.86 on PNO), while it shows near zero values for the other two cases. On the networks with semi-pillar (SEO) and non-pillar (NNM) community structures, $\text{ABACUS}_{(31)}$ seems to perform generally better than the other methods.

Summarizing the conclusions drawn from the analysis on synthetic networks, we observed that detecting semi-pillar and non-pillar communities appears to be extremely difficult for all methods, both on node-partitioning and node-overlapping cases. These results may indicate that, even though multiplex methods are generally able to profitably exploit the information coming from different

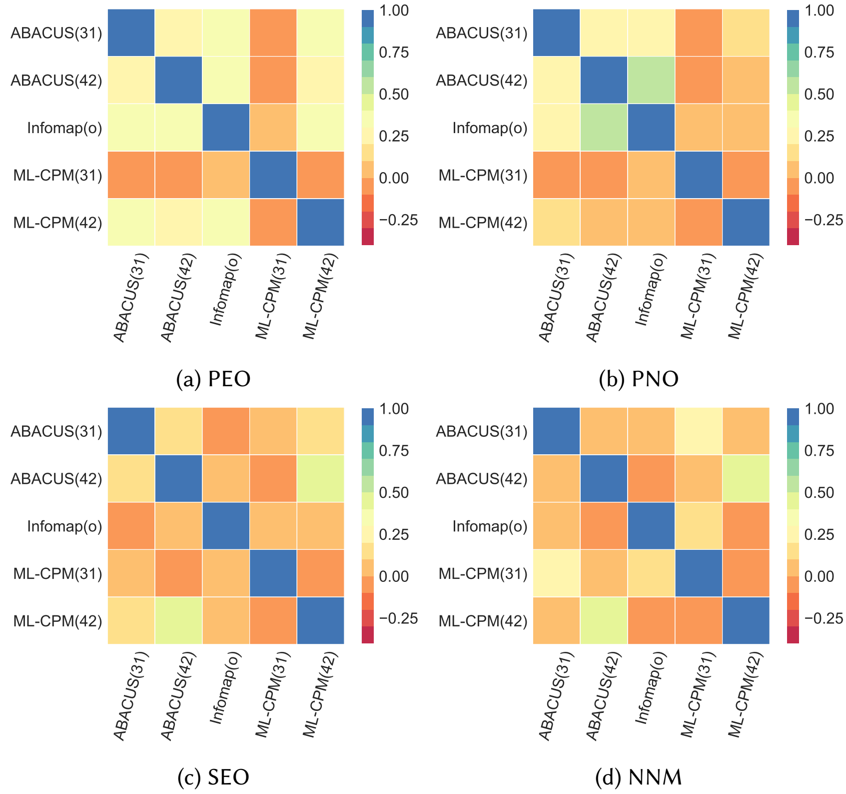
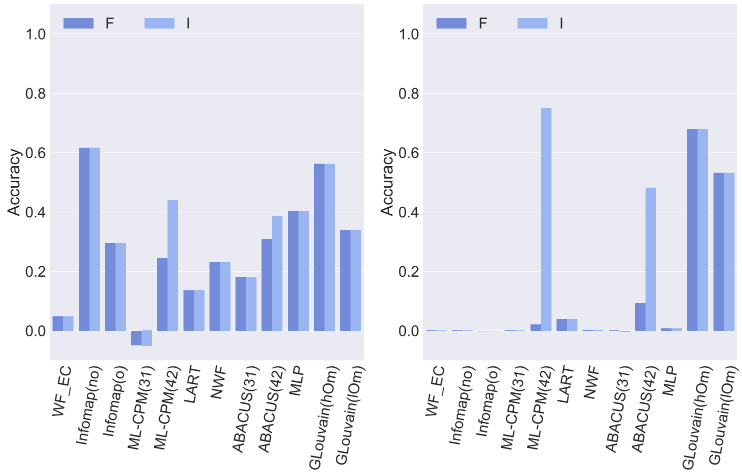


Fig. 10. Pairwise comparison among node-overlapping community detection methods using Omega-index, on synthetic networks with Node-Overlapping communities

layers (e.g., performing better than flattening methods), they are still not able to fully cope with "non-standard" multiplex community structures.

As a final remark, the difference in performance between real-world and synthetic networks confirms how the "ideal" concept of community, i.e., the one based on topological density that is used to build the synthetic ones and to drive the detection process of the methods, is often far from the ground truth communities observed in real cases (which are, in turn, often questionable and subjective). This is a well known problem in the community detection field, and poses challenges in both ways, i.e., concerning the need to design both more powerful methods and more reliable ground truths.

5.1.4 Scalability Analysis. In order to answer **Q3** ("To what extent are the evaluated methods scalable?", cf. Section 4), we tested the scalability of the selected methods with respect to number of actors and number of layers. We generated 2 groups of synthetic multiplex networks using mLFR benchmark [8]. More details about these datasets are provided in the Appendix. Figures 14–15 report the scalability of each method with respect to an increment in the number of actors and the number of layers respectively. Note that in both cases the scalability of the flattening algorithms largely depends on the one of the community detection method used at the final step, since the computational cost of the flattening process is irrelevant. Therefore, we chose not to include flattening methods in our scalability analysis. We chose also not to include different variations of



(a) aucs

(b) dkpol

Fig. 11. Accuracy with respect to a ground-truth for real-world networks, measured using omega-index.

the same algorithm when they show similar trends in their scalability (for example Infomap$_{(o)}$ and Infomap$_{(no)}$ are both referred to as Infomap).

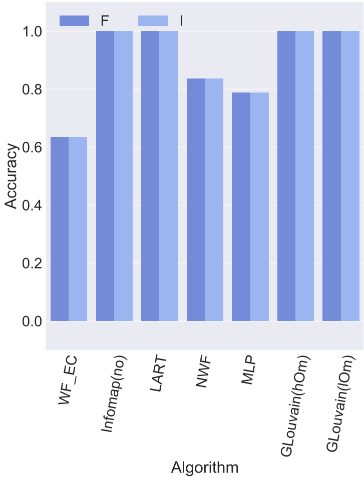
For scalability in the number of actors (Figure 14), we performed tests by varying the number of layers $l = 2, 5, 10$. Some methods proved to be extremely scalable, more specifically, ABACUS, Infomap and MLP - all of which could run in a few seconds on networks containing up to 10000 actors for all values of l. Similar results were obtained by GLouvain for $l = 2$ and $l = 5$, while running time increased for $l = 10$ and a number of actors higher than 5000. ML-CMP and LART proved to be much less scalable, with a running time strongly dependent on the number of layers. Moreover, in several cases they were not able to terminate in anreasonable time (i.e., less then an hour) when increasing the number of actors. The only exception is ML-CMP$_{(42)}$ with $l = 2$, which maintains reasonable running times up to 10000 actors.

As regards to the scalability in the number of layers (Figure 15), Infomap, GLouvain and MLP proved to be extremely scalable, showing running times of a few seconds on networks containing up to 50 layers. The same thing does not hold for ABACUS, whose scalability is good only up to 20 layers. As for the previous case, ML-CPM and LART confirmed to be the worse algorithms in terms of scalability.

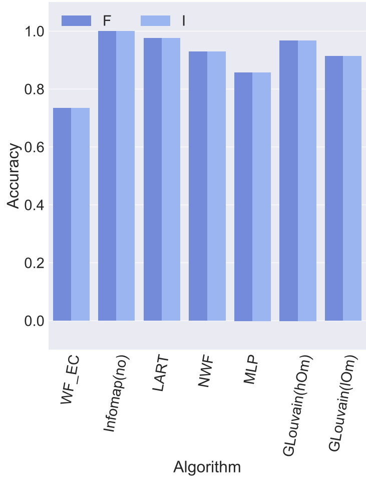
5.2 Local Methods

In this section we will report the experimental results of the comparative evaluation of local multiplex community detection methods. The section is structured as follows: Section 5.2.1 reports on the results of the pairwise comparison between different methods, Section 5.2.2 presents the results of the accuracy analysis, while Section 5.2.3 discusses scalability issues.

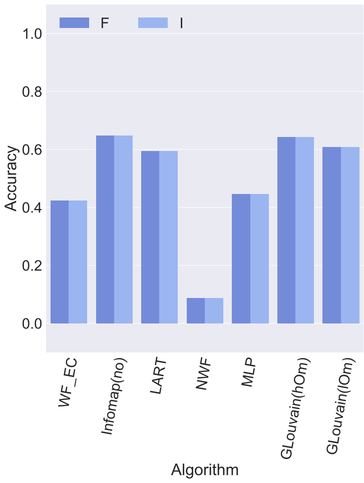
5.2.1 Pairwise comparison. As seen in Section 5.1.2 for global methods, we set up an equivalent evaluation stage based on pairwise comparison between the local methods. In this case, we resorted to Jaccard-index to measure the similarity of the community solutions produced by two local methods. Since these methods are query-dependent (i.e., they return the local community of a given query/seed node), we computed the Jaccard similarity between each pair of communities



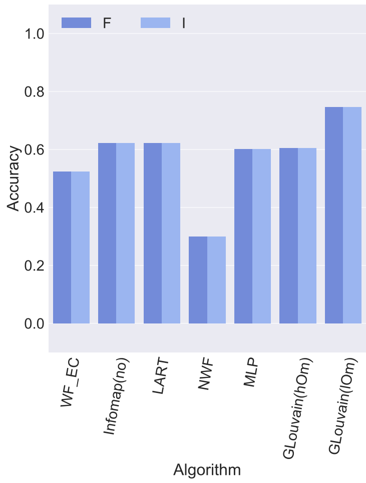
(a) Pillar Equal Partitioning (PEP)



(b) Pillar Non-equal Partitioning (PNP)



(c) Semi-pillar Equal Partitioning (SEP)

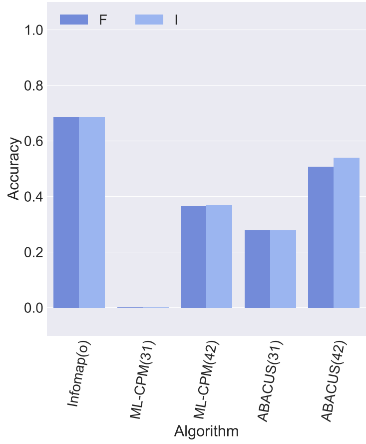


(d) Non-pillar Hierarchical Non-equal (NHN)

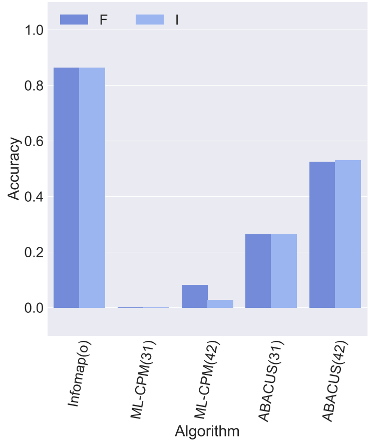
Fig. 12. Accuracy of node-partitioning methods with respect to a ground-truth, for synthetic networks with node-partitioning community structure, measured using omega-index.

obtained using the same actor as seed, and then averaged the results over all actors. The standard deviation of these average values is provided in the Appendix.

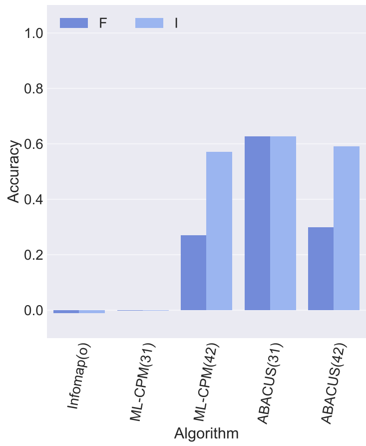
Figure 16 reports on the results obtained on real-world networks. On most of these networks (i.e. Dkpol, Airports, and Rattus), we can note that communities identified by different variants of ML-LCD and ACLcut tend to be very different. However, by looking at Aucs, the communities identified by all variants of both ML-LCD and ACLcut tend to be less different and a higher similarity can be observed among the three variants of ML-LCD.



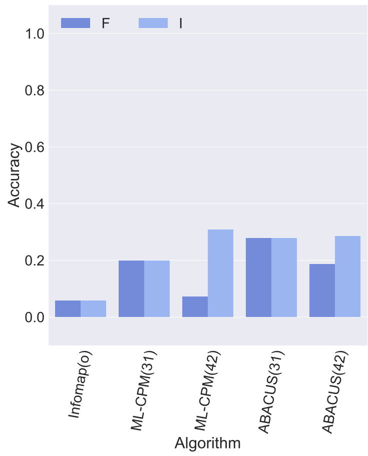
(a) Pillar Equal Overlapping (PEO)



(b) Pillar Non-equal Overlapping (PNO)



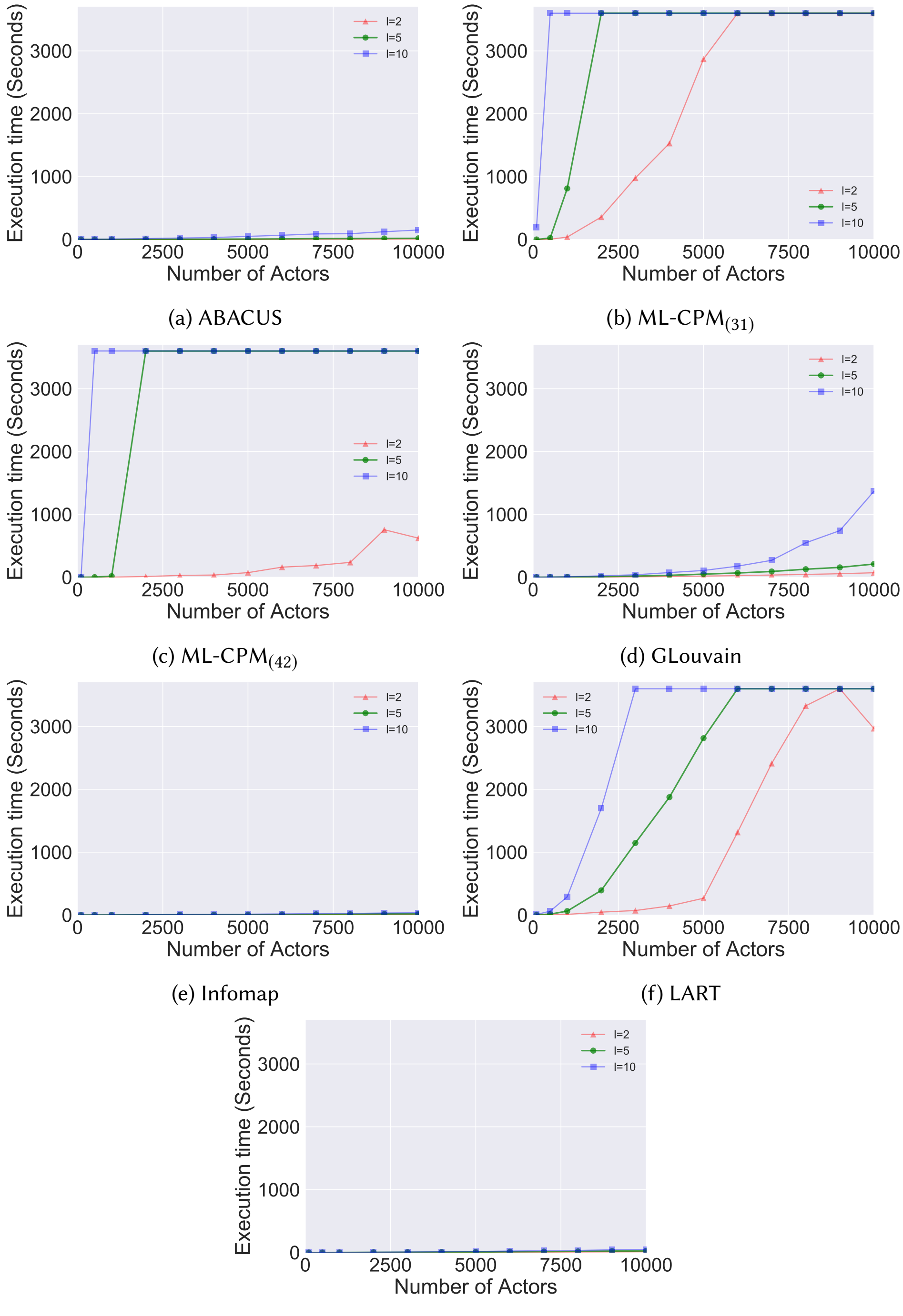
(c) Semi-pillar Equal Overlapping (SEO)



(d) Non-pillar Non-equal Mixes (NNM)

Fig. 13. Accuracy of node-overlapping methods with respect to a ground-truth, for synthetic networks with node-overlapping community structure, measured using omega-index.

For synthetic networks (Figure 17), it can be noted how similarities are higher for networks based on pillar community structures. In some cases (i.e., PEP and PNP) all methods are practically interchangeable, with all similarities equal or near to 1.0. In other networks with pillar (i.e., PEO and PNO), semi-pillar (i.e., SEP and SEO) or both (NNM and NHN) community structures, similarities are stronger between the different variants of each method. Summing up, we observed some similarities in the behavior of all local methods on some real-world and synthetic networks, with an expected tendency of the variants of a same method to identify similar local communities. Nevertheless, this cannot be taken as a general rule, since we also observed specific cases where all methods behaved differently from each other, both on real-world and synthetic networks.



(a) ABACUS

(b) ML-CPM$_{(31)}$

(c) ML-CPM$_{(42)}$

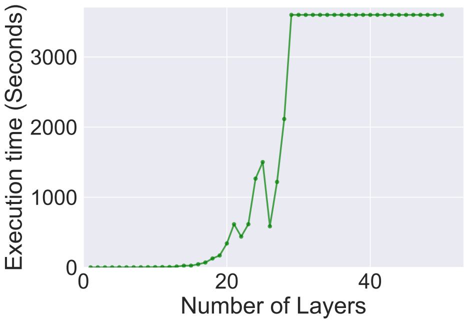
(d) GLouvain

(e) Infomap

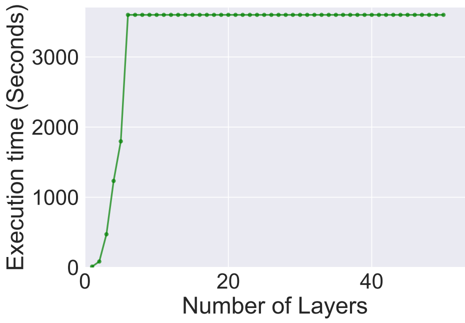
(f) LART

(g) MLP

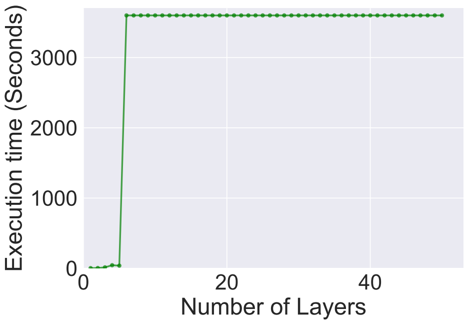
Fig. 14. **Group I:** Scalability of different community detection methods with respect to the number of actors



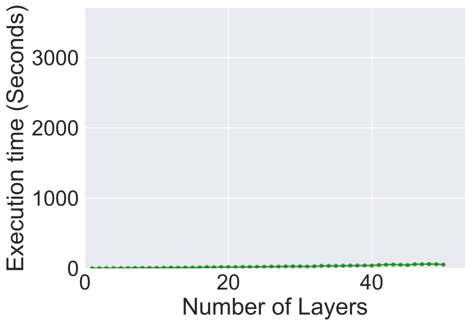
(a) ABACUS



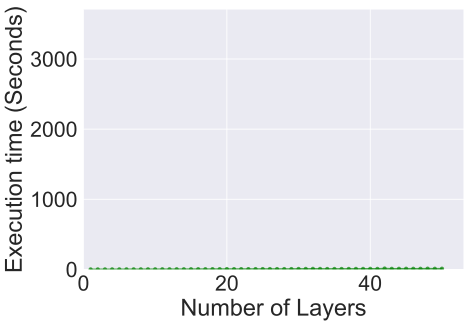
(b) ML-CPM$_{(31)}$



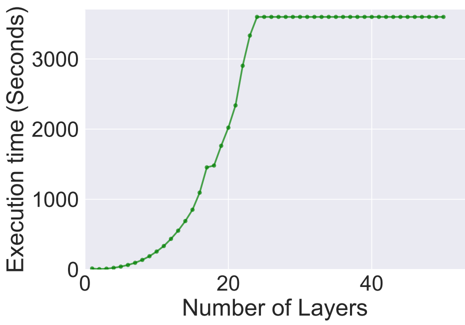
(c) ML-CPM$_{(42)}$



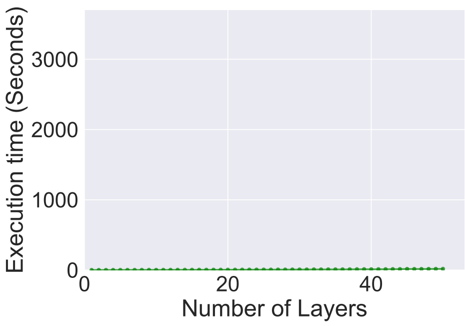
(d) GLouvain



(e) Infomap

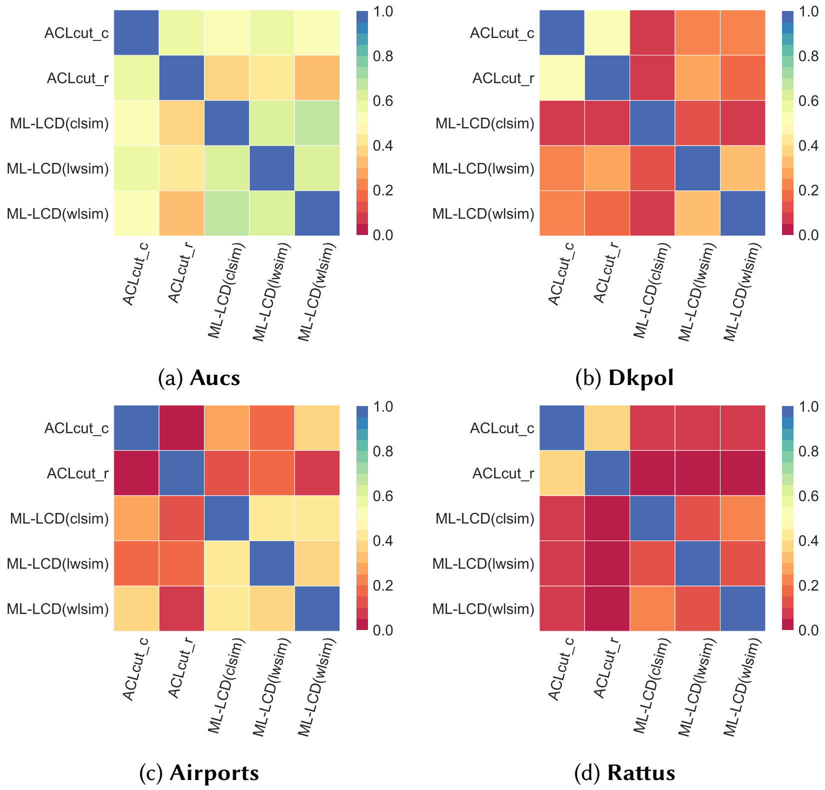


(f) LART



(g) MLP

Fig. 15. **Group II:** Scalability of different community detection methods with respect to the number of layers



(a) **Aucs** (b) **Dkpol**

(c) **Airports** (d) **Rattus**

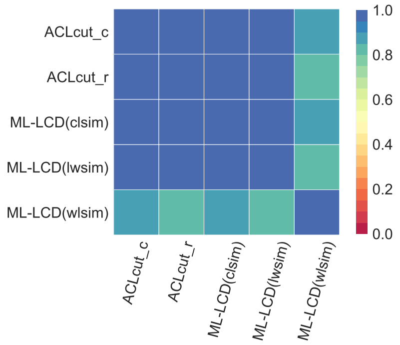
Fig. 16. Average pairwise similarity among the different local methods when the same seed is used as an input, on real-world networks

5.2.2 Accuracy analysis. We performed an accuracy analysis on the local community detection methods, by comparing the local community of each actor to the one that same actor belongs to in the ground truth. As per the previous section, pairwise similarity is computed using Jaccard, while the final accuracy value is the average over all actors.

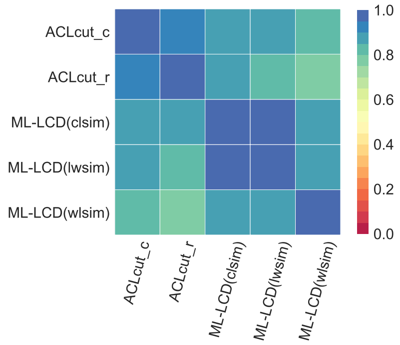
Figure 18 shows results on real world networks. On Aucs, accuracy is in the range of 0.5–0.7 for 4 out f 5 methods, with ML-LCD(wlsim) being the best performer (0.7). Much lower accuracy values were obtained on Dkpol, where the best performing method was ML-LC(lwsim) (0.27).

Concerning synthetic networks (Figure 19), the analysis is limited to networks with pillar actor partitioning community structure (i.e., PEP and PNP), for compatibility with the methods' output (i.e, both return actor communities). It is evident how accuracy values are much higher than the ones observed for real-world networks, with all values in the range of 0.8–1.0. ML-LCD(clsim) is the best performing method, since it is able to perfectly identify the ground truth community structure on both networks.

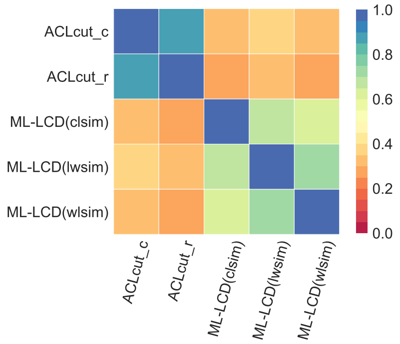
Summarizing, while all methods proved to be able to identify synthetic pillar community structures, their performance was much worse on real world networks. These results confirm the behavior observed for global methods (cf. Section 5.1.3). Moreover, it should be pointed out that comparing a global community structure (i.e., the ground truth) to a set of local ones (i.e., the results obtained by local methods on all actors) may not be completely fair, but unfortunately no networks with an associated ground truth of multiplex local communities are available at the time of writing.



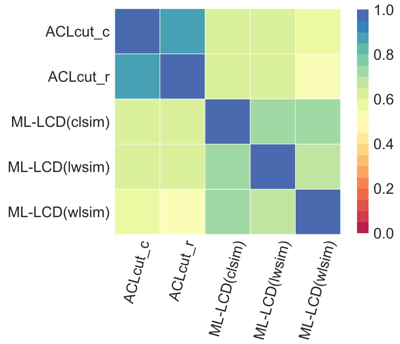
(a) **PEP**



(b) **PNP**



(c) **PEO**



(d) **PNO**

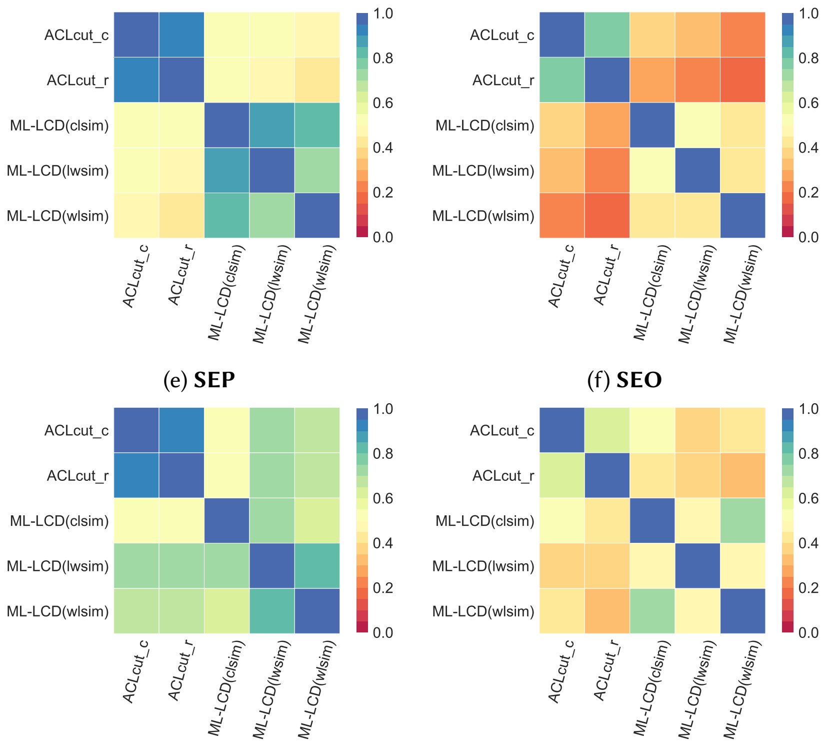
Fig. 17. Average pairwise similarity among the different local methods when the same seed is used as an input, on synthetic networks

5.2.3 *Scalability Analysis.* We tested the scalability of local community detection methods in terms of number of actors and number of layers. To carry out the experiment we used the synthetic networks already used for the global case (cf. Section 5.1.4). For each network, we present average execution times obtained on 100 random seeds. For each method, we choose the least scalable variant as a representative of that method's scalability.

Figures 20–21 show results related to scalability in terms of number of actors and of layers respectively. In both cases, ACLcut showed good scalability up to 10000 actors and 50 layers, with fairly linear trends. On the other hand, ML-LCD was scalable just up to 8000 actors for networks with 10–layers network. Moreover, its scalability trend with respect to the number of layers seems to be extremely variable.

6 DISCUSSION

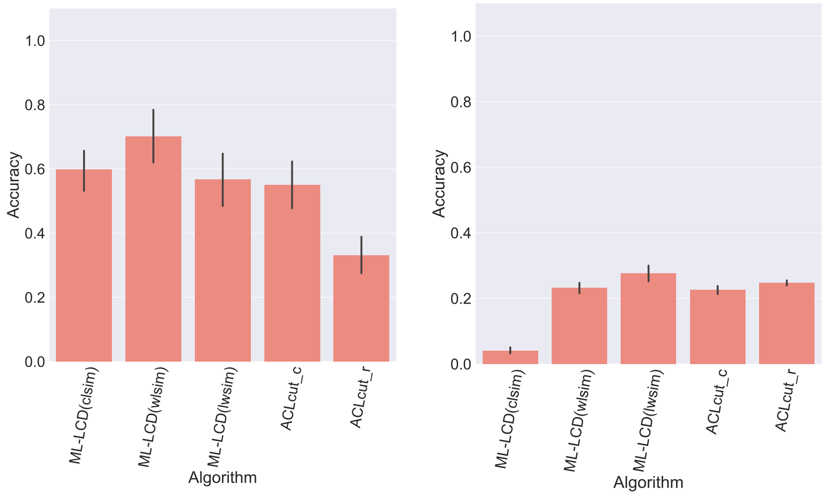
As mentioned in Section 4, the main goal of our experimental study on a high level was to assess the accuracy, the scalability with respect to the multiplex network size, and the pairwise similarity of multiplex community detection methods. This allowed us to provide a few guidelines on the relatively most appropriate method for the data and the task at hand. At the same time, observing in which cases the selected methods have consistently failed in identifying the communities accurately allowed us to identify the multiplex community structures that are challenging with the currently



(e) **SEP** (f) **SEO**

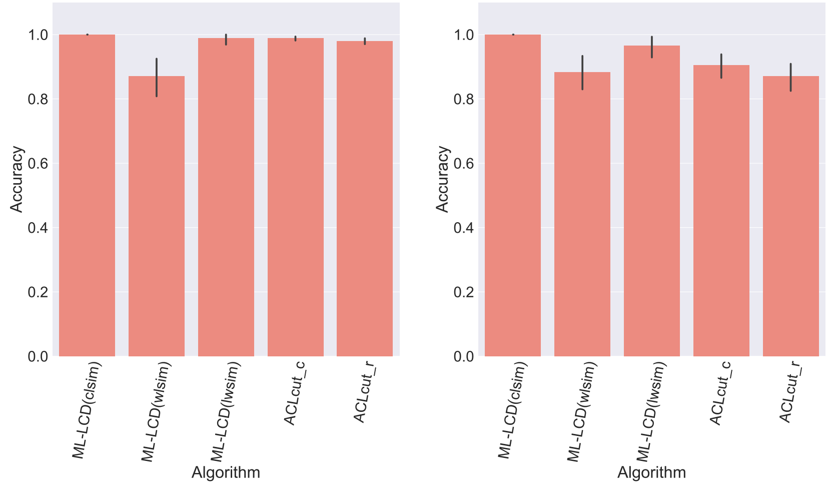
(g) **NHN** (h) **NNM**

Fig. 17. Average pairwise similarity among the different local methods when the same seed is used as an input, on synthetic networks



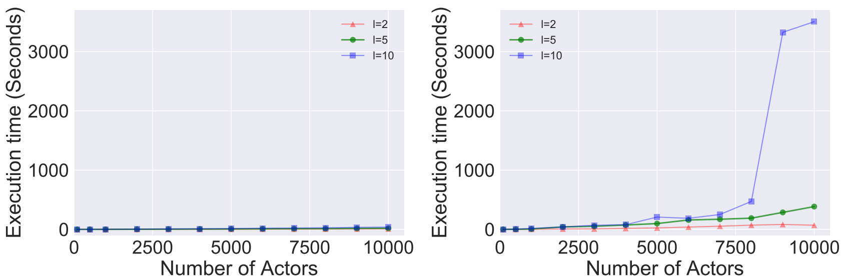
(a) **Aucs** (b) **Dkpol**

Fig. 18. Average accuracy of the local methods with respect to a ground-truth, on real-world networks



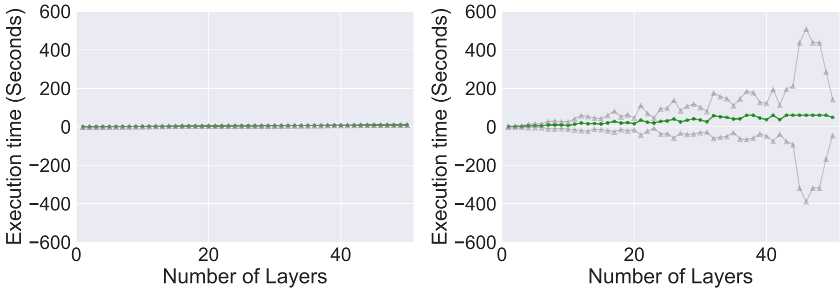
(a) **PEP** (b) **PNP**

Fig. 19. Average accuracy of the local methods with respect to a ground-truth, on synthetic networks with actor-partitioning community structures



(a) **ACLcut** (b) **ML-LCD**

Fig. 20. Average scalability of local methods with respect to the number of actors in the multiplex network for three variants of the number of layers (l=2, l=5, and l=10)



(a) **ACLcut** (b) **ML-LCD**

Fig. 21. Average scalability of local methods with respect to the number of layers in the multiplex network (standard deviation provided)

available community detection algorithms. This proposes a set of open problems for community detection methods in multiplex networks.

Accuracy analysis on synthetic multiplex networks, which were generated after forcing a specific community structure, has revealed that the various different methods can achieve relatively high performance when the community structure in the input multiplex is a pillar structure. In this type of multiplex networks, Infomap appears to outperform other global methods by discovering community structures that are closer to the ground-truth, whereas ML-LCD(clsim) appears to be the best choice among the local methods.

With regard to non-pillar community structures, we have observed a considerable reduction in the achieved accuracy scores for almost all methods. This observation raises the following question: what kind of assumptions are considered by different methods when multiplex communities are identified? It is clear that there is a tendency, even if not explicitly declared, to assume that multiplex communities are pillar communities expanding over all the layers of the multiplex network. For instance, multi-slice modularity [30] rewards pillar communities when calculating the modularity score, meaning that any community structure that is not pillar is less optimal, or of lower quality, than pillar communities. While pillar community structures are perfectly reasonable and can be observed in many relevant scenarios, we claim that they also represent a simplification of many real-world scenarios. Assuming that multiplex approaches have been developed and are used mainly to overcome the oversimplification of monoplex networks, relying on a single type of ideal community structure seems, at least, a missed opportunity. Thus, more work has to be done on improving the accuracy of community detection methods for non-pillar community structures.

A second set of considerations can be drawn by looking at the results obtained by the evaluated methods when applied to real-world datasets. Our experiments have shown that, on real-world datasets, the community structures detected largely differ from the ground-truth. This raises two interesting questions: the first question is to which extent is the assumed ground-truth itself a valid assumption? In other words, does the ground-truth given for a real dataset always describe the community structures identified by a community detection method, or does it capture only one part of the whole picture? The answer to this question is never trivial even in monoplex networks. Nevertheless it is easy to see how adding additional layers of relations makes it further complicated. For example, both Dkpol and Aucs ground-truths group together individuals belonging to the same ”organization" (political parties in one case and research groups in the other). The question then becomes whether it is reasonable to assume that the selected relations, observed in the multiplex networks, will produce a community structure corresponding to this formal grouping, and to some extent, how different relations (thus different layers) can be more or less aligned with the hypothesis described above. Will members of the same research group work together, or publish together? Have lunch and fun together? Will members of the same political party retweet each other on Twitter, and reply to each other? Indeed, looking at the accuracy of the community structures identified for the real world dataset, especially in the case of Dkpol, one might ask whether we are observing a generalized failure of the community detection methods, or conversely, whether the community detection methods were actually able to observe relevant structures that just were different from the community structures assumed in the ground-truth.

The second question, which is strongly related to the first one, is whether all the layers included in these datasets positively contribute to an accurate identification of the community structure in these datasets, or whether some of them add more noise that heavily affect the identification process. Indeed, the fact that a community detection method always (or mostly always) gives an output — no matter what layers are included in the input multiplex network — does not make adding more layers to the model a good choice. Layers, besides being defined by a specific internal topology, are also defined by internal logic that might or might not be coherent with those of the other layers.

The Dkpol dataset represents a good example of this problem since some in-detail analysis of the three layers composing the multiplex network has shown that retweets and following/follower interactions follow a relatively associative dynamic for political parties. The replies, however, follow a negative associative dynamic for political parties. Here we think that more efforts have to be made in the modelling phase of the multiplex network and some layer-specific measures should be developed to lead the choice of the layers that contribute to the identification of the communities.

A final consideration should be made about the similarities of the obtained results. Focusing, for the above-mentioned reason, mainly on the results obtained from the synthetic networks, it is possible to observe some general patterns. Global partitioning methods show a remarkable level of similarity in detecting community structures based on a pillar-like model. Semi-pillar and hierarchical community structures show a lower degree of similarity between the retrieved community structures. Overlapping methods are less stable and while they still perform better on pillar communities than on other types of structures, they still show a higher level of diversity on every different configuration we have tested.

Local methods show a behavior that is, to some extent, similar to the global partitioning methods. When tested on pillar-communities they show a remarkable similarity between the produced communities, which can easily lead to calling them interchangeable. Nevertheless, the less pillar-like the community structure in the data is, the higher the differences seem to be at first between ACLcut and ML-LCD and then also between the same algorithm.

Scalability analysis of global methods, like ABACUS, Infomap and MLP proved to be extremely scalable with respect to number of actors in the multiplex network. As regards scalability with respect to the number of layers, Infromap, GLouvain and MLP proved to be extremely scalable. As for local methods, ACLcut seems to be extremely scalable.

It should be clear the this extensive overview cannot easily answer the question: "Which algorithm should I use?". Nevertheless, it is possible to list a set of considerations that should be asked when a specific algorithm is selected. Considerations like: "What kind of community am I interested in? What is its structure in the multiplex network? How far is my expected community structure from a pillar model?" As we have shown, pillar-like communities can be well detected with the methods we have available, with Infomap and GLouvain achieving very good results. However, the more we move away from that ideal model of multiplex community structure, the more the expected accuracy drops and the differences between various algorithms become more pronounced.

7 CONCLUSION

In this work, we have provided a taxonomy of approaches for the problem of multiplex community detection, which covers the most relevant methods in the literature that deal with this problem. We have characterized the different algorithms based on several properties and discussed the type of communities detected by each method. Upon this, we have theoretically compared the methods under study. A major contribution of this work is in an extensive evaluation of the reviewed methods according to different criteria, including accuracy with respect to given ground-truth on both synthetic and real-world multiplex networks, scalability in terms of the size of the multiplex network, and robustness against their model parameters.

A APPENDIX: EVALUATION DATASETS

We selected three types of datasets: (i) *real networks* widely used in the literature, for two of which the ground truth is available, (ii) *networks* generated using the mLFR benchmark [8], and (iii) *community structure-controlled synthetic networks* generated by forcing specific community structures. The following is a description of the evaluation networks supported by our motivation of choice for each of them.

A.1 Real networks

For real networks, a selection of publicly available multiplex networks have been chosen to cover different properties and domains of multiplex networks. More specifically, we selected the following networks:

- **Aucs**: In this multiplex network, the multiple layers refer to different relationships between 61 employees/PhD students at a University department. The relationships are **(i)** Being friends on Facebook, **(ii)** Having lunch together at the university, **(iii)** Co-working, **(iv)** Co-authoring, **(v)** Offline friendship [29]. The ground truth of this dataset reports the affiliation of actors to research groups.
- **Dkpol**: This is a Twitter interaction network among 490 danish politicians during the month leading up to the parliamentary elections of 2015. The three layers model the different Twitter interactions *follow*, *reply*, and *retweet* among these politicians. A ground truth for this dataset is available in the form of pairs ⟨politician name, political affiliation⟩. The political affiliation is one of the major ten political parties in Denmark (i.e., Alternativet, Radikale Venstre, Enhedslisten, Socialdemokratiet and Socialistisk Folkeparti,Dansk Folkeparti, KristenDemokraterne, Liberal Alliance, Venstre, or Det Konservative Folkeparti).
- **Airports**. This multiplex network models the connections between 417 European airports on a certain day. Each of the 37 layers in this dataset models the connections made by one commercial airline [9].
- **Rattus**. This is a multiplex network that models different types of genetic interactions for organisms in the Biological General Repository for Interaction Datasets (BioGRID, thebiogrid.org) - a public database that archives and disseminates genetic and protein interaction data from humans and model organisms. We used BioGRID 3.2.108 (updated 1 Jan 2014) which concerns Rattus Norvegicus that makes use of the following layers: physical association, direct interaction, colocalization, association, additive genetic, interaction defined by inequality and suppressive genetic interaction defined by inequality [36]

A.2 Benchmark networks

For single-layer networks, there is a number of widely accepted benchmarks to generate artificial graphs. *GN benchmark* [16] is used to generate fixed size graphs (128 nodes) with nodes of the same degree (approximately 16) and communities of the same size (four 32-node communities). *LFR benchmark* [25, 26] is used to generate more realistic graphs with built-in community structure such that both the degree of the nodes and the community size distributions follow a power law distribution. An extension of LFR benchmark proposed by the same authors gives the possibility to generate weighted and directed graphs with overlapping communities. *RB-LFR benchmark* [44] is an extension of LFR benchmark used to generate graphs with hierarchical community structure.

As for multiplex networks, to the best of our knowledge, the only available benchmark is *mLFR benchmark* [8], which is an extension of the LFR benchmark that considers the multi-layer aspect of multiplex networks. This extension first creates one layer using the LFR benchmark then uses that layer as a template (base) to generate the other layers by distributing edges of the same group on the different layers. mLFR benchmark introduces several parameters of which the most important for our experiments are: number of layers (#l), number of actors (#a - the number of nodes in each layer), and mixing parameter (μ), which expresses the proportion between the external degree of a node (number of connections with nodes from other communities) to the total degree of a node.

To test the scalability of different community detection algorithms, two groups of synthetic multiplex networks were generated. The mixing parameter in both groups was fixed to $\mu = 0.2$ as the default value recommended by the author to guarantee a community structure. The two groups are:

- **Group I**: Aimed at testing the scalability with respect to the number of actors. To analyze the effect of increasing the number of layers, 12 multiplex networks were generated for each of three different values of $\#l$ ($\#l = 2$, $\#l = 5$, and $\#l = 10$ layers) with $\#a \in \{100, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000\}$.
- **Group II**: Aimed at testing the scalability with respect to the number of layers. For this purpose, 50 different multiplex networks were generated with 50 different values of $\#l$ in the range (1,50). To keep the focus on the number of layers, the number of actors in these networks was fixed to $\#a = 1000$ as used in [8].

Since it is not possible to force different types of multiplex community structures using the mLFR benchmark, we used the multiplex networks generated by this benchmark only for assessing the scalability of the different methods. Conversely, in order to assess the accuracy, we considered the real datasets associated with a ground truth or other artificially created multiplex networks with a controlled community structure.

A.3 Community structure-controlled synthetic networks

Different multiplex community detection methods have different underlying assumptions about what a multiplex community is. To shed light on these assumptions, we generated 8 different multiplex networks with 8 different built-in community structures. To keep the focus on the community structure, each of the 8 multiplex networks is comprised of 3 layers, 100 actors, and 300 nodes (100 per layer). After forcing a specific community structure on each multiplex, the edges were generated with a probability $P_{in} = 0.7$ to be internal (within a community) and a probability $P_{ext} = 0.01$ to be external (among communities). The following is a brief description of each multiplex network:

- **Pillar Equal Partitioning (PEP)**: The community structure in this multiplex is a set of pillar non-overlapping communities that are approximately equal in size. (Figure 6a).
- **Pillar Equal Overlapping (PEO)**: Similar to PEP in terms of the size of the communities and the pillar structure. The communities in PEO are however overlapping (Figure 6b).
- **Pillar Non-Equal Partitioning (PNP)**: The community structure in this multiplex is a set of pillar non-overlapping communities. As to the size distribution of the communities, there are few big pillar communities and many small pillar communities (Figure 6c).
- **Pillar Non-Equal Overlapping (PNO)**: Similar to PNO in terms of the community size distribution and the pillar structure. The communities in PNO are however overlapping (Figure 6d).
- **Semi-pillar Equal Partitioning (SEP)**: The community structure in this multiplex is a set of semi-pillar non-overlapping communities that are approximately equal in size and a set of single-layer communities (Figure 6e).
- **Semi-pillar Equal Overlapping (SEO)**: Similar to SEP except that the semi-pillar communities are overlapping (Figure 6f).
- **Non-pillar Hierarchical Non-equal (NHN)**: The community structure in this multiplex reflects some hierarchy among communities on the actor level. Some big node-level communities (like C_7 in Figure 6g) on a layer L_3 are constituted of smaller communities (on the actor-level but not the node-level) on layer L_2.
- **Non-pillar Non-equal Mixed (NNM)**: The community structure in this multiplex is a set of single-layer communities some of which are overlapping (Figure 6h).

Table 6. Basic statistics about the communities in the **Synthetic multiplex networks with controlled community structure**. **#c**: number of communities, **sc1**: size, in nodes, of the biggest community, **sc2/sc1**: ratio between size of the second largest to the size of the largest community, **%n**: percentage of nodes assigned to a community, **%p**: percentage of pillars assigned to a community (a pillar is all nodes of an actor), **%ao** percentage of actors appeared in more than one community, and **%no** percentage of nodes appeared in more than one community

mutliplex	#c	sc1	sc2/sc1	%n	%p	%ao	%no
PEP	10 (all pillar)	30	1.00	1.00	1.00	0.00	0.00
PEO	10 (all pillar)	39	0.77	1.00	1.00	0.27	0.27
PNP	10 (all pillar)	90	0.67	1.00	1.00	0.00	0.00
PNO	10 (all pillar)	99	0.70	1.00	1.00	0.26	0.26
SEP	19 (10 semi-pillar)	20	0.50	0.97	0.00	0.90	0.00
SEO	19 (10 semi-pillar)	26	0.77	0.97	0.00	0.90	0.19
NHN	18 (10 small, 5 medium, 3 big)	40	0.75	1.00	0.00	1.00	0.00
NNM	6	30	0.67	0.35	0.00	0.40	0.14

Table 6 provides information about the communities in these multiplex networks and Figure 6 illustrates the different types of multiplex community structures.

B APPENDIX: COMMUNITY STRUCTURE STATISTICS ON REAL DATASETS

Statistics of each method occupy one row in each a table (multiple settings of the input parameters for some methods are represented as separated entries for the same method). Since some of the methods are non-deterministic, we carried out each of such methods multiple times (10 times), and provided the average values. Moreover, each method was given an upper limit of 24 hours of the machine time to give an output, else the execution was suppressed. Therefore, we use "NA" row if the execution was suppressed because of the long execution time (longer than 24 hours). Here we present the general statistics about the communities detected using multiplex community detection methods on Dkpol and Rattus datasets (Table 7)

C APPENDIX: STANDARD DEVIATION OF THE AVERAGE JACCARD INDEX USED FOR PAIRWISE ANALYSIS AMONG LOCAL METHODS

Figures 22, 23 report the standard deviation of the average pairwise similarity among local methods on real-world datasets and synthetic datasets reported in Section 5.2.1

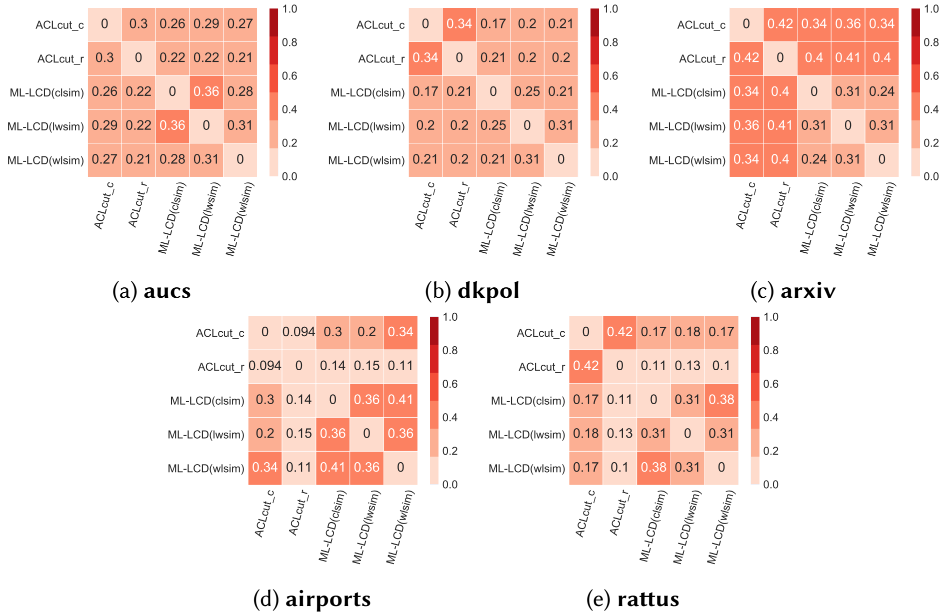
Table 7. Basic statistics for real datasets

method	#c	sc1	sc2/sc1	%n	%p	%ao	%no	%s
NWF	27.30	782.50	0.01	1.00	1.00	0.00	0.00	0.74
WF_EC	1.00	839.00	1.00	1.00	1.00	0.00	0.00	0.00
MLP	51.60	585.10	0.07	0.89	0.83	0.00	0.00	0.63
$\text{GLouvain}_{(hOm)}$	8.30	179.80	0.88	1.00	0.90	0.04	0.00	0.00
$\text{GLouvain}_{(lOm)}$	12.40	144.60	0.94	1.00	0.59	0.23	0.00	0.01
LART	427.00	102.00	0.40	1.00	0.93	0.03	0.00	0.95
$\text{Infomap}_{(no)}$	1.00	839.00	1.00	1.00	1.00	0.00	0.00	0.00
$\text{Infomap}_{(o)}$	22.00	839.00	0.23	1.00	1.00	0.26	0.44	0.45
$\text{ML-CPM}_{(31)}$	NA	NA	NA	NA	NA	NA	NA	NA
$\text{ML-CPM}_{(42)}$	9.00	24.00	0.91	0.12	0.16	0.07	0.07	0.00
$\text{ABACUS}_{(31)}$	52.70	437.80	0.11	0.83	0.41	0.47	0.42	0.00
$\text{ABACUS}_{(42)}$	28.00	45.60	0.80	0.41	0.40	0.23	0.12	0.00

Real dataset (Dkpol)

method	#c	sc1	sc2/sc1	%n	%p	%ao	%no	%s
NWF	486.60	563.60	0.41	1.00	1.00	0.00	0.00	0.21
WF_EC	113.00	2400.00	0.05	1.00	1.00	0.00	0.00	0.05
MLP	504.70	573.20	0.42	0.80	0.84	0.00	0.00	0.20
$\text{GLouvain}_{(hOm)}$	140.80	575.40	0.46	1.00	1.00	0.00	0.00	0.04
$\text{GLouvain}_{(lOm)}$	147.10	595.60	0.49	1.00	0.89	0.05	0.00	0.04
LART	3237.00	11.00	0.54	1.00	0.65	0.19	0.00	0.99
$\text{Infomap}_{(no)}$	116.00	2081.00	0.08	1.00	1.00	0.00	0.00	0.05
$\text{Infomap}_{(o)}$	295.00	2157.00	0.19	1.00	1.00	0.18	0.41	0.04
$\text{ML-CPM}_{(31)}$	63.00	219.00	0.03	0.12	0.46	0.11	0.06	0.00
$\text{ML-CPM}_{(42)}$	NA	NA	NA	NA	NA	NA	NA	NA
$\text{ABACUS}_{(31)}$	314.00	520.29	0.41	0.77	0.69	0.12	0.03	0.00
$\text{ABACUS}_{(42)}$	4.20	10.00	0.80	0.01	0.90	0.00	0.00	0.00

Real dataset (Rattus)



(a) **aucs** (b) **dkpol** (c) **arxiv**

(d) **airports** (e) **rattus**

Fig. 22. Standard deviation for average pairwise similarity among the different local method when the same seed is used as an input (Real networks)

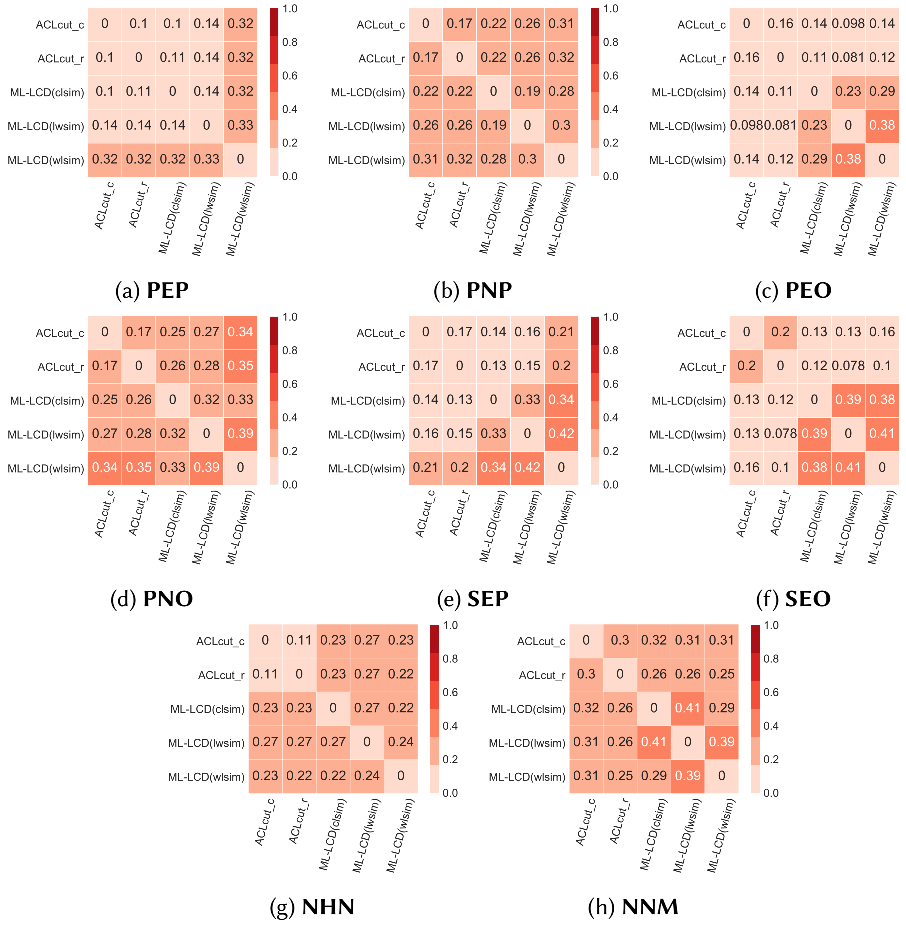


Fig. 23. Standard deviation for average pairwise similarity among the different local method when the same seed is used as an input (Synthetic networks)

REFERENCES

[1] Nazanin Afsarmanesh and Matteo Magnani. Finding overlapping communities in multiplex networks. 2018.

[2] Michael J. Barber. Modularity and community detection in bipartite networks. *Physical Review E*, 76:066102, 2007.

[3] Michele Berlingerio, Michele Coscia, and Fosca Giannotti. Finding and characterizing communities in multidimensional networks. In *International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 490–494. IEEE Computer Society Washington, DC, USA, 2011.

[4] Michele Berlingerio, Fabio Pinelli, and Francesco Calabrese. ABACUS: frequent pAttern mining-BAsed Community discovery in mUltidimensional networkS. *Data Mining and Knowledge Discovery*, 27(3):294–320, 2013.

[5] Brigitte Boden, Stephan Günnemann, Holger Hoffmann, and Thomas Seidl. Mining coherent subgraphs in multi-layer graphs with edge labels. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12*, page 1258. ACM Press, aug 2012. ISBN 9781450314626. doi: 10.1145/2339530.2339726.

[6] Cecile Bothorel, Juan David Cruz, Matteo Magnani, and Barbora Micenkova. Clustering attributed graphs: models, measures and methods. *Network Science*, 3(3):408–444, 2015.

[7] Oualid Boutemine and Mohamed Bouguessa. Mining Community Structures in Multidimensional Networks. *ACM Transactions on Knowledge Discovery from Data*, 11(4):1–36, jun 2017. ISSN 15564681. doi: 10.1145/3080574.

[8] Piotr Bródka. A method for group extraction and analysis in multilayer social networks. *CoRR*, abs/1612.02377, 2016.

[9] Alessio Cardillo, Jesús Gómez-Gardeñes, Massimiliano Zanin, Miguel Romance, David Papo, Francisco del Pozo, and Stefano Boccaletti. Emergence of network features from multiplexity. *Scientific Reports*, 2013.

[10] Linda M. Collins and Clyde W. Dent. Omega: A General Formulation of the Rand Index of Cluster Recovery Suitable for Non-disjoint Solutions. *Multivariate Behavioral Research*, 23(2):231–242, apr 1988. ISSN 0027-3171.

[11] Manlio De Domenico, Andrea Lancichinetti, Alex Arenas, and Martin Rosvall. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X*, 5:011027, Mar 2015. doi: 10.1103/PhysRevX.5.011027.

[12] Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora. ARTICLE Structural reducibility of multilayer networks. *Nature Communications*, 6, 2015. doi: 10.1038/ncomms7864.

[13] Mark E. Dickison, Matteo Magnani, and Luca Rossi. *Multilayer Social Networks*. 2016.

[14] Xiaowen Dong, Pascal Frossard, Pierre Vandergheynst, and Nikolai Nefedov. Clustering on Multi-Layer Graphs via Subspace Analysis on Grassmann Manifolds. *IEEE Transactions on Signal Processing*, 62(4):905–918, feb 2014. ISSN 1053-587X. doi: 10.1109/TSP.2013.2295553.

[15] Daniel Edler, Ludvig Bohlin, and Martin Rosvall. Mapping higher-order network flows in memory and multilayer networks with infomap. *CoRR*, abs/1706.04792, 2017.

[16] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002. ISSN 0027-8424. doi: 10.1073/pnas.122653799.

[17] Roger Guimera, Marta Sales-Pardo, and Luis A. Nunes Amaral. Module identification in bipartite and directed networks. *Physical Review E*, 76:036102, 2007.

[18] Obaida Hanteer and Luca Rossi. The meaning of dissimilar: An evaluation of various similarity quantification approaches used to evaluate community detection solutions. In *Proceedings of the 2019 IEEE/ACM International Conference onAdvances in Social Networks Analysis and MiningFAB 2019, FOSINT-SI 2019, HI-BI-BI 2019*, pages 513–518, Vancouver, Canada, 2019. ISBN 978-1-4503-6868-1.

[19] Roberto Interdonato, Andrea Tagarelli, Dino Ienco, Arnaud Sallaberry, and Pascal Poncelet. Node-centric community detection in multilayer networks with layer-coverage diversification bias. apr 2017.

[20] Lucas G. S. Jeub, Michael W. Mahoney, Peter J. Mucha, and Mason A. Porter. A local perspective on community structure in multilayer networks. *Network Science*, 5(2):144âĂŞ163, 2017. doi: 10.1017/nws.2016.22.

[21] Inderjit S Jutla, Lucas G S Jeub, and Peter J Mucha. A generalized Louvain method for community detection implemented in Matlab. Technical report, 2011-2017. URL http://github.com/GenLouvain.

[22] Jungeun Kim and Jae-Gil Lee. Community Detection in Multi-Layer Graphs. *ACM SIGMOD Record*, pages 37–48, dec 2015.

[23] Jungeun Kim, Jae-gil Lee, and Sungsu Lim. Differential Flattening: A Novel Framework for Community Detection in Multi-Layer Graphs. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(2):27, 2016. ISSN 2157-6904.

[24] Zhana Kuncheva and Giovanni Montana. Community Detection in Multiplex Networks using Locally Adaptive Random Walks. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15*, pages 1308–1315. ACM Press, 2015. ISBN 9781450338547. doi: 10.1145/2808797.2808852.

[25] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. apr 2009. doi: 10.1103/PhysRevE.80.016118.

[26] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. may 2008. doi: 10.1103/PhysRevE.78.046110.

[27] Huajing Li, Zaiqing Nie, Wang-Chien Lee, Lee Giles, and Ji-Rong Wen. Scalable community discovery on textual data with relations. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 1203–1212, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. doi: 10.1145/1458082.1458241.

[28] Chuan Wen Loe and Henrik Jeldtoft Jensen. Comparison of communities detection algorithms for multiplex. *Physica A: Statistical Mechanics and its Applications*, 431:29–45, 2015. ISSN 03784371. doi: 10.1016/j.physa.2015.02.089.

[29] Matteo Magnani and Luca Rossi. Towards effective visual analytics on multiplex and multilayer networks. 72, 01 2015.

[30] Peter J. Mucha and Mason A. Porter. Communities in multislice voting networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(4), 2010. ISSN 10541500. doi: 10.1063/1.3518696.

[31] Gabriel Murray, Giuseppe Carenini, and Raymond Ng. Using the omega index for evaluating abstractive community detection. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 10–18. Association for Computational Linguistics, 2012.

[32] G. Qi, C. C. Aggarwal, and T. Huang. Community detection with edge content in social media networks. In *2012 IEEE 28th International Conference on Data Engineering(ICDE)*, volume 00, pages 534–545, 04 2012. doi: 10.1109/ICDE.2012.77.

[33] Nandini Raghavan, RÃČÂĪka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. 76:036106, 10 2007.

[34] Yiye Ruan, David Fuhry, and Srinivasan Parthasarathy. Efficient Community Detection in Large Networks using Content and Links. 2012.

[35] Arlei Silva, Wagner Meira, Jr., and Mohammed J. Zaki. Mining attribute-structure correlated patterns in large attributed graphs. *Proc. VLDB Endow.*, 5(5):466–477, January 2012. ISSN 2150-8097. doi: 10.14778/2140436.2140443.

[36] Chris Stark, Bobby-Joe Breitkreutz, Teresa Reguly, Lorrie Boucher, Ashton Breitkreutz, and Mike Tyers. BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34(Database issue):D535–D539, jan 2006. ISSN 0305-1048. doi: 10.1093/nar/gkj109.

[37] Y Sun and Jiawei Han. Mining heterogeneous information networks: a structural analysis approach. *ACM SIGKDD Explorations Newsletter*, 14(2):20–28, 2013.

[38] Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhijun Yin, Hong Cheng, and Tianyi Wu. RankClus: integrating clustering with ranking for heterogeneous information network analysis. In *International Conference on Extending Database Technology (EDBT)*, pages 565–576. ACM Press, 2009. ISBN 9781605584225. doi: 10.1145/1516360.1516426.

[39] Yizhou Sun, Yintao Yu, and Jiawei Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, page 797. ACM Press, 2009. ISBN 9781605584959. doi: 10.1145/1557019.1557107.

[40] Andrea Tagarelli, Alessia Amelio, and Francesco Gullo. Ensemble-based community detection in multilayer networks. *Data Mining and Knowledge Discovery*, 31(5):1506–1543, Sep 2017. ISSN 1573-756X. doi: 10.1007/s10618-017-0528-8.

[41] Lei Tang, Xufei Wang, and Huan Liu. Uncoverning groups via heterogeneous interaction analysis. *2009 Ninth IEEE International Conference on Data Mining*, pages 503–512, 2009.

[42] Lei Tang, Xufei Wang, and Huan Liu. Community detection via heterogeneous interaction analysis. *Data Mining and Knowledge Discovery*, 25(1):1–33, jul 2012. ISSN 1384-5810. doi: 10.1007/s10618-011-0231-0.

[43] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. A model-based approach to attributed graph clustering. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 505–516, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1247-9. doi: 10.1145/2213836.2213894.

[44] Zhao Yang, Juan I Perotti, and Claudio J Tessone. Hierarchical benchmark graphs for testing community detection algorithms. Technical report, 2017.

[45] Yang Zhou and Ling Liu. Social influence based clustering of heterogeneous information networks. *ACM SIGKDD international conference on Knowledge discovery and data mining*, page 338, 2013. doi: 10.1145/2487575.2487640.

[46] Yang Zhou, Hong Cheng, and Jeffrey Xu Yu. Graph clustering based on structural/attribute similarities. *Proc. VLDB Endow.*, 2(1):718–729, 2009. ISSN 2150-8097. doi: 10.14778/1687627.1687709.

Appendix B

Unspoken Assumptions Behind Multilayer Modularity Maximization

Unspoken Assumptions In Multi-layer Modularity Maximization

Obaida Hanteer[1,*] **and Matteo Magnani**[2,+]

[1]IT University of Copenhagen, Copenhagen, 2300, Denmark
[2]InfoLab, Department of Information Technology, Uppsala University, Uppsala, Sweden
[*]obha@itu.dk
[+]matteo.magnani@it.uu.se

ABSTRACT

A principled approach to recover communities in social networks is to find a clustering of the network nodes into modules (i.e groups of nodes) for which the modularity over the network is maximal. This guarantees partitioning the network nodes into sparsely connected groups of densely connected nodes. A popular extension of modularity has been proposed in the literature so it applies to multi-layer networks, that is, networks that model different types/aspects of interactions among a set of actors. In this extension, a new parameter, the coupling strength ω, has been introduced to couple different copies (i.e nodes) of the same actor with specific weights across different layers. This allows two nodes that refer to the same actor to reward the modularity score with an amount proportional to ω when they appear in the same community. While this extension seems to provide an effective tool to detect communities in multi-layer networks, it is not always clear what kind of patterns maximising the generalised modularity can identify in multi-layer networks and whether these patterns are inclusive to all possible community structures prone to evolve in multi-layer networks. In addition, it has not been thoroughly investigated yet how to interpret ω in real-world scenarios, and whether a proper tuning of ω, if exists, is enough to guarantee an accurate recoverability for different types of multi-layer community structures. In this article, we investigate the meaning of ω in the generalised modularity formula and provide a review of the different ways used in the literature to tune that parameter. We analyse the different patterns that can be identified by maximising the generalised modularity in relation to ω. We propose different models for multi-layer communities in multiplex and time-dependent networks and test if they are recoverable by modularity-maximization community detection methods under any assignment of ω. Our main finding is that only few simple models of multi-layer communities in multiplex and time-dependent networks are recoverable by modularity maximisation methods while more complex models are not accurately recoverable under any assignment of ω.

Introduction

Community detection is one of the core tasks in the analysis of complex networks. It involves partitioning the network into groups of nodes, also known as communities, partitions, cohesive groups or clusters. The importance of this task comes from its ability to break large networks into smaller building blocks so it becomes easier to understand the structure of the network and of the role played by each node. Applications of community detection algorithms are ubiquitous, including social media group detection[1], thematic community detection[2], second-order flow analysis in human mobility[3], and topic detection in information networks[4]. A community detection solution is usually represented as a clustering $\mathscr{C}$ where $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$, and C_1, C_2, $\ldots$, C_k are disjoint or non-disjoint groups of nodes.

Although there is no one agreed-upon definition for communities in networks, it is widely accepted in mono-layer networks to consider a community, loosely speaking, the group of nodes that are more densely connected with each other than with the rest of the network[5]. This definition is mainly inspired by the quality function modularity[6,7], which is a function $f(\mathscr{C})$ that produces values in the range $[-1, 1]$ and for a given clustering $\mathscr{C}$ over a network, it measures the extent to which the network provides a good fit for $\mathscr{C}$ with respect to the definition of communities mentioned above. In other words, it measures the extent to which the network nodes are more densely connected within the communities $\{C_1, C_2, \ldots, C_k\} \in \mathscr{C}$ than across these communities. While modularity was firstly proposed as a quality measure to evaluate the accuracy of community detection methods at the time[6], it gave birth to a new class of community detection methods that interpreted the community detection problem into finding a clustering with the maximal modularity over the network. When applied on mono-layer networks, one expects to retrieve groups of nodes where the within-group edges are denser than those connecting nodes from different groups.

In response to the advances on complex network analysis, the multi-layer network model has been proposed as an efficient tool to consider different types and/or different time-windows of interactions in a given system[8]. The intuition behind such generalisation is that interactions among a set of actors do not happen in isolation. Instead, they inherit certain dependencies among each other such that one might cause the other one, or one can be used as predictor for the other one. Think of an example

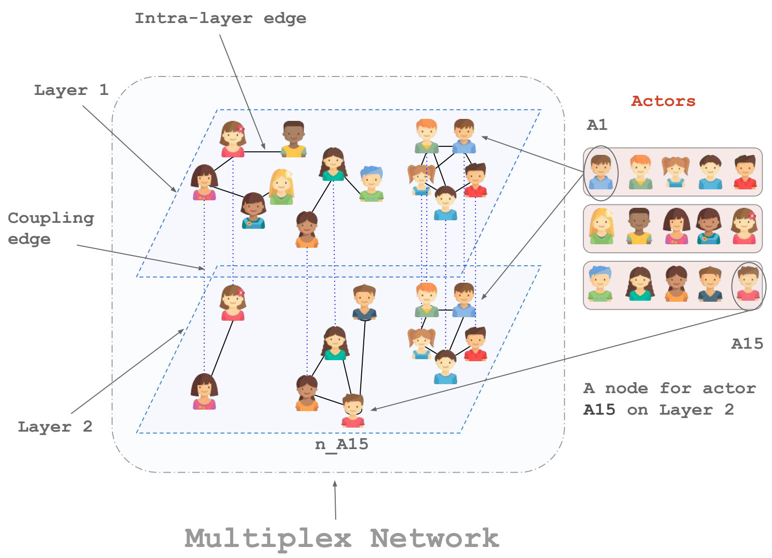


Figure 1. A toy-example of a multiplex network with two types of interaction represented by Layer 1 and Layer 2 among 15 actors. The two nodes existing in both layers and representing the same actor (e.g. the same person) are linked by a dotted line s

where two people are friends on Facebook. The probability of having those two connected via other means of communication (i.e, Twitter, Whatsapp, ect) is much higher than if they were not connected via any communication mean. Hence, studying these interactions separately without accounting for any dependencies among them might yield into misleading results. Two special cases of multi-layer networks are multiplex networks and time-dependent networks. In the former, different layers refer to different types of interactions among a set of actors while in the latter, different layers refer to different time-windows of an interaction among a set of actors. While the general multi-layer framework can consider both aspects in one model, we keep our focus in this article on those two special cases for simplicity. Unless mentioned otherwise, we will refer to those two cases as multi-layer networks throughout the article. Figure 1 shows a typical layered representation of a multiplex network where each of the layers corresponds to a different type of interaction, and nodes in different layers can be associated to the same actor (the same person for example). The same figure can also be seen as a time-dependent network if layers were to refer to time-windows of an interaction instead. Here, we adopt the term *actor* from the field of social network analysis, where multiplex networks and time-dependent networks have been first applied, and the term *layer* from recent generalizations of these models[8].

In spite of the plethora of methods developed for detecting communities in multi-layer networks, there is very little work on proposing a definition for multi-layer communities, like the mono-layer community definition. Recent work[9] argues that there are two primary differences between a mono-layer community and a multi-layer one. First, a multi-layer community can expand over multiple layers. Second, edges of a multi-layer community in one layer might depend on the connectivity patterns in another layer. For example, in a multiplex network that models different Twitter interaction among a set of users (i.e, Following and Retweeting each represented as a layer in a 2-layer multiplex network), it might be the case that the re-tweet edges among a set of actors are largely dependent on whether these actors follow each other or not, which might explain the existence of multi-layer communities that expand over the two layers.

Undoubtedly, the generalisation proposed by the multi-layer framework has introduced new challenges to the community detection problem. While some researchers worked on that by tailoring new community detection methods for multi-layer networks[10], there has been more tendency to extend some of the already existent methods used with mono-layer networks like collapsing the multiple layers into a mono-layer network and then use any of the mono-layer community detection methods[11], extended label propagation[12] and extended clique percolation[13] just to mention few. A popular extension from mono-layer to multi-layer is the extension of the quality function *modularity* into the *generalised modularity*[14]. In that extension, the authors introduce a new parameter, the coupling strength ω, to the modularity function. The new proposed parameter, ω, assumes that nodes of the same actor in different layers are coupled via coupling edges and these coupling edges are weighted with an amount equals to ω. With that extension, a modularity maximisation method does not only maximise the within-community intra-layer edges and minimise the cross-community intra-layer edges, but also maximises the sum of coupling edges weights, i.e nodes that belong to the same actor, within a community given ω. Indeed, with ω being the only multi-layer ingredient in the generalised modularity formula, there has been few tries in the literature to tune ω such that it reflects some of the inter-layer information across different layers (i.e closeness between layers, common neighbour across layers, etc.). It is not clear, however, how these different interpretations affect multi-layer community detection using modularity maximisation and whether the modularity function responds to these interpretations by providing clusterings that support the intuition behind them. For example, the intuition behind tuning ω such that it reflects the percentage of common neighbours among nodes in different layers is that it makes more sense for multi-layer communities to group nodes of the same actor in one community only if they have similar neighbourhoods across the layers[15]. The question is whether modularity responds to that tuning by clustering the network nodes into communities that respect that intuition. We claim that methods that maximise the generalised modularity inherited the popularity and the success mono-layer modularity maximisation methods had before, yet the patterns identified by these methods are not thoughtfully investigated enough. The interpretation of coupling edges and coupling strength is not well elaborated on in the literature, which makes tuning these parameters in real-world scenarios a cumbersome task. In addition, with the absence of a precise definition of multi-layer communities, it is not clear whether different patterns identified by maximising multi-layer modularity under all possible assignments of ω in a multi-layer network are inclusive to all community structures that are prone to evolve in multi-layer networks.

The goal of this article is to analyse the assumptions behind the generalised modularity and the effect of that on the possible patterns that can be identified in multi-layer networks as a result of maximising that quality function. We investigate the meaning of ω in the generalised modularity and review different ways used in the literature to tune that parameter. We provide different models for communities in multi-layer networks and test the ability of modularity maximisation based methods to recover these community models under any possible parameter tuning. We conclude from our experiments that only few simple models of multi-layer communities in multiplex and time-dependent networks are recoverable by modularity maximisation methods while more complex models are not accurately recoverable under any parameter tuning. The novelty of this article comes in the community models it proposes to model different community structures which can exist in multi-layer networks, and the discussion about the limitations of the highly popular and extensively referenced methods based on modularity maximisation for recovering some patterns in multi-layer networks. It is worth mentioning that we chose to opt out from digging deeply into the mathematical representation of different concepts in this article so we provide the most general and abstract mathematical representation when needed. The aim is to keep our focus on the intuition behind some concepts rather than the interpretation of that into mathematical formulae. Unless mentioned otherwise, we will refer to the generalised modularity as modularity, and to the modularity applicable only on mono-layer networks as mono-layer modularity.

Our article is structured as follows. Since understating the generalised modularity requires a very good understanding of mono-layer modularity, we first start by defining modularity in mono-layer networks (section 1) followed by the extension of mono-layer modularity into the generalised modularity in section 2. We study the patterns that can be identified by maximising modularity in multi-layer networks as a function of ω in section 3 and then provide models of multi-layer communities possible to evolve in multiplex and time-dependent networks in section 4. We report the results of our experiments on the recoverability of these models using modularity maximisation based community detection methods in section 5. We discuss our findings in section 6, and report the methods used for our experiments in section 7 .

1 Modularity in Mono-layer Networks

As firstly proposed by[6,7], the modularity of a clustering $\mathscr{C}$ over a mono-layer network characterized by an adjacency matrix A, where $A_{(i,j)}$ is 1 if there is an edge between nodes i and j and zero otherwise, can be written as:

$$Q = \frac{1}{2|E|} \sum_{C \in \mathscr{C}} \sum_{(i,j) \in C} \left[A_{(i,j)} - P_{(i,j)} \right] \tag{1}$$

where $|E|$ is the number of edges in the network. The summation is performed only over pairs of nodes that belong to the same community $C \in \mathscr{C}$. $P_{(i,j)}$ is the probability of having an edge between nodes i and j in a null model.

Away from its mathematical representation, the intuition behind modularity is to measure the extent to which the distribution of edges in a network is far from what one would expect shall the edges were distributed in a community-less manner. It can be seen as a normalised distance between a network N that has a specific edge distribution and an equivalent network, usually referred to as the null model, where edges are distributed randomly. Two common choices for the null model are: the uniform random model, where the probability of having an edge among any pair of nodes in the network is fixed, or the non-uniform random model, where the probability of having an edge among two nodes depends on their degrees (also called a null model with preferential attachment dynamic).

Modularity assumes that nodes within a community tend to interact more densely with each other than with the rest of the network. Hence, the quality of a community, from a modularity perspective, is in the percentage of within-community edges out of all edges incident to nodes of that community. With that being said, we stress out the fact that modularity is not a property of the network, but rather a property of a clustering over the network. It is common, however, to describe a network as modular and that is to refer to the existence of a significantly modular clustering over that network. The modularity of a clustering $\mathscr{C}$ in a network can be translated into calculating the normalised sum of the network edges contributions to $\mathscr{C}$. While iterating through network edges, some edges come as a surprise (they exist in the network, but not highly probable to appear in the null model), and others are expected (highly probable in the null model). Surprising edges contribute more to the final score when they happen to be within a community. Equation 1 therefore guarantees:

[P1] Rewarding existent edges within communities. Each edge connecting two nodes i and j in a community $C_x \in \mathscr{C}$ contributes positively to the total sum with an amount equals the difference $(1 - P_{i,j})$. Clearly, edges that come as a surprise, i.e are not expected in the null model and hence result in a very low value of $P_{i,j}$, contribute more to the final modularity than the expected ones.

[P2] Punishing non-existent edges within communities. If two nodes i, j happened to be in the same community $C_x \in \mathscr{C}$ and they are not directly connected (i.e $A_{i,j} = 0$), this contributes negatively to the modularity score. The negative contribution equals $(-P_{i,j})$. Meaning that highly expected edges contribute more negatively when they are absent than the lowly expected ones.

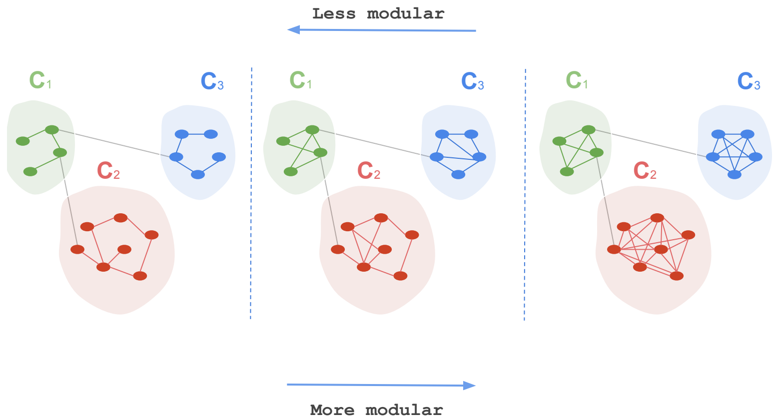
[P3] Punishing existing edges among communities. Even though it is not straightforward to see that in the equation, clusterings with less cross-community edges score higher modularity than those with more cross-community edges. The reason is that modularity counts only the contribution of edges lying within communities. The existence of edges among two communities will result on a larger number of edges in the network (larger $2|E|$ in the equation) and zero contribution to the modularity score. This will result on a lower modularity than if edges did not exist across communities.

[P4] Rewarding non-existing edges among communities. Following the same reasoning discussed in P3, non-existent edges among communities means smaller $|E|$ which results in a higher modularity.

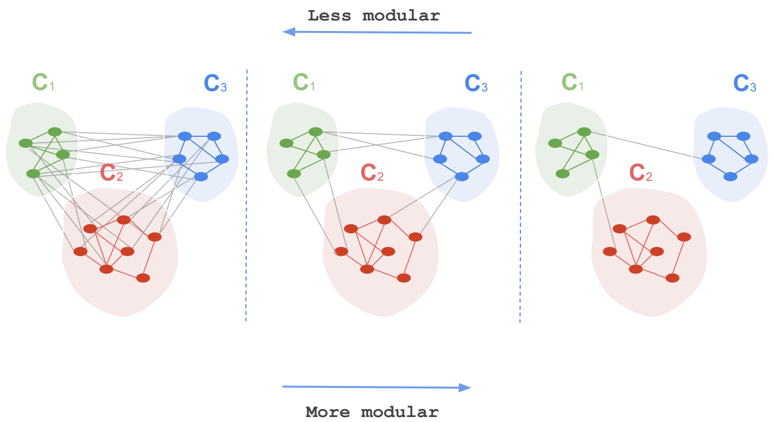
Figure 2 illustrates the effect of within-community and cross-community edges on the modularity score.

2 From Mono-layer Modularity to Generalized Modularity

Authors in[14] proposed a generalization of modularity so it applies to multi-layer networks. In this generalisation, the authors propose a new parameter ω to the modularity equation. ω, also called the coupling strength, is a weight assigned to the coupling edges connecting nodes of an actor across different layers. On the layer level, it has been interpreted as the closeness among different layers[16]. On the actor level, it has been claimed that this parameter should reflect information about the extent to which an actor tends to have the same behavior across layers (high ω) or a different behaviour in different layers (low ω)[15]. With this introduction of ω to the generalisation of mono-layer modularity, the modularity score does not only reward existent within-community intra-layer edges and absent cross-community intra-layer edges, and penalise the absent within-community intra-layer edges and existent cross-community intra-layer edges in each layer, but also rewards the coupling edges (i.e inter-layer edges) within communities with an amount proportional to ω. This means that if two nodes n_{ix}, n_{iy} which refer to the same actor i, and hence are coupled, happen to appear in the same community, this contributes to the multi-layer modularity score with an amount proportional to ω_{xy}, that is the coupling strength assigned to the coupling edge between n_{ix}, n_{iy}. With that being said, the result of maximising modularity in multi-layer network is not necessarily a clustering that groups all the nodes of an actor in one community. There are two forces that drive the partitioning in multi-layer networks using modularity maximisation. The first tries to keep the node in its optimal mono-layer modularity grouping, and the second tries to group the node together with other nodes that refer to the same actor. To have an idea about modular structures in multi-layer networks according to multi-layer modularity, we report the multi-layer modularity scores (Figure 3) of three different clusterings over a toy multiplex network constituted of 3 layers and 15 actors and three cliques in each layer. The figure shows that even though the three clusterings equally optimise the distribution of intra-layer edges within and across the communities, the generalised modularity favors clusterings that maximise the coupling edges within communities in addition (Figure 3.a).



(a) P1 and P2



(b) P3 and P4

Figure 2. The effect of within-community and cross-community edges on the modularity score of the clustering $\mathscr{C} = \{C_1, C_2, C_3\}$ over the mono-layer network illustrated in figures (a) and (b)

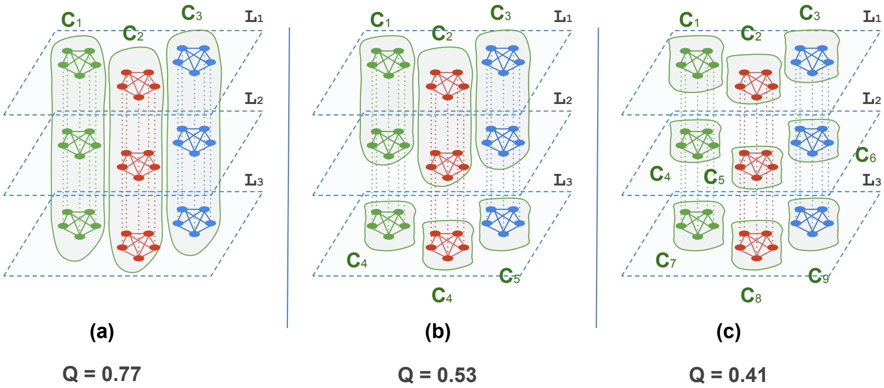


Figure 3. Modularity scores, calculated using multi-layer modularity, for three different clusterings over the same toy multiplex network constituted of 3 layers and 15 actors and edges that are distributed across three cliques per layer

The generalization form of equation 1 as proposed by[14] can be written as :

$$Q = \frac{1}{2\mu}\underbrace{\Big(\sum_{C\in\mathscr{C}}\sum_{(i_s,j_s)\in C}\Big[A^s_{(i,j)} - P_{(i,j)}\Big]\Big)}_{\text{part 1}} + \frac{1}{2\mu}\underbrace{\Big(\sum_{C\in\mathscr{C}}\sum_{(i_s,j_r)\in C}\Big[W^{s,r}_{i,j} * \delta(i_s,j_r)\Big]\Big)}_{\text{part 2}} \tag{2}$$

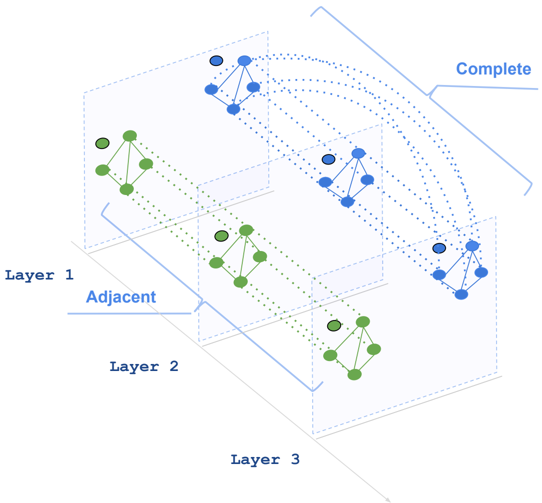
Where $A^s_{i,j}$ is the adjacency matrix of layer s, $W^{s,r}$ is the coupling matrix that describes inter-layer edges between layers s, r [1] and $W^{s,r}_{i,j}$ therefore equals the value of the coupling strength $\omega_{i,j}$ assigned to the coupling edge connecting nodes i from layer s and j from layer r. $\delta(i_s, j_r)$ is a Kronecker delta and it equals 1 if nodes i_s and j_r refer to the same actor, else 0. μ is a normalisation factor.

The first part of equation 2 is the same used to calculate mono-layer modularity in equation 1. This part alone reaches its maximum when nodes in each layer are grouped according to their optimal mono-layer modularity. The second part of this equation is the added multi-layer ingredient to the modularity score. This part alone is maximised when the mono-layer optimised groupings are cross-merged across the layers such that all the overlapping mono-layer groupings appear together in the same multi-layer community. The main difference between the two parts is that the first part penalises any other grouping of the nodes that does not respect the optimal mono-layer modularity grouping, while the second part (assuming $\omega > 0$) does not penalise but only rewards the co-existence of the same actor nodes in one community. If the contribution of the second part of equation 2 is small (i.e ω is small), optimising modularity will keep nodes in each layer grouped according to their optimal mono-layer modularities so the first part of equation 2 is optimised to its maximum value. At the same time if the contribution added by the second part is big enough (i.e ω is large), this might not guarantee that nodes in each layer will be grouped according to their optimal mono-layer modularity as the contribution of the coupling edges added to the total sum might become big enough to compensate for the penalties resulted by grouping nodes not according to their optimal mono-layer modularity.

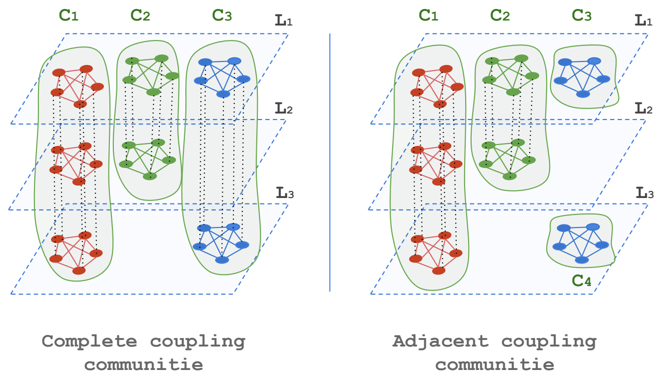
The type of coupling chosen to connect nodes of the same actor across layers is of great importance for studying any dynamic in multi-layer networks especially for community detection using modularity maximisation. When layers are *ordinal*, i.e they have a specific order like in the case of time-dependent networks, coupling edges connect each layer s with the successive layer s+1 and the previous layer s-1. When layers are *categorical*, i.e they do not follow a specific order like the case of multiplex networks, coupling edges connect all pairs of layers. We refer to these two types of couplings as adjacent coupling, and complete coupling for time-dependent and multiplex networks respectively. Figure 4 illustrates the two different types of coupling strategies and the effect of that on a process like community detection in multi-layer networks.

The original authors where the generalised modularity was firstly proposed did not discuss much about ω. In their paper, coupling edges where uniformly assigned to values greater then or equal to 0. In[16] authors suggested that coupling edges can convey more information and hence, they do not necessary have to be uniformly assigned across layers. They can express the

[1] We use W instead of C , as mentioned in its original paper, to refer to the coupling matrix not to confuse it with our notation of a community which we chose to be C



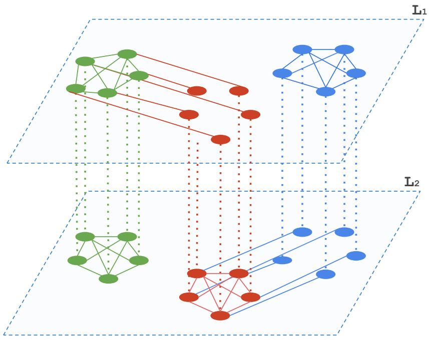
(a) The two styles of coupling nodes in multi-layer networks



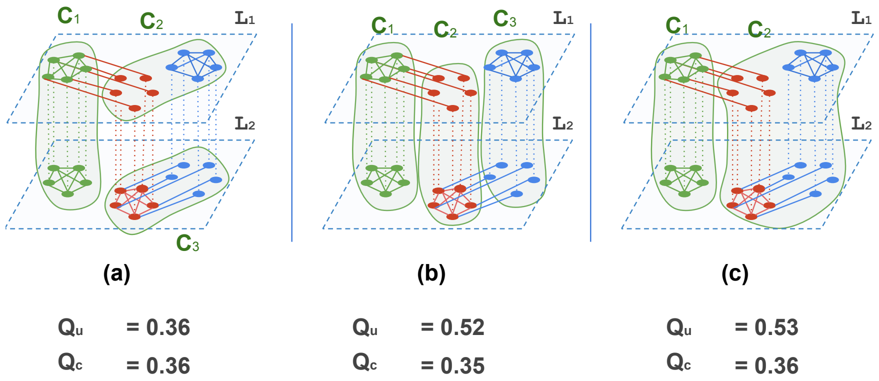
(b) A possible effect of the coupling strategy on the output of community detection

Figure 4. Styles of coupling nodes in multi-layer networks

closeness among layers and hence closer layers are assigned larger values of ω and larger otherwise, and absent couplings among two layers should be expressed as couplings with a negative coupling strength ($\omega < 0$). This allows the second part of equation 2 to penalise the co-existence of two nodes that refer to the same actor if the nodes are in two un-coupled layers. In[15], the authors argued that couplings should be looked at from even a lower level, that is, the actor-level so nodes of an actor with similar neighbourhood over the layers should be maximally coupled (i.e assigned a large ω), while those with different neighborhoods across the layers should be minimally coupled. Figure 5b, for example, reports the multi-layer modularity scores using uniform coupling Q_u with $\omega = 1$ versus those using customised coupling Q_c for three different clusterings over the network illustrated in Figure 5a. The figure shows that assigning uniform coupling strength to all couplings might lead to favoring communities that expand over multiple layers even if actors nodes have different neighborhoods across the layers (Q_u is higher in cases (b) and (c)).



(a) A toy example of a multiplex network where actors change neighborhood over layers



(b) Modularity scores using customized coupling strength Q_c and uniform coupling strength Q_u for three different clusterings over the network illustrated in figure 5a.

Figure 5. Effect of using customised coupling strength on the modularity scores.

3 The Patterns Identified by Maximizing Multi-Layer Modularity

In this section, we will refer to the clustering of a single layer nodes resulted by maximising the mono-layer modularity on that layer as the optimal mono-layer groups, and to the clustering resulted by maximising the generalised modularity on the whole multi-layer network as multi-layer communities. We consider two groups of nodes in two different layers overlapping, if there exist at least two nodes (one in each group) that refer to the same actor. When two different groups of nodes in two different layers merge together, we refer to that as cross-merging, to differentiate it from the within-layer merging possible to happen between two gropes of nodes belonging to the same layer. As to the two parts in equation 2, we refer to the first part as the *intra-layer gain*, and to the second part as the *coupling gain*.

A multi-layer community detection task that maximizes multi-layer modularity involves partitioning the multi-layer network nodes into a clustering $\mathscr{C}$ that maximizes both parts of equation 2, as shown before. The intra-layer gain pushes nodes towards being partitioned into their optimal mono-layer groups. The coupling gain pushes the overlapping optimal mono-layer groups belonging to different layers towards cross-merging so to constitute multi-layer communities. The patterns identified by maximising both parts together largely depend on the value given to ω which seems to play the role of cross-merging orchestrator in the maximisation process (assuming that nodes are scanned in each layer separately first then across layers). Remember that for any solution not to partition the nodes according to their optimal mono-layer groups, this results in a punishment in the intra-layer gain in equation 2. However, if this punishment can be compensated by the coupling gain, this gives some freedom (even if limited) to produce clustering that does not firmly respect the mono-layer optimal modularity constraints.

Let us assume that community labeling by multi-layer modularity maximisation happens in an order such that it scans the network nodes as follows. First, it scans pairs of nodes belonging to the same layer to maximise only the intra-layer gain, so nodes are labeled according to their optimal mono-layer groups, and then it scans pairs of nodes belonging to different layers so their community labels can be updated such that the coupling gain is maximised. During the re-labeling phase, a node might face two types of relabeling: **(i)** free re-labeling, that is the one that results only on a larger coupling gain without affecting the intra-layer gain. The result of this is cross-merging the overlapping optimal mono-layer groups across layers without altering their mono-layer grouping (i.e without having to put two nodes that belong to two different mono-layer groups in one layer together in the same multi-layer community), and we will refer to that as *free cross-merging*. **(ii)** expensive re-labeling, that is when adopting the new label requires the node to give up its optimal mono-layer grouping and having to co-exist with nodes that will punish the intra-layer gain. The result of this is cross-merging overlapping groups across layers and altering their mono-layer grouping, and we will refer to that as *expensive cross-merging*. Figure 6 illustrates the difference between free cross-merges and expensive cross-merges.

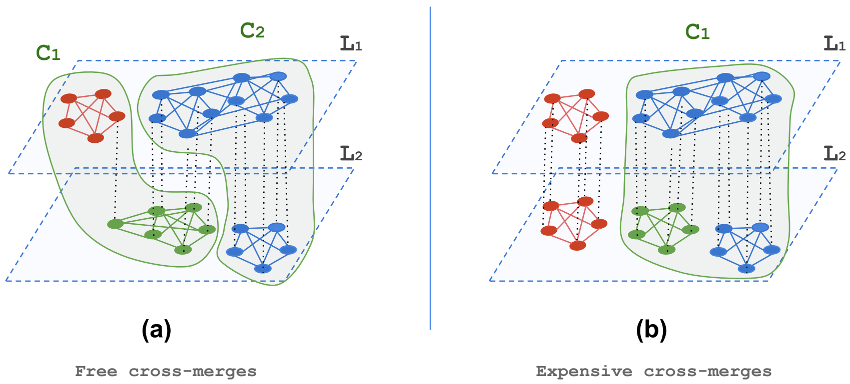
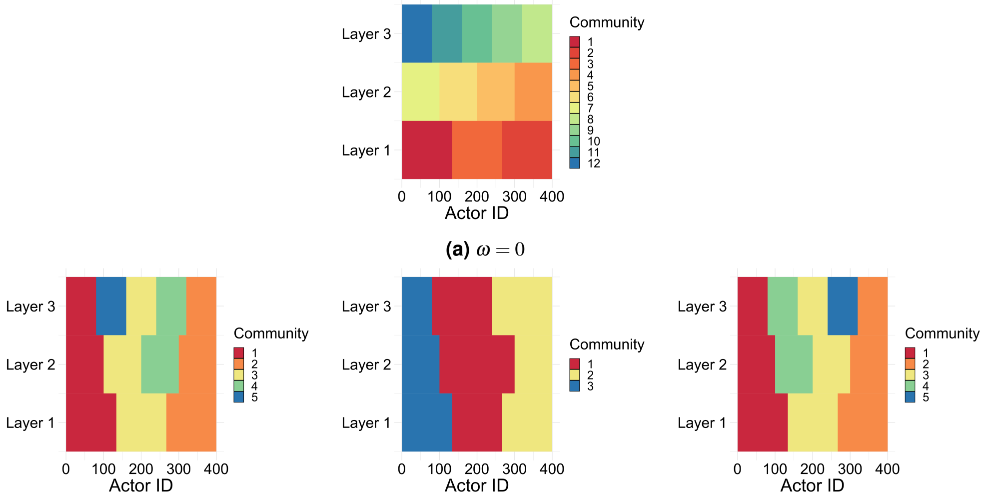


Figure 6. Example of free cross-merges, (a), which does not alter the optimal mono-layer grouping specified by the colors of the nodes in each layer, and expensive cross merges, (b), which alters the optimal mono-layer grouping (i.e it forces the blue and green nodes in layer L_2 to appear together in one community). Nodes in each layer are colored according to their optimal mono-layer grouping.

Assuming a multi-layer network constituted of modular layers, assigning the smallest non-zero value to ω uniformly across layers results in performing all the possible free cross-merges across these layers such that the maximise the coupling gain. That is because the free cross-merges results in a positive coupling-gain which can be too small to compensate for any punishment resulted by altering the optimal mono-layer groupings. Hence, the resulted multi-layer communities are those where nodes in each layer are densely connected within the community and sparsely connected with the rest of the network. The larger ω gets, the larger the coupling gain can be till it gets to a point when it becomes large enough to compensate for some intra-layer penalties, and this gives space to perform expensive cross-merges. The result is multi-layer communities where not all nodes necessarily fall into their optimal mono-layer groups, but at least in one layer they do. Figure 7 provides a colored illustration of the grouping of nodes in a 3-layer 400-actor multiplex network for different assignments of ω. The value $\omega = 0$ shows the implanted grouping assigned to the nodes in each layer where each different color refers to a different group label. The multi-layer community structure resulted by maximising modularity is shown for $\omega = 0.0001$, $\omega = 1$ and when ω is customised based on the common neighbours across layers. As shown in the figures, low ω resulted in multi-layer communities constituted



(a) $\omega = 0$

(b) $\omega = 0.0001$ **(c)** $\omega = 1$ **(d)** ω is customised

Figure 7. A colored representation of the community assignment resulted by maximizing the generalised modularity as a function of ω in a 3-layer 400-actor multiplex network. The x axes represent the actor id, while the y axes represent the layer, and different colors represent different community assignments

of free cross-merges among optimal mono-layer groups, while larger ω resulted on expensive cross-merges.

Understanding the effect of ω on the resulted multi-layer community structure is very important for the interpreteablity of thees communities. In general, when the coupling style used to couple nodes across layers is adjacent or complete, low ω results on multi-layer communities where nodes in each layer are in their optimal mono-layer grouping (i.e densely connected with each other in their layer and sparsely connected with the rest of the network), and high ω results on multi-layer communities where nodes fall in their optimal mono-layer groups at least in one layer but not necessarily all of them. When the optimal mono-layer groups are not aligned across layers, there is less to claim about these multi-layer communities on the actor level. Think of the clustering illustrated in figure 6.a which is resulted by maximising modularity on that network for any positive assignment of $\omega > 0$. For community C_1, these is a set of actors that are grouped together in one community only because they are constituted of two groups with one common actor in both. Another example is community C_1 in Figure 6.b which is resulted by maximising modularity on that network with $\omega = 1$. That community is constituted of a group of actors that densely interacted with each other in one layer, but were divided into two groups on another. In a time-dependent network where layers refer to different time-windows, interpreting such multi-layer community might be problematic.

4 Models for Communities in Multiplex and Time-dependent Networks

In this section, we propose different models for disjoint multi-layer communities possible to evolve in multiplex and time-dependent networks under different conditions. We claim that the five models we propose provide a good coverage for the types of patterns (i.e multi-layer communities) possible to evolve in multiplex and time-dependent networks. Later, we use these models to test the ability of modularity maximisation methods on recovering different patterns in multi-layer networks. In this section, we refer to the nodes of an actor a in different layers with a non-zero degree in their layers as the *active nodes* of a. For a multi-layer community C that expands over multiple layers, we refer to the set of nodes of C in a single layer l as the induced nodes of C in l, and to the set of actors resulted by mapping each of the induced nodes of C in a layer l to their actors as the induced actors of C in l. These models are:

[M1] Pillar communities: We call a multi-layer community C a pillar community if there exists a set of actors $\mathscr{A} = \{a_1, a_2, ...a_k\}$ such that C is constituted of only the active nodes of all the actors in $\mathscr{A}$ in all layers of the multi-layer network. Pillar communities result from a very high dependency among actors connectivity patterns on all layers of the network, which results in an aligned grouping of the relevant nodes in each layer.

[M2] Semi-pillar layer-adjacent communities: We call a multi-layer community C a semi-pillar layer-adjacent community if there exists a set of actors $\mathscr{A} = \{a_1, a_2, ...a_k\}$ such that C is constituted of only the active nodes of all the actors in $\mathscr{A}$ in a subset of layers of the multi-layer network and these layers are adjacent to each other. Semi-pillar layer-adjacent communities usually evolve in time-dependent networks where layers refer to specific time-windows. In these networks, a set of actors might engage in the same community for a limited time and then dissolve or engage in other groups in subsequent time-windows. This might result in semi-pillar communities that expand over an subset of the layers that are adjacent to each other.

[M3] Semi-pillar non-layer-adjacent communities: Similar to [M2] except that the layers where the community expands are not adjacent to each other. These communities might evolve in multiplex networks where layers do not necessarily have an order and the layers where the semi-pillar communities expand are not adjacent to each other. These communities also might evolve in time-dependent networks if a group of actors engage in a community for a couple of consecutive time-windows then dissolve at some windows, then engage again in the same community.

[M4] Partially overlapping communities: A multi-layer community C that expands over multiple layers is partially overlapping if the sets of the induced actors of C in each layer where the community expands partially (but not completely) overlap. These communities evolve in cases when the community membership of a set of actors in one layer l_1 influences the community membership of only a subset of these actors in another layer l_2 while the membership of the rest of these actors in l_2 does not necessarily depend on their membership in l_1. Think of an example where the network is a three-layer multiplex network modelling twitter interactions (following, retweeting and replying) among a set of actors. It might be the case that the community membership of a set of actors in the 'following' layer influences the community membership of only a subset of these actors in the 'retweet' or 'reply' layers (i.e user a_1 retweets user a_2 because they follow each other) while the community membership of the rest of these actors on these layers does not really depend on the 'following' layer.

[M5] Hierarchical communities: A multi-layer community C that expands over multiple layers is hierarchical if there is a hierarchy among the sets of the induced actors of C in the layers where it expands. A hierarchical community evolves when the grouping of a set of actors in a layer l_1 still depends on the community membership of these actors in another layer l_2 but additional non-layer specific dependencies might result on different divisions of this grouping across the layers which breaks the perfect cross-layer group alignment that happens in the pillar model. Think of the 3-layer multiplex modelling Twitter interactions mentioned above. A grouping of a set of actors in the 'retweeting' layer might still depend on whether they follow each other or not (i.e user a_1 retweets user a_2 only if they follow each other), but some other user-specific properties (political affiliation for example) might result on multiple divisions of their groupings across the two layers.

5 Results

From a multi-layer modularity perspective, inter-layer coupling edges are not perceived as identity edges but edges that couple nodes of the same community

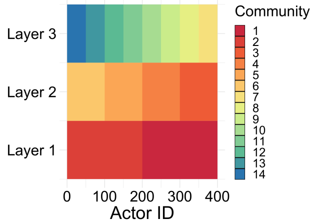
Together with the adjacent and the complete coupling styles usually used in the literature to couple nodes in multiplex and time-dependent networks, we tested the accuracy of modularity maximization with an additional coupling style, we refer to as community-based style, where we coupled nodes across layers only if they belong to the same multi-layer community which resulted in a non-uniform coupling. Our experiments show that the highest accuracy of community discovery is always achieved when the coupling style is community-based which suggest that multi-layer modularity maximization treats inter-layer coupling edges as community edges and when coupling-edges are used as identity-edges, the accuracy of modularity maximization seems to drop down.

Less accurate does not necessarily imply less modular according to multi-layer modularity

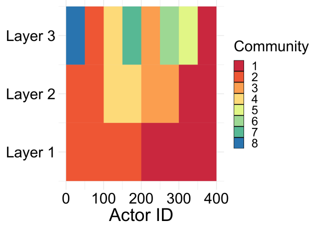
While the accuracy scores fluctuate as a function of both the coupling style and the value of the coupling strength (figure 9), modularity scores seem to be stable. This suggests that less accurate clusterings are not necessarily less modular.

Pillar communities are accurately recoverable under all coupling styles and a very small positive assignment of ω

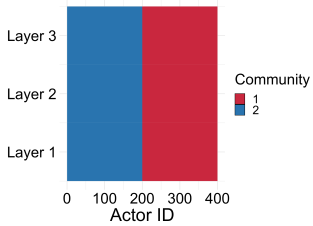
As shown in figure 9a, a very small positive assignment of ω ($\omega = 10^{-13}$) set uniformly across layers guarantees a very high accuracy for recovering pillar communities independently from the coupling styles (complete, adjacent, or community based) used to couple nodes across layers.



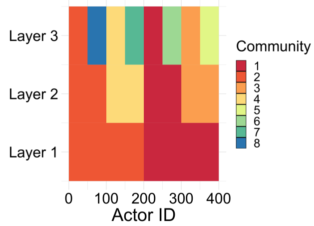
(a) $\omega = 0$



(b) $\omega = 0.0001$



(c) $\omega = 1$



(d) ω is customised

Figure 8. A colored representation of the community assignment resulted by maximizing the generalised modularity as a function of ω in a 3-layer 400-actor multiplex network. The x axes represent the actor id, while the y axes represents the layer, and different colors represent different community assignments

Semi-pillar communities and partially overlapping communities cannot be accurately recovered using adjacent or complete coupling styles and a uniform positive assignment of ω

As the community model gets more complex compared to the pillar model, modularity maximization seems to fail at accurately recovering the ground truth communities when the coupling style is the adjacent or the complete one independently from the value given to the coupling strength ω (Figures 9b, 9c, 9d). However, if coupling edges are placed only among nodes belonging to the same multi-layer community, multi-layer modularity maximization seems to be able to recover the ground truth communities and shows a stable behavior with respect to the value given to ω.

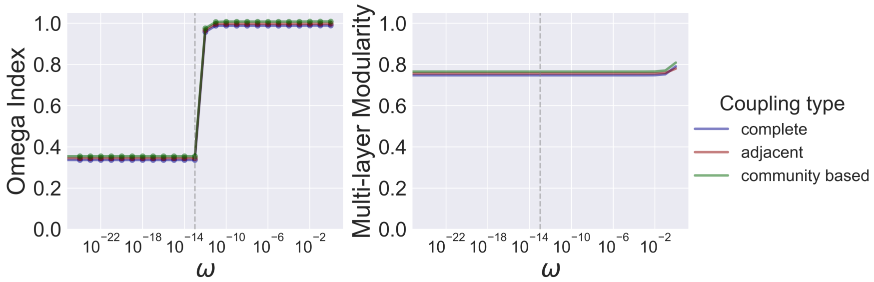
With hierarchical communities, the value ω controls the granularity of community detection

With hierarchical community model (M5) it is not clear what should be the ground truth. Hence we study the structure of the detected communities, instead of their accuracy, as a function of ω. Figures 8b, 8c, 8d illustrates the structure of the detected multi-layer communities using modularity maximization. We show only the results when the coupling type is complete since the other types do not significantly change the output in this model. We can see that using high values of ω forces the community membership of the highest level in the hierarchy across the layers. Nonetheless, using lower values of ω does not force the community membership of the highest level of the hierarchy across all the nodes across the layers but only part of them such that the mono-layer optimal groupings are guaranteed.

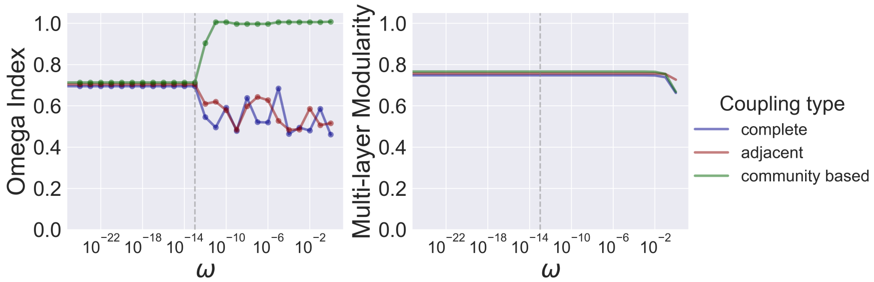
6 Discussion

In this article, we investigated multi-layer modularity maximization, the extensively used tool for community discovery in multi-layer networks. We infer unspoken assumptions that condition the ability of multi-layer modularity maximization to recover ground truth communities by investigating the role of ω and coupling edges introduced in multi-layer modularity. Our main findings can be summarized as follows: ***(1)*** the high accuracy of multi-layer modularity maximization is conditioned by a coupling style that couples only nodes of the same community together, ***(2)*** when a proper coupling is chosen in a multi-layer network generated using one of the models (M1-M4), the value of ω assigned to the coupling edges has no impact on the accuracy of community detection using multi-layer modularity maximization (as long as $\omega > 0$) , ***(3)*** less accurate does not necessarily mean less modular according to multi-layer modularity, and ***(4)*** with hierarchical multi-layer communities, ω controls the granularity of the detected patterns and the qualitative interpretation of the identified patterns is not always clear.

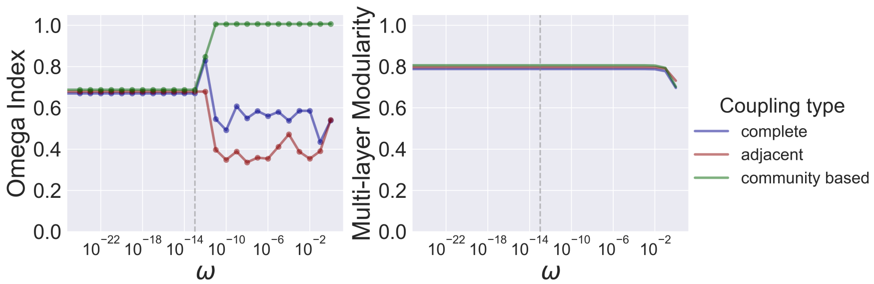
The fact that the accuracy of multi-layer modularity maximization with more complex multi-layer community models is conditioned by coupling nodes of the same community together (rather than the same actor) sheds a light on the importance of



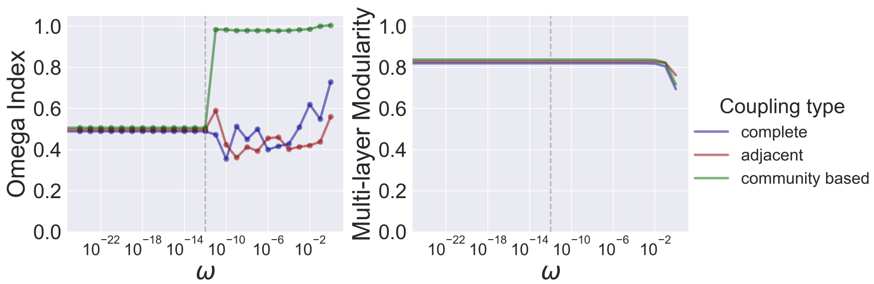
(a) Pillar communities



(b) Semi-Pillar layer-adjacent communities



(c) Semi-Pillar non-layer-adjacent communities



(d) Partially overlapping communities in a multiplex network

Figure 9. Accuracy using omega-index and modularity scores of community detection using multi-layer modularity maximization as a function of the coupling type and the value of the coupling strength

coupling style for a successful recovery of the communities and hints to limitations in the coupling strategies available in the literature to couple nodes in multi-layer networks. While some heuristics can be developed to predict whether two nodes of an actor belong to the same community or not, and a community-based coupling can be built on top of that, a valid question can be: why should the accurate recovery of multi-layer communities be conditioned by knowing which nodes of the same actor fall in the same community, which is what the community detection method itself is supposed to infer?

The reason why networks should be analyzed in a multi-layer manner is that there are dependencies among different interactions/relationships and providing any analysis by considering only one of these interactions might give misleading or incomplete findings. This means that for the formation of multi-layer communities, different dependencies among the layers might result on different structures of multi-layer communities (a very high dependency among all layers for example results in pillar communities). While this motivation is sound, the question becomes whether it is always possible to reflect these inter-dependencies in the tools used for multi-layer network analysis, especially community detection. Thinking of multi-layer modularity, the coupling strength ω (as suggested by the original authors[14]) is a binary variable that takes one of the two values: either 0 to refer to the non-existence of a coupling edge between two nodes of an actor, or a positive value $\omega > 0$ to refer to the existence of that coupling edge. When the proper coupling is provided, our experiments showed that the positive value assigned to the existent couplings does not affect the accuracy of community detection. This means that this parameter with models (M1-M4) has no role more than referring to existent ($\omega > 0$) or non-existent ($\omega <= 0$) couplings. This adds another limitation to modularity maximization based methods because with this limited interpretation of ω, the only multi-layer ingredient in the generalized modularity, there is no way to reflect different levels of inter-dependencies among the different layers and/or nodes of an actor and to take these inter-dependencies into account in community discovery.

As shown in our experiments, multi-layer modularity scores do not necessarily follow the same trends the accuracy of the recovered communities follow. Indeed, the modularity scores of the accurately recovered clusterings are not necessarily higher than those with lower accuracy. This raises another important question that we invite more research to investigate on, is multi-layer modularity maximization a valid proxy for finding complex multi-layer communities? Or is the idea of it being as such mostly inherited from the success and the popularity the mono-layer implementation had before?

The qualitative interpretation of the recovered patterns in networks generated using one the community models (M1-M4) does not change much from those identified using the mono-layer modularity. If edges were created such that nodes of the same community are densely connected within the community and sparsely connected with the rest of the network, the patterns identified by maximizing multi-layer modularity satisfy that condition. With the hierarchical community model, however, it is not clear how to interpret the identified patterns for values of ω other than 0 (the lowest granularity of the hierarchy), and large ω (the highest level of the hierarchy). The question is whether using multi-layer community detection in this case provides any additional information the single-layer community detection for each layer separately cannot provide and we tend to believe that multi-layer community detection here using multi-layer modularity maximization can give misleading results from a qualitative perspective.

7 Methods

For recovering the multi-layer communities that maximise multi-layer modularity, we chose Generalised Louvain method[17] as a representative method for the class of modularity maximisation community detection methods. The choice of this method comes for both being a well-referenced method in the literature and serving as one of the best approximation methods among other modularity-maximisation methods in terms of its accuracy and performance. Since communities resulted by this method might vary from one execution to another because the order by which the nodes are scanned by this method is chosen at random, we provide the final result of community detection after 10 executions and choosing the one with the maximum modularity as an output.

For each of the community models (M1-M4) we calculate the accuracy of the resulted communities as a function of the coupling style (adjacent, complete, and community based) and the value of the coupling strength ω assigned to the coupling edges. between the dependency p among layers chosen. For measuring accuracy, we chose omega-index for its sensitivity to different types of dissimilarities among clusterings and its ability to remove the by-chance agreement from the final score as we discussed in a previous study[18].

As regards the generation of the multi-layer networks, we refer to the generative model in[9]. To the best of our knowledge, this provides the most general platform in the literature for generating synthetic multi-layer communities that takes into consideration different types of multi-layer networks. The generation of a multi-layer community using this model goes through the following three steps. First, assuming a multi-layer network of n actors and m layers and initial number of communities k to be planted in each layer, the model starts by assigning nodes in each layer randomly to k community memberships (a categorical distribution can be used for this step). At the end of this phase, nodes in each layer will be distributed over k not necessarily equal in size groups. In the second phase, nodes start to propagate their community memberships across layers with a probability equals to the dependency between the layers. The type of multi-layer network we want to generate, i.e multiplex or time-dependent, controls the order and the way this propagation of community labels across layers happen. At the end of this phase, community memberships that were set in the first phase might be updated based on the assumed dependency p set among layers. At the third phase, a multi-layer edge generation model can be used to create edges of the multi-layer network given the implanted community assignments resulted from the previous step. In our experiments, we generate out toy examples using a variant of the degree-corrected SBM benchmark that avoid the creation of self-loops and parallel edges[9]. We chose to fix the mixing parameter in edge generation phase in our experiments to a very small value ($\mu = 0.05$). That is because our main goal from the experiments is not to test the ability of multi-layer modularity maximisation to recover noisy patterns, but to test the ability to recover complex multi-layer patterns.

We chose our multi-layer toy networks to be 4-layer networks of 1000 actors (i.e 4000 nodes). To generate pillar communities (M1), we use the aforementioned generative model and we generate a temporal network with a very high dependency across the layers (p=1). To generate the semi-pillar layer-adjacent communities (M2), we generate two sets of pillar partitions on two 2-layer temporal networks. We use the resulted partitions to generate a 4-layer multi-layer network where in the first two layers we plant the first set of partitions and the last two layers we plant the second set of partitions. For (M3), we do the same except that the 4-layer multi-layer network is resulted by implanting the first set of partitions in the first and the third layer, while in the second and the fourth we plant the second set of partitions. As to partially overlapping communities, we generate multiplex networks with a moderate dependency among layers (p = 0.3). This, according to the used generative model, results in communities that are not perfectly pillar but partially overlapping.

References

1. Papadopoulos, S., Kompatsiaris, Y., Vakali, A. & Spyridonos, P. Community detection in social media. *Data Min. Knowl. Discov.* DOI: 10.1007/s10618-011-0224-z (2012).
2. Hanteer, O. & Rossi, L. An innovative way to model twitter topic-driven interactions using multiplex networks. *Front. Big Data* DOI: 10.3389/fdata.2019.00009 (2019).
3. Rosvall, M., Esquivel, A., Lancichinetti, A., West, J. & Lambiotte, R. Memory in network flows and its effects on spreading. *Nat. Commun.* DOI: 10.1038/ncomms5630 (2014).
4. Gargi, U., Lu, W., S. Mirrokni, V. & Yoon, S. Large-scale community detection on youtube for topic discovery and exploration. (2011).
5. Fortunato, S. & Barthélemy, M. Resolution limit in community detection. *Proc. Natl. Acad. Sci.* DOI: 10.1073/pnas.0605965104 (2007).
6. Newman, M. E. J. & Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* DOI: 10.1103/PhysRevE.69.026113 (2004).
7. Newman, M. E. J. Modularity and community structure in networks. *Proc. Natl. Acad. Sci.* DOI: 10.1073/pnas.0601602103 (2006).
8. Dickison, M. E., Magnani, M. & Rossi, L. *Multilayer Social Networks* (Cambridge University Press, 2016).
9. Bazzi, M., Jeub, L. G. S., Arenas, A., Howison, S. D. & Porter, M. A. Generative benchmark models for mesoscale structures in multilayer networks. *ArXiv* **abs/1608.06196** (2016).
10. Berlingerio, M., Pinelli, F. & Calabrese, F. ABACUS: frequent pAttern mining-BAsed Community discovery in mUltidimensional networkS. *Data Min. Knowl. Discov.* 294–320 (2013). arXiv:1303.2025v2.
11. Berlingerio, M., Coscia, M. & Giannotti, F. Finding and characterizing communities in multidimensional networks. In *International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 490–494 (IEEE Computer Society Washington, DC, USA, 2011).
12. Boutemine, O. & Bouguessa, M. Mining Community Structures in Multidimensional Networks. *ACM Transactions on Knowl. Discov. from Data* **11**, 1–36, DOI: 10.1145/3080574 (2017).
13. Afsarmanesh, N. & Magnani, M. Finding overlapping communities in multiplex networks. (2018).
14. Mucha, P. J., Richardson, T., Macon, K., Porter, M. A. & Onnela, J.-P. Community structure in time-dependent, multiscale, and multiplex networks. *Science* DOI: 10.1126/science.1184819 (2010).
15. Aslak, U., Rosvall, M. & Lehmann, S. Constrained information flows in temporal networks reveal intermittent communities. *Phys. Rev. E (Statistical, Nonlinear, Soft Matter Physics)* DOI: 10.1103/PhysRevE.97.062312 (2018).
16. Zhang, H., Wang, C.-D., Lai, J.-H. & Yu, P. Modularity in complex multilayer networks with multiple aspects: A static perspective. *Appl. Informatics* DOI: 10.1186/s40535-017-0035-4 (2016).

17. Lucas G. S. Jeub, I. S. J., Marya Bazzi & Mucha, P. J. A generalized louvain method for community detection implemented in matlab. https://github.com/GenLouvain/GenLouvain,http://netwiki.amath.unc.edu/GenLouvain/GenLouvain (2011-2019).

18. Hanteer, O. & Rossi, L. The meaning of dissimilar: An evaluation of various similarity quantification approaches used to evaluate community detection solutions. In *Proceedings of the 2019 IEEE/ACM International Conference onAdvances in Social Networks Analysis and MiningFAB 2019, FOSINT-SI 2019, HI-BI-BI 2019*, 513–518 (Vancouver, Canada, 2019).

Acknowledgements (not compulsory)

Acknowledgements should be brief, and should not include thanks to anonymous referees and editors, or effusive comments. Grant or contribution numbers may be acknowledged.

Author contributions statement

Must include all authors, identified by initials, for example: A.A. conceived the experiment(s), A.A. and B.A. conducted the experiment(s), C.A. and D.A. analysed the results. All authors reviewed the manuscript.

Additional information

To include, in this order: **Accession codes** (where applicable); **Competing interests** (mandatory statement).

The corresponding author is responsible for submitting a competing interests statement on behalf of all authors of the paper. This statement must be included in the submitted article file.

Appendix C

The Meaning of Dissimilar: an Evaluation of Various Similarity Quantification Approaches Used to Evaluate Community Detection Solutions

The Meaning of Dissimilar: An Evaluation of Various Similarity Quantification Approaches Used to Evaluate Community Detection Solutions

1st Obaida Hanteer
Data Science & Society Lab
IT University Of Copenhagen
Copenhagen, Denmark
obha@itu.dk

2nd Luca Rossi
Data Science & Society Lab
IT University Of Copenhagen
Copenhagen, Denmark
lucr@itu.dk

***Abstract*—Evaluating a community detection method involves measuring the extent to which the resulted solution, i.e clustering, is similar to an optimal solution, a ground truth. Different normalized similarity indices have been proposed in the literature to quantify the extent to which two clusterings are similar where 1 refers to a perfect agreement between them (i.e the two clusterings are identical) and 0 refers to a perfect disagreement. While interpreting the similarity score 1 seems to be intuitive, it does not seem to be so when the similarity score is otherwise suggesting a level of disagreement between the compared clusterings. That is because there is no universal definition of dissimilarity when it comes to comparing two clusterings. In this paper, we address this issue by first providing a taxonomy of similarity indices commonly used for evaluating community detection solutions. We then elaborate on the meaning of clusterings dissimilarity and the types of possible dissimilarities that can exist among two clusterings in the context of community detection. We perform an extensive evaluation to study the behaviour of different similarity indices as a function of the dissimilarity type with both disjoint and non-disjoint clusterings. We finally provide practitioners with some insights on which similarity indices to use for the task at hand and how to interpret their values.**

I. Introduction

Community detection is one of the core tasks in the analysis of complex networks. It involves clustering the network nodes into groups, also known as communities or clusters, based on the properties they share and/or the way they are connected together. Since members of a community tend to generally share some properties, revealing the community structure of a network helps breaking it into its smaller building blocks and provides a better understanding of the overall functioning of the network. This has many applications in social media group detection [1], second-order flow analysis in human mobility [2], and topic detection in information networks [3], just to cite a few.

ASONAM '19, August 27-30, 2019, Vancouver, Canada

ACM ISBN 978-1-4503-6868-1/19/08/
http://dx.doi.org/10.1145/3341161.3342941

A lot of effort has been devoted in the last decade to develop community detection algorithms to cope with different properties and complexities in social networks. While this resulted in a plethora of community detection methods that might differ from one another in the way they define what a community is, the ultimate goal is always the same - to produce a clustering $\mathcal{C}$ over the network nodes where $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$, and C_1, C_2, …, C_k are disjoint or non-disjoint groups of nodes. To maintain consistency in the used terms over the paper, we will refer to a community detection solution as a *clustering*, and to a single community in a clustering as a *cluster* unless mentioned otherwise. We will also use the term *dissimilarity* to refer to a disagreement between two clusterings.

With the growing number of available community detection methods, an urgent need has arisen for tools to evaluate the resulted clusterings of these methods such that we can either compare these clusterings against a reference clustering, usually referred to as the ground truth, or we compare the different clusterings resulted by different community detection methods on the same data. Multiple normalized similarity indices have been proposed or borrowed from other disciplines to quantify the level of agreement between two clusterings such that the score is 1 when the two clusterings are identical and 0 when they are perfectly dissimilar. While the meaning of similar seems to be quite intuitive when the similarity score is 1, it is unclear what is perceived as less similar or completely dissimilar according to these similarity indices and whether there is an agreement among the different indices on that. To the best of our knowledge, no previous work has tackled this question and its impact on interpreting the outputs of community detection methods. In this paper, we contribute to the literature by:

- Providing a taxonomy of the similarity indices commonly used for evaluating community detection solutions.
- Elaborating on the meaning of clustering dissimilarity and the types of possible dissimilarities that can exist among

two clusterings in the context of community detection.
- Performing an extensive evaluation to study the behaviour of the different similarity indices as a function of the dissimilarity type with both disjoint and non-disjoint clusterings.
- Providing practitioners with insights on which similarity indices to use for the task at hand, and how to interpret their values.

II. A Taxonomy of Similarity Indices Used in Evaluating Community Detection Solutions

Here we provide a taxonomy for similarity indices commonly used to quantify similarity among clusterings identified by community detection methods (Figure 1). The top-level in the proposed taxonomy specifies whether the similarity index is used for comparing disjoint clusterings, i.e clusterings where clusters do not overlap among each other, or for comparing non-disjoint clusterings, i.e clusterings where clusters might overlap. We will refer to these two types as disjoint indices and non-disjoint indices respectively.

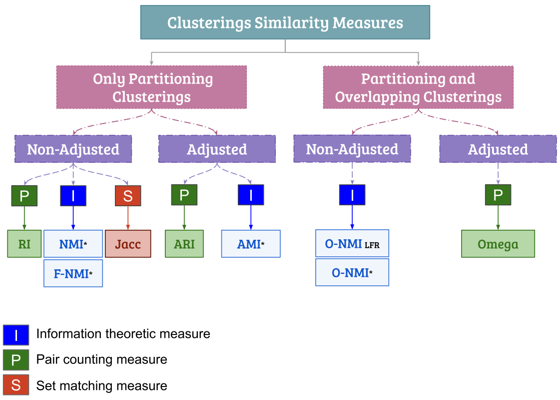


Fig. 1. A taxonomy for similarity indices commonly used to quantify similarity among clusterings identified by community detection methods. * refers to different ways of normalization (max, min, sqrt, mean, and joint)

The second level focuses on whether or not the similarity index is adjusted for the by-chance agreement. Take any two random clusterings over the same set of objects - there might always be a level of agreement between them. This agreement is usually referred to as the by-chance agreement, and adjusting indices is the practice of removing the effect of this by-chance agreement from the final score such that the score reflects only the actual similarity among two clusterings. The general formula used for adjustment is suggested by [4] to be:

$$Adjusted\_Index = \frac{Index\_Value - Expected\_Value}{Max\_Value - Expected\_Value} \quad (1)$$

Where $Index\_Value$ refers to the non-adjusted similarity score calculated by the index, Max_Value refers to the upper bound of the non-adjusted similarity score of that index, and $Expected\_index$ is a quantification for the by-chance agreement.

The third level in our taxonomy is to differentiate among three different practices used in the literature to define similarity among clusterings, namely ***(1)*** *pair counting* which calculates similarity on the level of node-pairs, i.e it considers the existence of two nodes together in the same cluster in both clusterings as an agreement and then averages the number of agreements among the two clusterings to calculate the final score, ***(2)*** *set matching* which calculates similarity on the set-level, i.e it considers clusters as sets, and it defines similarity as an average of the similarity among clusters (sets) of the first clustering with those of the second one, and ***(3)*** *information theoretic* which is borrowed from the information theory discipline. These measures treat each clustering $\mathcal{C}$ as a random variable X such that X is the vector of the nodes' community-memberships , which it then maps calculating the similarity between the two clusterings $\mathcal{C}$, $\mathcal{C}'$ into calculating the mutual information among their relevant community membership vectors I($X_{\mathcal{C}}$, $Y_{\mathcal{C}'}$).

The leaves in our taxonomy tree are the chosen similarity indices for our evaluation. Since the clustering problem is much more general than the community detection problem, we do not claim inclusiveness of all the similarity indices used to compare clusterings in general. For the sake of this paper, we chose to focus only on the similarity indices that are commonly used to evaluate community detection solutions. As regards the disjoint indices, we chose to consider Rand Index RI [5], Adjusted Rand Index ARI [4], Normalized Mutual Information NMI [6], Adjusted Mutual Information AMI [6], Fair Normalized Mutual Information F-NMI [7] which is a version of NMI that penalizes the NMI score when differences in the number of clusters exist among the compared clusterings, and Jaccard index [8]. We implemented a clustering-level Jacquard index by averaging the Jacquard coefficient of each cluster in the first clustering when compared with the cluster with the highest intersection from the second clustering. As regards the non-disjoint indices, we chose to consider O-NMI$_{LFK}$ [9], O-NMI with the different normalization options proposed by [10] (i.e, Min, Max, Mean, and Sqrt), and Omega index [11] which is an generalization of the Adjusted Rand Index to compare non-disjoint clusterings.

III. Types of Dissimilarity Among Clusterings

When it comes to comparing two disjoint clusterings $\mathcal{G}$ and $\mathcal{C}$, five types of dissimilarity are possible to exist among them. Those are: **(1)** *misplaced nodes dissimilarity*, i.e a fraction of nodes that are not placed in the right cluster in $\mathcal{C}$ according to $\mathcal{G}$, **(2)** *missing nodes dissimilarity*, i.e some nodes that do not exist in any cluster in $\mathcal{C}$ but they do in $\mathcal{G}$, **(3)** *merging dissimilarity*, that is when a cluster in $\mathcal{C}$ is constituted of some clusters from $\mathcal{G}$ that are merged together, **(4)** *splitting dissimilarity*, that is when a cluster in $\mathcal{G}$ is split into multiple clusters in $\mathcal{C}$, and **(5)** *random dissimilarity*, that is when a random combination of the aforementioned dissimilarities happen between $\mathcal{G}$ and $\mathcal{C}$. We will refer to these dissimilarities as disjoint dissimilarities.

As regards to the dissimilarity among non-disjoint clusterings (non-disjoint dissimilarities), we refer to two types of overlappings (Figure 2) mentioned in [12], namely *crisp overlapping* where the node belongs with a certainty equals 1 to a cluster, and *fuzzy overlapping* where the node belongs to a cluster with a given probability. For the sake of this paper, we chose to focus only on crisp overlapping. On a structural level, crisp overlapping can happen in various ways. It can be **(1)** hierarchical, where the overlapping within a clustering is represented by the existence of bigger clusters that are constituted of other smaller clusters - this corresponds to *hierarchical overlapping dissimilarity*, **(2)** non-hierarchical, where the overlapping within a clustering is represented by partial intersections among different clusters and/or by replicated clusters within the clustering - those correspond to *partial overlapping dissimilarity* and *replicated overlapping dissimilarity* respectively, or **(3)** mixed overlapping, where both hierarchical and non-hierarchical overlapping exist - this corresponds to *mixed overlapping dissimilarity*.

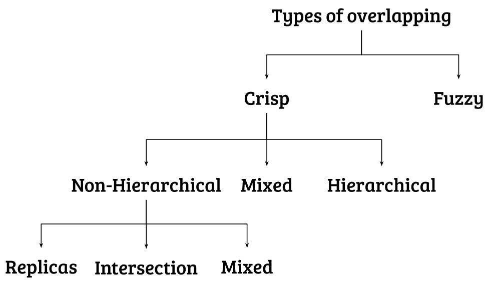


Fig. 2. Types of overlapping in non-disjoint clusterings

IV. Experiments

The main goal of our experiments is to study the behaviour of different similarity indices commonly used to evaluate community detection solutions. More specifically, we are interested in answering the following questions:

- **Q1** Do the different similarity indices have the same trends in any condition and can they, as a result, be used interchangeably to evaluate community detection solutions?
- **Q2** Are the different clustering similarity indices equally sensitive to the different types of dissimilarity between two clusterings?
- **Q3** Are the different clustering similarity indices on an agreement regarding what is considered dissimilar when comparing clusterings?
- **Q4** Given that these indices are normalized, is the relationship between the similarity score of each of these indices and the amount of dissimilarity between the compared clusterings linear?

To answer these questions, two main stages of evaluation were devised: one for comparing disjoint indices (section IV-A), and another for comparing non-disjoint indices (section IV-B). We report our observations in section IV-C.

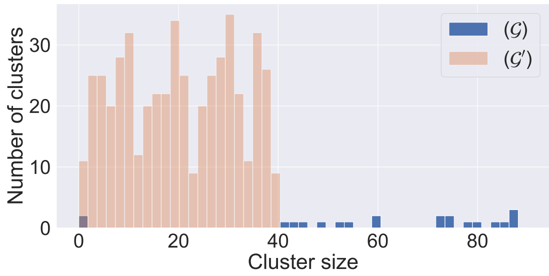


Fig. 3. Cluster size distribution for the two ground truth clusterings $\mathcal{G}$, $\mathcal{G}'$ used for our experiments

Our experiments make use of two synthetically generated ground truth clusterings $\mathcal{G}$, $\mathcal{G}'$ as roots to generate the amount of clusterings needed for each experiment. To make sure that our experiments on these indices are in the scope of community detection, the first ground truth clustering $\mathcal{G}$ is generated by detecting communities using a well known community detection method (Louvain [13]), on a representative benchmark for social networks (LFR benchmark [14]) with 10000 nodes. [1]. This resulted in a ground truth $\mathcal{G}$ clustering with 57 clusters (i.e communities). To consider also clusterings with more clusters and different cluster size distribution, we construct the second ground truth clustering $\mathcal{G}'$ by creating a random clustering over 10000 nodes. This resulted in a ground truth clustering $\mathcal{G}'$ with 493 clusters. The cluster size distribution of both $\mathcal{G}$, $\mathcal{G}'$ is illustrated in Figure 3. While it was not possible to perform our experiments on significantly larger node-sets because of the computational complexity of pair-counting indices, we think that our choices are sufficient to get useful insights and to generalize our findings about the behaviour of different indices in different conditions .

A. Comparing disjoint indices

Given the disjoint dissimilarities discussed in Section III, we construct a separate experiment for each dissimilarity type. Each of these experiments starts with comparing two identical clusterings, the ground truth clustering and a copy of it $\mathcal{C}$, then at each step i a new different disjoint clustering $\mathcal{C}_i$ is constructed and compared with the ground truth clustering. To improve the readability of the resulted figures in these experiments, we use only one type of normalization with the information theoretic indices (NMI, AMI, F-NMI) which is the MAX normalization as recommended by [15]. The disjoint clusterings for each experiment were constructed as follows:

[1] The benchmark was generated using *networkx* Python package with the following parameters: number of nodes n = 10000, power law exponent for the degree distribution = 2.6, power law exponent for the community size distribution =1.7, minimum degree = 2, maximum degree =n, minimum size of communities = 40, and maximum size of communities = n

- For the *misplaced nodes dissimilarity* experiment, a clustering $\mathcal{C}_i$ was constructed at step i by randomly choosing i nodes from the ground truth clustering and moving them from their original clusters into another randomly selected cluster.
- For the *merging dissimilarity* experiment, at each step i a new clustering $\mathcal{C}_i$ was constructed by merging two randomly selected clusters from the previous step clustering $\mathcal{C}_{i-1}$.
- As regards the *splitting dissimilarity* experiment, at each step i a new clustering $\mathcal{C}_i$ was constructed by picking a cluster from the previous step clustering $\mathcal{C}_{i-1}$ randomly and splitting it into two clusters whose sizes are also chosen at random.
- For the *missing nodes dissimilarity* experiment, at each step i a clustering $\mathcal{C}_i$ was derived from the ground truth clustering by randomly selecting i nodes and removing them from their original clusters. Since the information theoretic indices do not account for missing nodes by definition and they require the two input clusterings node-sets to be of the same size, we assign a unique community membership for each of the missing nodes instead of totally removing the node. As a result, each missing node results in a singleton cluster (i.e, a cluster with only one node in it)
- For the *random disjoint dissimilarity*, at each step i, a clustering $\mathcal{C}_i$ is constructed by applying a sequence of randomly selected disjoint dissimilarities from the aforementioned ones. The number of disjoint dissimilarities per step is chosen at random (up to 20). The goal of this experiment is to test if disjoint indices have similar trends in general. The scores of all the indices in this experiment are ordered based on the ascending order of the AMI scores.

Figure 4 reports the effect of misplaced nodes, missing nodes, merging, splitting, and random disjoint dissimilarities respectively on the disjoint indices. $\mathcal{G}$ in the figure refers to the small ground truth clustering (57 clusters) resulted by detecting communities on a LFR benchmark, and $\mathcal{G}'$ refers to the big ground truth clustering (493 clusters) resulted by a random clustering of the node-set. Note that both $\mathcal{G}$, $\mathcal{G}'$ are on the same node-set (10000 nodes).

B. Non-Disjoint clusterings

To consider the different non-disjoint dissimilarities mentioned in section III, we devise four different experiments to study the behaviour of non-disjoint indices in each case. For these experiments, we create non-disjoint ground truth clusterings $\mathcal{G}_o$, $\mathcal{G}'_o$ starting from the disjoint ground truth clusterings $\mathcal{G}, \mathcal{G}'$ used in the previous experiments by randomly creating overlappings among the clusters. Each experiment starts with comparing the ground truth clustering with an identical clustering $\mathcal{C}$, then at each step i a new non-disjoint clustering $\mathcal{C}_i$ is constructed by applying a non-disjoint dissimilarity. The resulted clustering at each step is compared with the

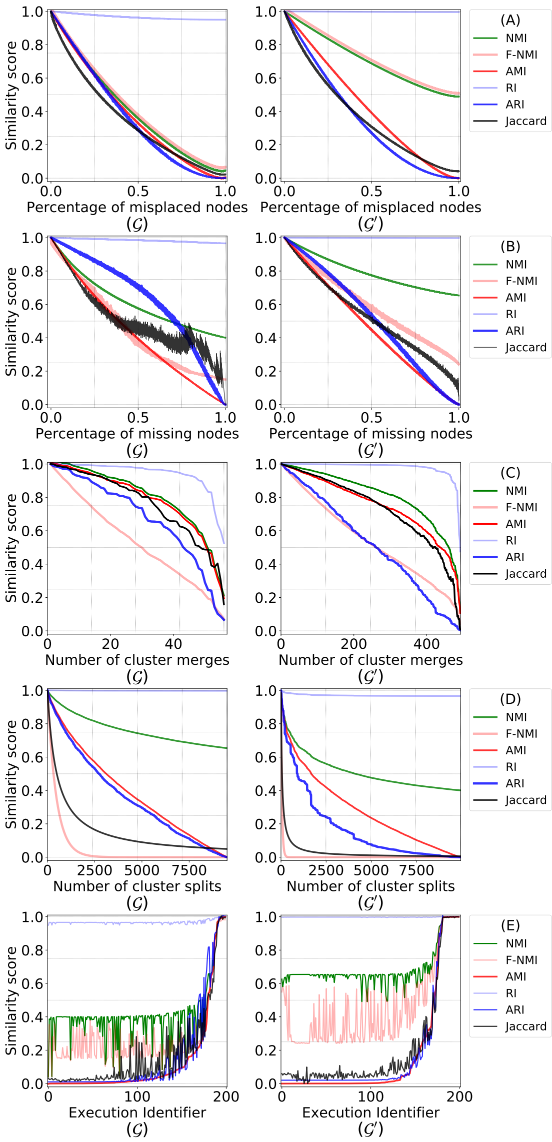


Fig. 4. Effect of misplaced nodes dissimilarity **(A)**, missing nodes dissimilarity **(B)**, merging dissimilarity **(C)**, splitting dissimilarity **(D)**, and random disjoint dissimilarity **(E)** on the disjoint indices. Each experiment is executed twice for two 10000-node ground truth disjoint clusterings different in terms of the number of clusters constituting them, a small ground truth clustering ($\mathcal{G}$ constituted of 57 clusters), and a big one ($\mathcal{G}'$ constituted of 493 clusters).

ground truth clustering. Clusterings of each experiment were created starting from the ground truth clustering as follows:

- For the *hierarchical overlapping dissimilarity* experiment, at each step i we construct a clustering $\mathcal{C}_i$ by copying the previous step clustering $\mathcal{C}_{i-1}$ and then adding a new cluster to $\mathcal{C}_i$ that is the result of merging two randomly chosen clusters from $\mathcal{C}_{i-1}$.
- For the *partial overlapping dissimilarity* experiment, we construct a non-disjoint clustering $\mathcal{C}_i$ at step i by copying $\mathcal{C}_{i-1}$ then randomly choosing two different clusters from $\mathcal{C}_{i-1}$ and creating a random intersection among them.
- For the *replicated overlapping dissimilarity* experiment, at each step i, a clustering $\mathcal{C}_i$ is generated by copying $\mathcal{C}_{i-1}$ then replicating a randomly chosen cluster from $\mathcal{C}_{i-1}$.
- For the *mixed overlapping dissimilarity* experiment, at each step i a clustering $\mathcal{C}_i$ is generated by applying a sequence of randomly selected non-disjoint dissimilarities on the ground truth clustering. The number of non-disjoint dissimilarities per step is chosen at random (up to 30 with the experiments on $\mathcal{G}_o$, and up to 400 with the experiments on $\mathcal{G}'_o$)

Figure 5 reports the effect of hierarchical overlapping, partial overlapping, replicated overlapping and mixed overlapping dissimilarities respectively on non-disjoint indices. $\mathcal{G}_o$, $\mathcal{G}'_o$ in the figure refer to the non-disjoint ground truth clusterings resulted by randomly creating overlappings within $\mathcal{G}$, $\mathcal{G}'$ clusters respectively.

C. Observations

A general observation on the disjoint indices is that rand index (RI), in all cases, fails at quantifying dissimilarity among clusterings. Thus, when we refer to disjoint indices in our observations, we exclude rand index unless mentioned otherwise.

As shown in Figure 4, both the dissimilarity type and the clustering size, in terms of the number of clusters, contribute to the existence of different trends among the disjoint indices. More specifically, misplaced nodes dissimilarity reveals similar trends amongst the disjoint indices when the compared clusterings are relatively small (4-A ($\mathcal{G}$)), with AMI being more linear than other indices. However, with bigger clusterings (4-A ($\mathcal{G}'$)), both NMI and F-NMI seems to fail at accurately reflecting the amount of disagreement among the compared clusterings while ARI, AMI and Jaccard maintain their similar trends. When it comes to the missing nodes dissimilarity, figures 4-B ($\mathcal{G}$), 4-B ($\mathcal{G}'$) show that NMI is the least sensitive to this type of dissimilarity with both small and big clusterings. While ARI, F-NMI and Jaccard indices seem to be more sensitive, AMI shows a more intuitive linear behaviour with respect to the number of missing nodes.

As regards to the merging dissimilarity, F-NMI seems to be the most sensitive with this type of dissimilarity with both small and big clusterings (figures 4-C ($\mathcal{G}$), 4-C ($\mathcal{G}'$)). While NMI, AMI and Jaccard indices have similar trends among each other and seem to be less sensitive to this type of dissimilarity, ARI seems to be the closest to F-NMI and shows a more linear

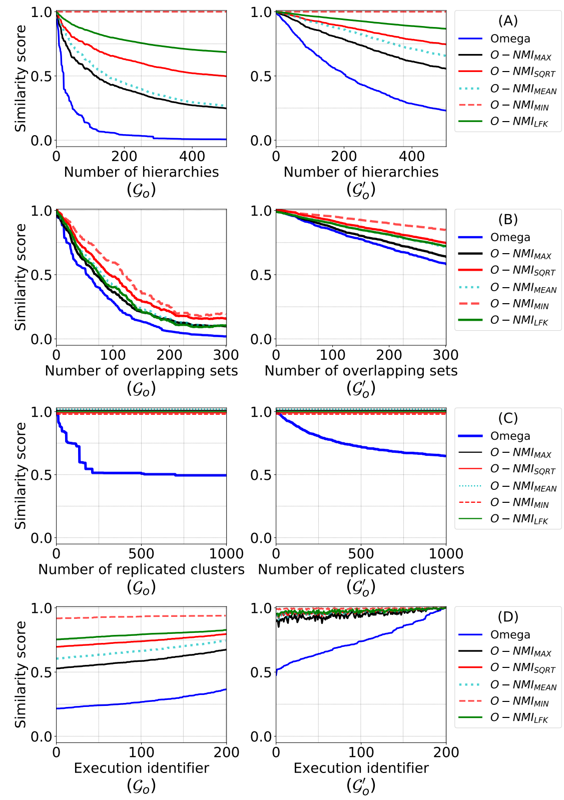


Fig. 5. Effect of hierarchical overlapping dissimilarity **(A)**, partial overlapping dissimilarity **(B)**, replicated overlapping dissimilarity **(C)** and mixed overlapping dissimilarity **(D)** respectively on the non-disjoint indices. $\mathcal{G}_o$, $\mathcal{G}'_o$ refer to the non-disjoint ground truth clusterings resulted by randomly creating overlappings within $\mathcal{G}$, $\mathcal{G}'$ clusters respectively.

behaviour with respect to the amount of cluster merges in big clusterings. Similarly to the merging dissimilarity, F-NMI seems to be the most sensitive with the splitting dissimilarity with both small and big clusterings (figures 4-D ($\mathcal{G}$), 4-D ($\mathcal{G}'$)). While Jaccard index seems to be the closest to F-NMI in its sensitivity to the number of cluster splits and NMI is the least sensitive, ARI and AMI have almost the same behaviour.

The random disjoint dissimilarity experiment (figures 4-E ($\mathcal{G}$), 4-E ($\mathcal{G}'$)) shows that ARI, Jaccard and AMI have similar trends and show almost the same behaviour on average. NMI and F-NMI, however, do not seem to have the same trends like the other disjoint indices.

As to the experiments on non-disjoint indices, figure 5 shows that Omega index is the most sensitive regarding non-disjoint dissimilarities in general. Another observation is that with large clusterings, non-disjoint indices are generally less sensitive to non-disjoint dissimilarities than with small clusterings. Experiments on hierarchical overlapping dissimilarity (figures 5-A ($\mathcal{G}_o$), 5-A ($\mathcal{G}'_o$)) show that $ONMI_{MIN}$ is the worst among other normalizations of $ONMI$ to reflect this type of non-disjoint dissimilarity. The replicated overlapping dissimilarity experiment (figures 5-C ($\mathcal{G}_o$), 5-C ($\mathcal{G}'_o$)) shows that $ONMI_{LFK}$ and all normalizations of $ONMI$ fail at reflecting this type of non-disjoint dissimilarity. The mixed overlapping dissimilarity experiment (figures 5-D ($\mathcal{G}_o$), 5-D ($\mathcal{G}'_o$)) confirms that Omega index is the most sensitive to non-disjoint dissimilarities in general with $ONMI_{MAX}$ being the closest of $ONMI$ alternatives to it. The experiment shows also that with big clusterings, $ONMI_{LFK}$ and all normalizations of $ONMI$ have the same behaviour on average.

V. Discussion

As mentioned in Section IV, the main goal of our experiments is to study the behaviour of different similarity quantification approaches used to compare clusterings identified by community detection methods. More specifically, we are interested in observing whether the different similarity indices have the same trends (**Q1**), analyzing their sensitivity to different types of dissimilarities (**Q2**), verifying whether or not they agree on what is considered dissimilar (**Q3**), and checking if the relationship between these indices and the level of dissimilarity between the compared clusterings is linear (**Q4**). Our ultimate goal is to provide practitioners with some insights on which indices to use for the task at hand and how to interpret their values.

As regards to (**Q1**), the different experiments show that similarity indices do not always follow the same trends and that largely depends on the type of dissimilarity between the two compared clusterings and their sizes as well. With disjoint indices for example, AMI, ARI and Jaccard indices seem to be equivalent on average. Thus, they can be used interchangeably when there is no knowledge in advance about the type of dissimilarity between the compared clusterings. When the type of dissimilarity among the compared clusterings is known upfront, a good choice is F-NMI for merging or splitting disagreements, and AMI for missing nodes and misplaced nodes disagreements. With non-disjoint indices, Omega index seems to be the best choice. Since Omega index can be computationally expensive with clusterings on large node-sets, $ONMI_{MAX}$ can serve as the best available alternative.

As regards to (**Q2**), it is clear based on our experiments that different indices have different sensitivities with respect to the type of dissimilarity between the compared clusterings. Hence, one should be careful when using these indices to draw conclusions about community detection solutions. At the same time, this can be used to anticipate the type of disagreement among the compared clusterings. For example, if F-NMI score is very low while AMI score is still relatively high, one can expect that the two compared clusterings have merging or splitting dissimilarities.

Whether different indices are on an agreement regarding what is considered dissimilar (**Q3**), it is evident that this is not the case as they do not always hit the score 0 together, for example, and some of them do not even hit 0. As to (**Q4**), the experiments showed that the relationship between similarity indices and the amount of disagreement is not necessarily linear as they do have different sensitivities with respect to the type of dissimilarity between the compared clusterings.

Acknowledgment

This research is supported by the VIRT-EU project funded by the European Union's Horizon 2020 research and innovation program under grant agreement No 727040

Acknowledgment

We thank Michele Coscia for useful discussions and his efforts in referring to the implementations of different clusterings similarity indices.

References

[1] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos, "Community detection in social media," *Data Mining and Knowledge Discovery*, 2012.

[2] M. Rosvall, A. Esquivel, A. Lancichinetti, J. West, and R. Lambiotte, "Memory in network flows and its effects on spreading," *Nature Communications*, 2014.

[3] U. Gargi, W. Lu, V. S. Mirrokni, and S. Yoon, "Large-scale community detection on youtube for topic discovery and exploration." 01 2011.

[4] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, pp. 193–218, 1985.

[5] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, pp. 846–850, 1971.

[6] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *J. Mach. Learn. Res.*, vol. 11, pp. 2837–2854, Dec. 2010.

[7] A. Amelio and C. Pizzuti, "Is normalized mutual information a fair measure for comparing community detection methods?" 08 2015.

[8] C. B. Urbani, "A statistical table for the degree of coexistence between two species," *Oecologia*, vol. 44, no. 3, pp. 287–289, Jan 1979.

[9] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, p. 033015, mar 2009.

[10] A. McDaid, D. Greene, and N. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," *CoRR*, 10 2011.

[11] L. Collins and C. Dent, "Omega: A general formulation of the Rand index of cluster recovery suitable for non-disjoint solutions," *Multivariate Behavioral Research - MULTIVARIATE BEHAV RES*, vol. 23, pp. 231–242, 04 1988.

[12] S. Gregory, "Fuzzy overlapping communities in networks," *Journal of Statistical Mechanics: Theory and Experiment*, 2011.

[13] 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*, 2008.

[14] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical review. E, Statistical, nonlinear, and soft matter physics*, 2008.

[15] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Is a correction for chance necessary?" in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 1073–1080.

Appendix D

From Interaction to Participation: The Role of the Imagined Audience in Social Media Community Detection and an Application to Political Communication on Twitter

2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

From interaction to participation: the role of the imagined audience in social media community detection and an application to political communication on Twitter

Obaida Hanteer*, Luca Rossi*, Davide Vega D'Aurelio†, Matteo Magnani†
* DssLab, IT University Of Copenhagen, Denmark
{obha, lucr}@itu.dk
† InfoLab, Department of Information Technology
Uppsala University, Sweden
{davide.vega, matteo.magnani}@it.uu.se

***Abstract*—In the context of community detection in online social media, a lot of effort has been put into the definition of sophisticated network clustering algorithms and much less on the equally crucial process of obtaining high-quality input data. User-interaction data explicitly provided by social media platforms has largely been used as the main source of data because of its easy accessibility. However, this data does not capture a fundamental and much more frequent type of participatory behavior, where users do not explicitly mention others but direct their messages to an invisible audience following a common hashtag. In the context of multiplex community detection, we show how to construct an additional data layer about user participation not relying on explicit interactions between users, and how this layer can be used to find different types of communities in the realm of Twitter political communication.**

I. Introduction

Community detection is one of the most studied topics in social network analysis. While effective community detection algorithms are certainly necessary to identify meaningful communities, another equally crucial aspect is the definition of which connections should form the input data. However, it is generally recognized today that online social media are complex communication systems where different types of interactions are supported, and different network datasets can be built depending on the type of interaction to be studied. If we focus on Twitter, different types of data and different combinations of them have been considered when looking for communities. A common approach is to build a network based on following/follower relations [1], that can be easily obtained from the Twitter API. Researchers have soon realized that interaction networks are also directly available from the tweets, either defined by retweets [2] or by explicit mentions indicated by the @ character [3]. More recently, advances in multiplex social network analysis have led to the application of multiplex community detection methods, motivated by the hypothesis that analyzing these three types of connections together can reveal new types of communities. More recent work [4] has also suggested the organization of interactions (e.g., @ mentions) between users into multiple layers based on the topic in the exchanged text.

IEEE/ACM ASONAM 2018, August 28-31, 2018, Barcelona, Spain
978-1-5386-6051-5/18/$31.00

We claim that a strong limitation of the aforementioned approaches is that they only focus on the explicit interactions among users that take place within the social media: following, retweeting, and mentioning. However, much of Twitter contemporary interaction takes place within the space of polyadic conversations defined by hashtags. By adding a specific hashtag to their tweets, users do not only label the content of the tweet declaring its general topic but also identify the imagined audience [5]. This participation in a shared discussion, taking place on this hashtag-defined topical space, is largely ignored when Twitter data is used for community detection purposes [6], [4]. In our opinion, the reason why this data has been ignored is that, as opposed to explicit interactions where specific users are directly mentioned in tweets, imagined audiences are not explicitly available from social media APIs - being in most cases not precisely known by the users when they are tweeting [7]. Our claim is that the implicit connections among users adopting common hashtags would be a valuable and natural input to a community detection algorithm.

In this paper, we provide the following contributions. First, we discuss the different choices to model Twitter interactions for community detection tasks, claiming that the connections explicitly provided by the Twitter platform omit a fundamental portion of the complex communication patterns happening on this platform, and specifically hide participation dynamics in favor of interaction dynamics. Then, we provide a way to capture this additional social layer about user participation.

II. Topical audience model

A common way to model the multiple types of relationships supported by a social media platform is to use multiplex

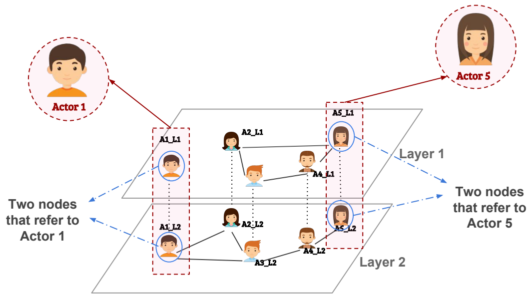


Fig. 1: An example of a multiplex network modeling two modes of interaction among five actors. This is modeled as five replicated nodes in two layers. A node and its replica are linked by a dotted line, to denote that they refer to the same actor, e.g., the same Twitter user

.

networks (Fig. 1). Existing work has already used multiplex networks where layers represent explicit interactions between users. Here we add a layer representing users participation. This layer, that we call topical audience model (TAM), aims at modeling the shared interests among users based on their participation in public discussions. We build the TAM layer in two phases. In the first phase, the discussions among the users of interest are modeled as a multiplex of n layers where n is the number of the topical discussions to be considered in the model. In the context of this paper, we use the explicit hashtag as a proxy for the topic of the shared conversation as suggested by [8]. Each discussion adds a layer to the multiplex and is modeled as a single clique that ties all the users who were part of the discussion by including the same hashtag in their messages. The intuition behind this is that the hashtag functions as a shared channel for discussions about a specific topic where one aims at broadcasting his/her views and opinions to everyone else in the channel.

In the second phase we compute a single network from these topical layers by applying a weighted flattening [9]. In our model, an edge e between $u1$ and $u2$ in the flattened graph has a weight w_e defined using the Jaccard coefficient as:

$$w_e = \frac{N(u_1, u_2)}{N(u_1) + N(u_2) - N(u_1, u_2)} \tag{1}$$

where $N(u_1)$ refers to the number of topical layers user u_1 has been part of and $N(u_1, u_2)$ refers to the number of topical layers users u_1 and u_2 have been both part of.

Once the weights have been computed, we can either keep them if we want to apply a weighted community detection algorithm, or we can use a threshold θ to create a TAM which considers only edges with weights exceeding θ. We do the latter in the case study described in the next section where we also show the effect of using different thresholds.

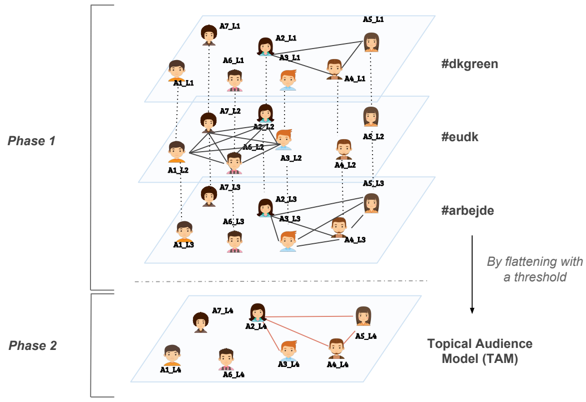


Fig. 2: The two phases in the formation of the Topical Audience Model (TAM). In this example, two actors are connected in Phase 2 if they are connected at least twice in Phase 1

III. A CASE STUDY

The data we use in our case study was collected during the month leading up to the 2015 Danish parliamentary election. Starting from a list of all the Danish politicians running for parliament and with a Twitter account, we collected all their Twitter content produced during the 30 days leading to the election. The initial dataset was formed by 490 politicians distributed across 10 parties, 5985 original tweets, 633 replies and 3993 retweets. Together with their Twitter activity, we also registered the political affiliation of the 490 politicians. Given the complexity of the Danish multi-party system, the parties have also been grouped according to their actual coalitions: Red Block and Blue Block.

	layer	#nodes	#edges	density	ccoef
1	Retweet	212	484	0.0007	0.011
2	Reply	127	169	0.0020	0.175
3	TAM.2	132	1594	0.0065	0.564
4	TAM.5	121	427	0.0017	0.738
5	TAM.7	68	152	0.0006	1.000

TABLE I: Layers used in the analysis. **ccoef** is clustering coefficient

The main focus in our experiments is to execute community detection on different multiplexes constituted of different combinations of Twitter interactions (retweet, reply, and topical interactions represented by TAM) so we can study the nature of the resulted communities on each multiplex. Table I shows the main descriptive data about the layers used to constitute these multiplexes. To build the TAM, the hashtags used by the politicians in the DKPol dataset were listed and qualitatively analyzed. We then excluded the hashtags used for the election campaign and those referring to political TV debates. After this filtering, we were left with only 23 hashtags used to refer to specific topics. The TAM was then constructed in two

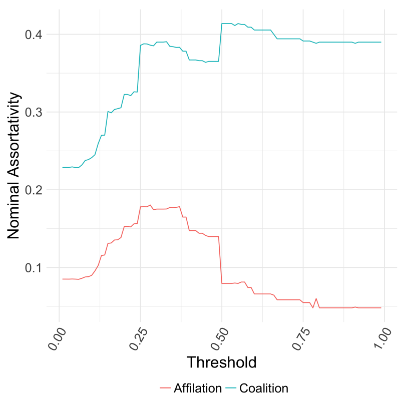


Fig. 3: Nominal assortativity of TAM with respect to the threshold θ

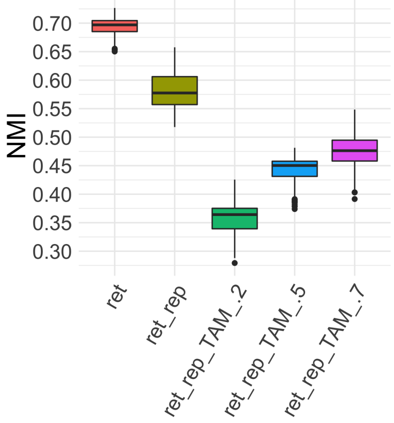


Fig. 4: (NMI) index of clusterigns identified by gLouvain and the political affiliation of ploticians

phases. In the first phase, a multiplex of 23 layers (layer per hashtag) was built as detailed in section II. In the second phase, the multiplex has been flattened into a TAM using a threshold θ to filter out all edges with a weight less than θ. We observed the impact of various thresholds on the nominal assortativity [10] of the TAM layer, measured on the political affiliation and the political coalition of the politicians (Fig 3). We built the TAM layer for 3 different values of θ (0.2, 0.5, and 0.7)

We performed our community detection by using *Generalized Louvain* on 1) only the retweet layer, 2) the multiplex constituted of both the retweet and the reply layers, and 3) the multiplex constituted of retweet, reply and TAM layer (one multiplex per threshold). Compared with other community detection methods, *Generalized Louvain* detected communities that are the closest to the groupings of politicians into political parties. While we do not use this as an evaluation criterion for how good a community detection method is, it is a good starting point to observe how the addition of other layers might affect the resulted communities. The results of community detection using this method on the same multiplex varied slightly from one execution to another, based on the order in which the nodes are scanned by the algorithm. Therefore, we have run the algorithm 1000 times for each experiment. To investigate the social dynamics behind the observed communities beside the structural elements, the communities were evaluated against the groupings of politicians in political parties using the normalized mutual information metric (NMI) [11]. Within the context of this paper, we do not interpret NMI as a "quality" measure of the proposed community structure but as a measure of similarity between the proposed community structure and how the politicians are grouped into political parties.

IV. Results and Discussion

Fig. 4 shows that the highest level of NMI is observed when the communities are detected from the single layer network containing the retweets. Both the multiplex network including the replies, as well as those including the topical layer, score a lower value of NMI when communities are detected. This means that while the retweet layer contains communities that reflect the political affiliation of the politicians, this is no longer clearly visible when communities are detected together with the other relations. This suggests the existence of two different dynamics behind the connections existing on the various layers of the multiplex structure: that of political homophily in the case of the retweet layer and that of different nature for the other layers. Fig. 5 shows the proportion of members belonging to each one of the two coalitions (Blue Block and Red Block) assigned to each one of the communities identified in (a) the retweet network (9 communities) and in the multiplex constituted of the retweet, the reply and the TAM with θ= 0.2 (10 communities). Looking at this figure, it is evident that while the retweet network communities are largely politically homogeneous, the multiplex network including the TAM shows a significant number of communities that are actually formed by the members of both coalitions. This suggests that adding the TAM to the multiplex network allows us to observe interactions between political members that not only belong to different parties but also to different coalitions. While the users on the topical layer were connected because they used the same hashtag to refer to discussion topic during the same political campaign, it is hard to claim that they were not participating in the same conversation. On the contrary, we claim that even if they were not explicitly referring to each other, they were very aware of each other's presence as they were debating in the public topical space defined by the hashtags [12]. While this interaction is not easily captured since it is not readily available through the Twitter API, the proposed approach quantitatively captures the idea of users dealing with their imagined audience as repeatedly observed in qualitative studies of Twitter use [13]. From a political point of view, these results show how Twitter works as a public sphere and how topical debates gathered politicians from opposite parties. This raises the question whether the levels of polarization that have been previously observed in political social media data [14], [2] were actual social dynamics, or the result of the inherently biased data available that was unable to observe non-explicit interactions among users.

While originally introduced by Twitter, the idea of using hashtags to gather communication of users that are not otherwise connected has been adopted in various platforms. These platforms have thus evolved into a form of digital public space where discussions about the news, casual conversations and also political participation take place [15]. While the study of these participatory processes is more and more relevant to understand contemporary society, network approaches have only looked at direct and explicit interactions. By introducing the topical model to study hashtag-based interaction, we propose to extend the range of phenomena that can be fruitfully studied with a network approach. Moreover, we suggest that this model should not be limited to Twitter data and that it could easily be applied to other hashtag-based communicative contexts (e.g. Instagram) as well as to other conceptually similar digital contexts (e.g. participation in Facebook pages).

A future extension to the proposed topical model should include the temporal aspects of interaction into the multiplex network model. While the current implementation assumes a topical stability, it is obvious that topics, as well as the association between actors and topics, change over time. Users might want to discuss a specific issue when it is highly relevant in society and then switch to another topic a few days or even hours later. Twitter itself acknowledges this dynamic through the identification of ever changing trending topics that describe what is being discussed in a specific moment in time, in a specific geographical context. Recent contributions in multiplex networks [4] have proposed to model the temporal dimension as layers of a multiplex structure to be subsequently used for community detection approaches that include temporal information. Such an approach, combined with the topical model we have introduced, could address more of the complexity we encounter in social media, where groups of users discuss within topical spaces constantly moving from one topic to the next one, in an ever evolving network of actors, moments and themes.

V. Conclusions

In this paper we have introduced a novel approach to model the participation in hashtag-based Twitter conversations. We have done this by modelling the participation into a hashtag-based discussion as a layer of a multiplex network where users are connected together if their shared participation is above a given threshold θ. We have also applied this approach in the context of Twitter data collected in 2015 in Denmark during the month leading the the general election.

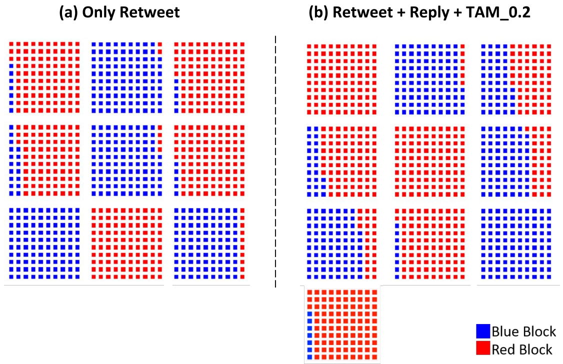


Fig. 5: Proportion of political coalitions (Red Block and Blue Block) within the communities detected on both **a)** only the retweet network **b)** the multiplex network including retweets, replies and the TAM $\theta = .2$ (color figure)

Acknowledgment

This research is supported by the VIRT-EU project funded by the European Union's Horizon 2020 research and innovation program under grant agreement No 727040

References

[1] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 591–600.
[2] M. Conover, J. Ratkiewicz, M. R. Francisco, B. Gonçalves, F. Menczer, and A. Flammini, "Political polarization on twitter," in *Proc. of the fifth International Conference on Web and Social Media*. AAAI Press, 2011, pp. 89–96.
[3] J. Yang and S. Counts, "Predicting the speed, scale, and range of information diffusion in twitter." *Icwsm*, vol. 10, no. 2010, pp. 355–358, 2010.
[4] D. Vega and M. Magnani, "Foundations of temporal text networks," *Applied Network Science*, 3 2018.
[5] E. Litt, "Knock, knock. who's there? the imagined audience," *Journal of Broadcasting & Electronic Media*, vol. 56, no. 3, pp. 330–345, 2012.
[6] T. Highfield, L. Kirchhoff, and T. Nicolai, "Challenges of tracking topical discussion networks online," *Social Science Computer Review*, vol. 29, no. 3, pp. 340–353, 2011.
[7] M. S. Bernstein, E. Bakshy, M. Burke, and B. Karrer, "Quantifying the invisible audience in social networks," in *Proc. of SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013, pp. 21–30.
[8] A. Bruns and J. E. Burgess, "The use of twitter hashtags in the formation of ad hoc publics," in *Proc. of 6th European Consortium for Political Research General Conference*, 2011, pp. 1–9.
[9] M. Berlingerio, M. Coscia, and F. Giannotti, "Finding redundant and complementary communities in multidimensional networks," in *International Conference on Information and Knowledge Management (CIKM)*, 2011, pp. 2181–2184.
[10] M. Newman, "Mixing patterns in networks," *Physical Review E*, vol. 67, no. 2, pp. 261–287, 2011.
[11] A. Lancichinetti and S. Fortunato, "Community detection algorithms: a comparative analysis," *Physical review E*, vol. 80, no. 5, pp. 561–577, 10 2009.
[12] A. Bruns and J. Burgess, "Twitter hashtags from ad hoc to calculated publics," *Hashtag publics: The power and politics of discursive networks*, 2015.
[13] A. E. Marwick and D. Boyd, "I tweet honestly, i tweet passionately: Twitter users, context collapse, and the imagined audience," *New media & society*, vol. 13, no. 1, pp. 114–133, 2011.

[14] A. Bessi, F. Zollo, M. D. Vicario, M. Puliga, A. Scala, G. Caldarelli, B. Uzzi, and W. Quattrociocchi, "Users polarization on facebook and youtube," *PloS ONE*, vol. 11, no. 8, pp. 17–37, 8 2016.
[15] Y. Bonilla and J. Rosa, "Ferguson: Digital protest, hashtag ethnography, and the racial politics of social media in the united states," *American Ethnologist*, vol. 42, no. 11, pp. 4–17, 1 2015.

Appendix E

An Innovative Way to Model Twitter Topic-Driven Interactions Using Multiplex Networks

frontiers in Big Data

ORIGINAL RESEARCH
published: 06 June 2019
doi: 10.3389/fdata.2019.00009

An Innovative Way to Model Twitter Topic-Driven Interactions Using Multiplex Networks

Obaida Hanteer and Luca Rossi**

Data Science and Society Lab, IT University of Copenhagen, Copenhagen, Denmark

We propose a way to model topic-based implicit interactions among Twitter users. Our model relies on grouping Twitter hashtags, in a given context, into themes/topics and then using the multiplex network model to construct a thematic multiplex where each layer corresponds to a topic/theme, and users within a layer are connected if and only if they used the same hashtag. We show, by testing our model on a real-world Twitter dataset, that applying multiplex community detection on the thematic multiplex can reveal new types of communities that were not observed before using the traditional ways of modeling Twitter interactions.

Keywords: multiplex networks, multiplex community detection, thematic communities, thematic clusters, thematic multiplex, social network analysis, social media data analysis

OPEN ACCESS

Edited by:
Roberto Interdonato,
Télédétection et Information Spatiale (TETIS), France

Reviewed by:
Sabrina Gaito,
University of Milan, Italy
Antonio Caliò,
University of Calabria, Italy

****Correspondence:***
Obaida Hanteer
obha@itu.dk
Luca Rossi
lucr@itu.dk

Specialty section:
This article was submitted to Data Mining and Management, a section of the journal Frontiers in Big Data

Received: *01 April 2019*
Accepted: *14 May 2019*
Published: *06 June 2019*

Citation:
Hanteer O and Rossi L (2019) An Innovative Way to Model Twitter Topic-Driven Interactions Using Multiplex Networks. Front. Big Data 2:9. doi: 10.3389/fdata.2019.00009

1. INTRODUCTION

The unprecedented amount of data that is produced, on a daily base, on social media has provided to researchers and practitioners a new opportunity to study, in depth, complex social dynamics at a large scale. Within this context, Twitter can easily claim the award for the most researched social media platform. Thanks to the large user-base and a relatively generous API policy, this micro-blogging platform has quickly evolved into the *de-facto* standard platform for multiple studies on social media dynamics.

The detection of cohesive subgroups in social networks, also called as community detection, has been perceived as one of the most valuable tools to better understand social networks (Papadopoulos et al., 2012). Given that members of the same community tend to share some properties, the community structure of a network can provide a better understanding of the overall functioning of this network. The application of this on social media data has provided useful insights about some of the dynamics and phenomena that take place in such systems (Silva et al., 2017).

A common approach to model Twitter interactions for community detection tasks is to build a network based on following/follower relations (Kwak et al., 2010), or networks based on either retweets (Conover et al., 2011) or explicit mentions indicated by the @ character (Yang and Counts, 2010). Advances on multiplex community detection have suggested that looking at more than one of these types of connections together can provide some insights that cannot be observed by looking at each of them separately. As to the content generated by Twitter users, it has been mostly used for topic detection tasks (Ibrahim et al., 2018) and sentiment analysis (Ceron et al., 2014). To the best of our knowledge, no previous work has addressed extracting network-like information from the content generated by users on social media platforms for community detection tasks.

Much of Twitter contemporary interactions happen in the form of conversations in many-to-many polyadic spaces defined by hashtags (Bruns and Burgess, 2011). In this type of interactions, Twitter users are not necessarily retweeting, replying to, or mentioning each other but engaging directly with specific issues. This suggests that analyzing Twitter data by considering only the direct interactions among users (i.e., following/follower, retweet, and mention networks) is still far from providing a complete picture of Twitter-based interactions. In this paper, we address this gap by proposing an innovative way to model topic-driven interactions of Twitter users using the multiplex network model (Dickison et al., 2016). We test our model, the thematic multiplex, on a real-world dataset capturing the Twitter interactions of the Danish politicians during the parliamentary elections of 2015. We show that detecting communities on the thematic multiplex can reveal different dynamics than those observed by analyzing only explicit interactions. For example, we observed, using thematic multiplex community detection, that while some themes/topics were discussed by almost all the parties within the month leading to the election day, left and right-wing parties, at the same time, have also focused on themes that were politically closer to their traditional ideologies.

The rest of this paper is organized as follows. In section 2 we introduce the thematic multiplex and the thematic multiplex community detection. This is followed by our analysis of a real-world use case (section 3) which captures the Twitter interactions among Danish politicians during the parliamentary elections of 2015. We discuss our results in section 4 and conclude our findings in section 5

2. THE THEMATIC MULTIPLEX

On platforms like Twitter, when a user uses a specific hashtag in a tweet, he/she is not only increasing the visibility of that tweet, but also implicitly, even if not directly, communicating with other Twitter users who are using the same hashtag. This concept has been referred to as the *imagined audience* in the literature (Litt, 2012). Thus, we can assume a social tie (an edge) between two users who used the same hashtag and this is the main idea behind the thematic multiplex. The thematic multiplex, as the name suggests, is a multiplex network where each layer corresponds to a topic/theme and users within a layer are connected via a clique, if and only if, they used the same hashtag. An edge among two actors in the resulted thematic multiplex does not necessarily imply a direct interaction among them yet it suggests that they share a topical-interest. **Figure 1** illustrates a thematic multiplex where each layer represents a specific topic/theme (for example, refugees, education, etc..), and users who used the same hashtag within a topic are connected via a clique, which might result in multiple cliques within a layer (for example, the education theme). **Figure 2** illustrates a possible output for community detection on the thematic multiplex.

We claim that detecting communities on the thematic multiplex network using multiplex community detection can reveal different dynamics than those observed by analyzing the direct interactions among users. The reason is two folded: on one side, direct interactions are often driven by heterogeneous behavior from the users, e.g., Retweets can represent a form of endorsement or just a way to spread an information deemed to be relevant, Replies can equally be produced by amused conversations or endless fights between users. On the other side, direct interactions are just part of the whole Twitter data, thus any approach focusing solely on those will loose potentially relevant information. Thematic multiplex community detection, on the opposite, results in thematic communities were users are grouped together if they tend to discuss/be involved in the same topics/themes through direct or indirect interactions. More over, given that the qualitative analysis is added in the modeling phase, this intrinsically contributes to the qualitative power of community detection on the thematic multiplex network.

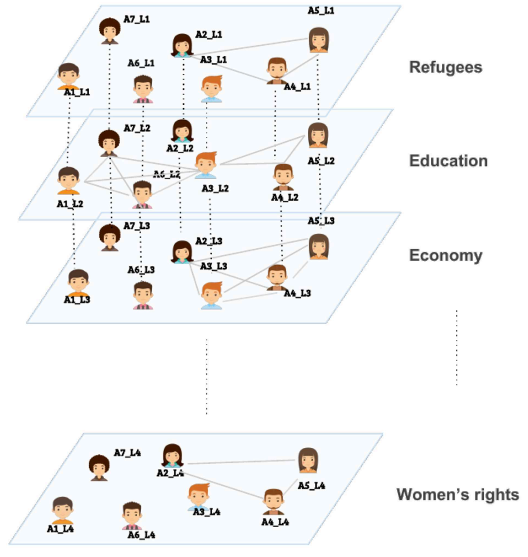


FIGURE 1 | An example of a thematic multiplex.

3. A CASE STUDY

We describe the dataset in section 3.1, then we discuss the construction of the correspondent thematic multiplex and some choices for our analysis tools in section 3.2. We report our observations on the results in section 3.3.

3.1. The DkPol Dataset

The data we use to test our model is collected during the month leading to the 2015 Danish parliamentary election. Starting from a list of all the Danish politicians running for the parliament who also had a Twitter account, we collected all the tweets written during the 30 days leading to the election. The initial dataset was formed by 490 politicians distributed across 10 parties, 5,985 original tweets, 633 replies, and 3,993 retweets. Together with their Twitter activity, we noted also the political affiliation of the 490 politicians. Given the complexity of the Danish multi-party system, the parties have also been

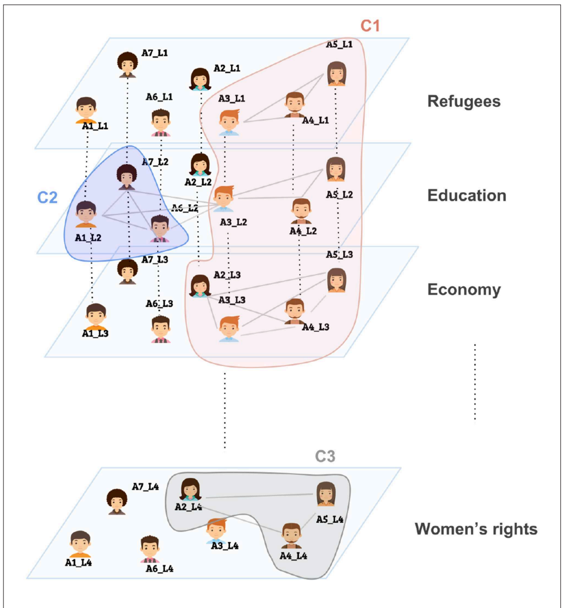


FIGURE 2 | A possible output for applying community detection on a thematic multiplex.

TABLE 1 | The main themes discussed on Twitter by the danish politicians during the parliamentary elections of 2015.

	Theme	Hashtag
1	Children	#dajegvar12
2	Climate	#dkgreen – #talklima – #verdensvildesteforskel
3	Economy	#talop – #dkain – #socialdumping – #nulv
4	Education	#skolechat – #uddpol
5	Election's Practices	#nypolitiskkultur
6	Europe	#eurdk
7	Government Interference	#frihed
8	Health	#sundpol – #sundhed
9	IT	#itpol – #itvalg
10	Refugees	#nuloverdeigen – #engangvarjegflygtning
11	Woman's Rights	#100aaret
12	Work	#arbejde – #dksocial – #dagpenge

grouped into two main coalitions existing at the time: Red Block, currently at the opposition, and the Blue block, currently in government[1]. In order to use the hashtag contained in the tweets to build a thematic multiplex, some initial data cleaning was necessary. The hashtags were first qualitatively analyzed. We then excluded the hashtags that were just about the election campaign as such (like #dkpol) and those referring to political TV debates (like #tv2valg and #DRdinstemme). After this filtering we were left with only 23 hashtags used to refer to specific topics (12 topics). **Table 1** shows the grouping of these hashtags into topics. While our suggested grouping can be further discussed as hashtags can be grouped in many other ways, we chose to keep our focus on the correspondent thematic multiplex and the resulted communities for the sake of this paper.

3.2. Experimental Settings

Given the DkPol dataset, we constructed a twelve-layer thematic multiplex (layer per theme/topic). A topic/theme with k hashtags is interpreted as k cliques in the correspondent layer (a clique per hashtag) among all the users who used the same hashtag. We first show that detecting communities on the thematic multiplex reveals communities that are largely different from those detected using the traditional ways of modeling twitter interactions. **Figures 3**, **4** illustrate the communities detected on the multiplex constituted of the following/follower layer, the retweet layer and the reply layer (A), and those detected on the thematic multiplex (B). The two solutions are largely different in terms of the number of detected communities (8 in the first multiplex, and 3 in the second one), and the composition of each community in terms of the political coalition and the political affiliation of the members constituting each community.

As to the selection of the community detection method for our multiplex networks in this paper, we chose a modularity-maximization based community detection method, Generalized Louvain (Jutla et al., 2017) for this task. The reason is that we consider, by assumption, our networks to be undirected networks and our initial focus is on analyzing the communities resulted by the structural features of the network rather than the information flow. For that reason, we chose Generalized Louvain given that it is a well referenced method in the literature to detect this type of communities. The method define communities by optimizing the modularity of the network. In simple graphs, i.e., one layer networks, this translate to finding the best partitioning of nodes into groups, i.e., communities, that maximize the amount of edges within these groups and minimize the number of edges among them. As to the multi-layer extension of this method, it finds the best partitioning that maximize the multi-layer modularity function which is an extension of the simple modularity defined for simple networks. The extended version of modularity introduces a new parameter to the modularity function that is the coupling parameter ω among nodes that belong to the same actor (i.e., the same Twitter user in our case). When $\omega = 1$ (the default case), this means that the coupling among nodes that belong to the same actor is strong. As a result, a partitioning where multiple nodes that belong to the same user (a node represents the existence of a user in a specific layer) lie in the same community contributes intrinsically to the final score of the extended-modularity. In the rest of this paper, we will refer to the output of a community detection method (which is a set of communities) as a clustering.

[1]The red block coalition groups the following parties: Alternativet, Radikale Venstre, Enhedslisten, Socialdemokratiet, and Socialistisk Folkeparti, while the blue block coalition groups: Dansk Folkeparti, KristenDemokraterne, Liberal Alliance, Venstre, and Det Konservative Folkeparti.

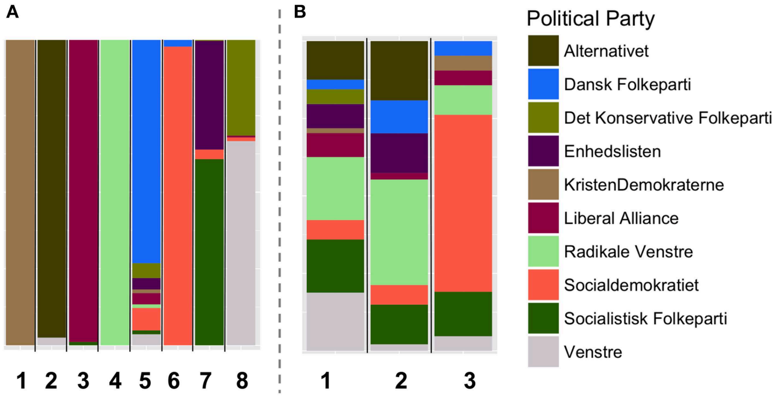


FIGURE 3 | The resulted communities by applying community detection using Generalized Louvain on two different multiplex networks over the DkPol dataset: **(A)** the multiplex constituted of the three layers (following/follower, retweet, and reply) and **(B)** the thematic multiplex. Each bar refers to a different community and the colors in each bar (i.e., community) refer to the composition of each community in terms of the political affiliation of the members constituting it.

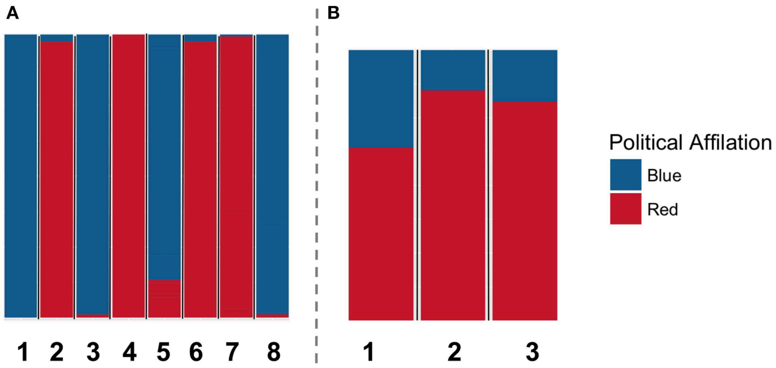


FIGURE 4 | The resulted communities by applying community detection using Generalized Louvain on two different multiplex networks over the DkPol dataset: **(A)** the multiplex constituted of the three layers (following/follower, retweet, and reply) and **(B)** the thematic multiplex. Each bar refers to a different community and the colors in each bar (i.e., community) refer to the composition of each community in terms of the political coalition (red, blue) of the members constituting it.

To better understand the topical dynamics during the month leading to the elections, we chose to create 4 thematic multiplex networks (one for each week content during the month leading the election day). The reason behind choosing "1 week" as a time-window based on which we split the data is that during the month leading to the elections, politicians had to debate on a public TV show once per week.

As illustrated in **Figure 2**, the resulted communities do not necessarily expand over all the layers, meaning that some topics can be absent in some communities. In addition, nodes may not be evenly distributed over layers (for example, community C_1 in **Figure 2** is constituted of 3 nodes in each of the Refugees layer and the Education layer and 4 nodes in the Economy layer). This suggests that topics have different weights, and as a result priorities, in each community which can be interpreted as: some communities, for example, discuss the topic Economy more intensely than they do with the topic Education. To clearly illustrate this, we construct a bipartite network from each clustering. The goal from these bipartite networks is to visualize the relationship between the communities of each clustering and the topics. The width of an edge in the bipartite network between a community and a topic reflects the extent to which that topic is prioritized in that community.

Figure 5 shows the resulted bipartite networks, one per week. We invite our reader to look at this figure together with **Figure 6** which reports , in the form of colored mini-tables , the composition of each community in terms of political coalitions. The existence of a party in a community is represented as a colored cell in the relevant column in that table. The color of that cell can either be red (if the party is from the red Block) or blue (if

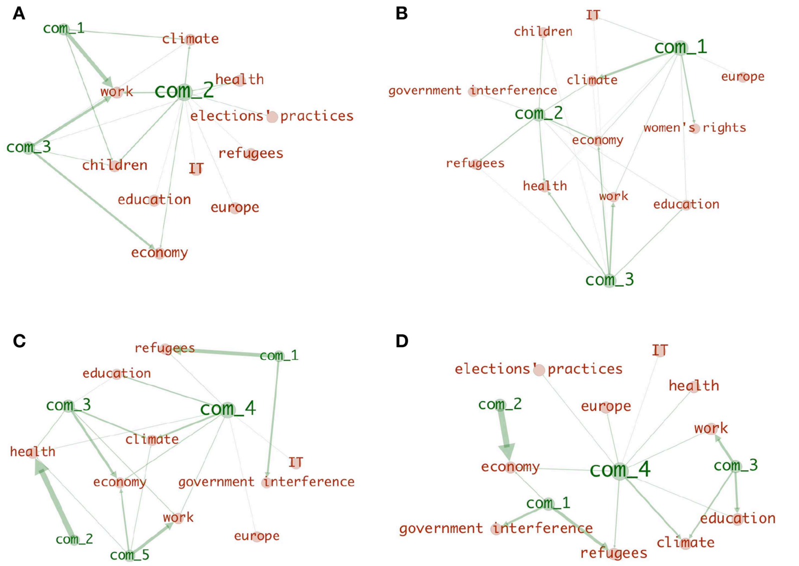


FIGURE 5 | The relationship between the topics and the thematic communities resulted by applying community detection on 4 thematic multiplex newtorks that captured the Twitter interactions of Danish politicians during the month leading the parliamentary lections of 2015 (one per week). **(A)** week 1, **(B)** week 2, **(C)** week 3, **(D)** week 4.

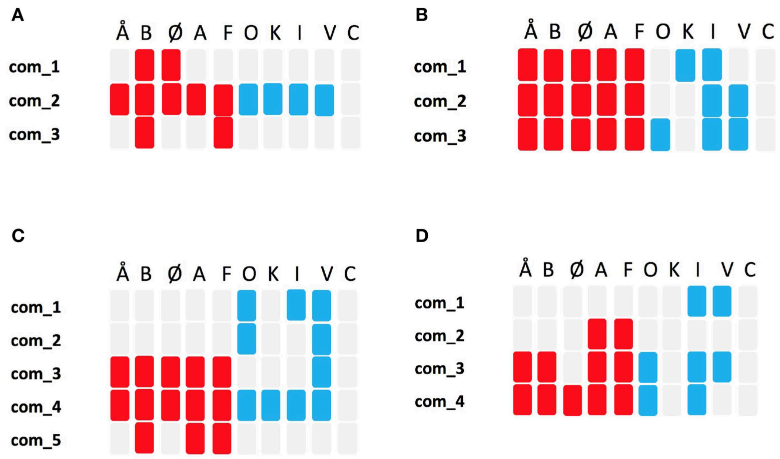


FIGURE 6 | The compositions of the communities reported in **Figure 5** in terms of their political affiliation/party and political coalition/block. **(A)** week 1, **(B)** week 2, **(C)** week 3, **(D)** week 4.

the party is from the blue block). A cell that is neither blue nor red implies the absence of that party (identified by the correspondent column) in the community identified by its row.

3.3. Observations

By looking at **Figure 5A** together with **Figure 6A**, we see that applying community detection on the thematic multiplex of the first week resulted in three communities. Two communities, com_1,com_3, that focus more on economic issues (economy theme and work theme) are composed solely of left-wing (red block) parties. In addition, one community, com_1, constituted of almost all the red block and the blue block parties, tackled all topics with more focus on children, climate, work, and economy themes. Only one of the 12 themes (woman's rights) is absent in all the online debates happened with the first week. The analysis of **Figures 5A**, **6A** shows how during the first week of the election campaign there was a set of bipartisan topics, that were deemed to be central and worth debating, from both political blocks and other themes that were part of political messages of only one of the two blocks.

This scenario seems to change during the second week as **Figure 5B** together with **Figure 6B** report the absence of single-coalition communities. However, the differences among the communities can be observed on the level of their topical interests. For example, com_1 has more focus on woman's rights and climate issues, com_2 equally prioritized refugees, health and economy issues, while com_3 had focused on work, education, work, and economy. It is also interesting to observe how some of the topics that were, during the previous week, part of a single coalition community (e.g., "economic issues" in com_3 during the first week but part of a bipartisan community - com_2 - in the second week), are now part of the bipartisan conversation. While the detailed study of this dynamic process is outside the goal of this paper, this seems to suggest that opposite coalition might follow each others' themes in order to be present in the topical debate.

During the third week can observe a new polarization of the picture. **Figures 5C**, **6C** show, com_1, com_2, constituted of only blue block parities with interests in refugees, government interference, and health issues. One community, com_5, is constituted of only red block parties with interests on economical issue (work and economy themes). One community, com_3, constituted of almost mostly red block parties (with only one blue block party) with interests in both climate and economy. A debate among almost all parties is still present in the third week represented by com_3 with more focus on climate. These topical division seems very much aligned with the core political values of the two blocks at the time of the election.

This topical difference is largely maintained into the fourth week, the week of election, where we can see—**Figures 5D**, **6D**—four thematic communities. Com_1 which is constituted of only right-wing parties (blue block) with interests in refugees and government interference issues, com_2 which is constituted of only left wing parties (red block) with interests only in economy, and both com_3, com_4 which are mixed in terms of the coalitions, and with main interests in (work/education/climate) and climate, respectively.

4. DISCUSSION

A clear difference has been shown when analyzing the communities on the thematic multiplex versus those detected on a multiplex constituted of the following/follower, retweet, and reply layers. This strongly suggests that community detection on the thematic multiplex reveals different dynamics than those observed using traditional ways of modeling twitter interactions. This is not to say that the thematic multiplex can substitute the traditional ways of modeling Twitter activities, but just to shed a light on different dynamics that can be observed using this way of modeling.

Applying longitudinal community detection on the thematic multiplex network obtained from Twitter data allowed us to observe several interesting dynamics. Given that the dataset captured the interactions among Danish politicians during the month leading the parliamentary elections of 2015, we were able to capture the interest of a political party (or coalition) in specific issues, regardless of the fact that the issue produced an explicit interaction with other users through retweets or replies. During a political campaign, when much of the communication is aimed at promoting the party's agenda to the potential voters, which does not necessarily involve retweeting or replying actions , this type of implicit communication is of key importance. Nevertheless, the thematic multiplex network approach was also able to observe the topics that were more contentious between the parties as well as the topics highly polarized. Moreover, the combination between multiplex thematic analysis and longitudinal data allowed us to show how the political debate, and resulting political communities, are highly dynamic and driven by the ongoing events or campaign themes.

While there might exist other ways to model topic driven implicit interactions on Twitter for clustering tasks, we still think that using multiplex network model offers clear advantages. First, the multiplex network model is a well-developed and widely used model for modeling complex systems (Cardillo et al., 2013; De Domenico et al., 2015) and therefore, provides a powerful, and at the same time flexible, modeling tool that allows for translating properties and variables of complex systems into multi-layer graph proprieties. Second, the plethora of community detection methods developed to detect communities in multiplex networks provides practitioners with more power to choose what works the best for the context of their data.

The idea of moving the qualitative analysis to the modeling phase in the thematic multiplex adds lots of power to the interpretability of the output of a community detection task on this multiplex network. While a fully automated approach to group hashtag into themes/topics could seem a tempting idea, the real complexity behind social media hashtagging is still far from being fully understandable by natural language processing tools and text mining technologies currently at hand. An example are two of the hashtags in our collection: *#engangvarjegflygtning* (translated: *one day I was a migrant*) and *#dajegvar12* (translate: *when I was twelve*). In both cases an *emotional* hashtag is used to discuss specific issues, the refugee crisis with the first and children policies with the latter. The connection between the topic and the hashtag is not explicit, and while both hashtags are clearly topical hashtags (thus referring to a specific topic or event and suggesting the desire of the user to participate to an ongoing larger conversation Bruns and Moe, 2014) they also contain an emotional layer that, as well as the specific topic, is hard to understand if taken out of the specific cultural and societal context.

A future iteration on this work should consider testing the thematic multiplex on other datasets. An important extension should also consider the scalability problem with large scale datasets. The main complexity of this model comes from the greedy approach of connecting the user with his imagined audience via a clique. This means that by using a hashtag for only one time, a user is adding to the model a number of edges equals to the number of all other users who used the same hashtag. While a naive approach to minimize the size of these cliques could be to apply a threshold on the number times a user should use a hashtag before being part of the clique, we still think that further research should be carried out to find other alternatives for the clique concept in the thematic multiplex without any loose in the information.

Even though the idea of using hashtags to gather communications of users that are not otherwise connected (e.g., not following each other) was originally introduced by Twitter, many other platforms such as Facebook and Instagram have adopted this idea in various ways. Thus, we suggest that this model should not be limited to Twitter data as it could be easily applied to other hashtag-based communicative contexts (e.g., Instagram) as well as to other conceptually similar digital contexts (e.g., participation in Facebook pages).

On a separate note, we would like to mention the fact the resulted communities may largely depend on the chosen community detection method. Indeed, weather or not the thematic communities will be significantly different among different community detection methods can be a research question on its own and we think that answering this question is out of the scope of this paper.

5. CONCLUSION

In this paper we propose an innovative model, the thematic multiplex, to model topic-driven interactions on Twitter. The thematic multiplex is a multi-layer network where each layer corresponds to a different topic, and users (nodes) within a layer will be connected via a clique if and only if they used the same hashtag. We explain the motivation behind the thematic multiplex which is the fact that it considers implicit interactions among users on Twitter that are usually neglected in other models. We construct the thematic multiplex of a real-world Twitter dataset describing the Twitter interactions among the danish politicians during the parliamentary elections of 2015. We show that applying multiplex community detection on the thematic multiplex allows us to observe different dynamics than those we would observe on other models.

DATA AVAILABILITY

The datasets analyzed for this study can be found in the dkpol GitHub repository on the following link [https://github.com/obaidaITU/dkpol].

AUTHOR CONTRIBUTIONS

OH and LR conceived of the presented idea and developed the theory, discussed the results, and contributed to the final manuscript. OH performed the experiments. LR supervised the findings of this work.

FUNDING

This research is supported by the VIRT-EU project funded by the European Union's Horizon 2020 research and innovation program under grant agreement No. 727040.

REFERENCES

Bruns, A., and Burgess, J. E. (2011). "The use of twitter hashtags in the formation of *ad hoc* publics," in *Proceedings of 6th European Consortium for Political Research General Conference* (Reykjavik: University of Iceland), 1–9.

Bruns, A., and Moe, H. (2014). "Structural layers of communication on twitter," in *Twitter and Society, Vol. 89*, eds K. Weller, A. Bruns, J. Burgess, M. Mahrt, and C. Puschmann (New York, NY: Peter Lang), 15–28.

Cardillo, A., Gómez-Gardeñes, J., Zanin, M., Romance, M., Papo, D., del Pozo, F., et al. (2013). Emergence of network features from multiplexity. *Sci. Rep*. 3:1344. doi: 10.1038/srep01344

Ceron, A., Curini, L., Iacus, S. M., and Porro, G. (2014). Every tweet counts? How sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to Italy and France. *New Media Soc.* 16, 340–358. doi: 10.1177/1461444813480466

Conover, M., Ratkiewicz, J., Francisco, M. R., Gonçalves, B., Menczer, F., and Flammini, A. (2011). "Political polarization on twitter," in *Proceedings of the Fifth International Conference on Web and Social Media* (AAAI Press), 89–96.

De Domenico, M., Nicosia, V., Arenas, A., and Latora, V. (2015). Structural reducibility of multilayer networks. *Nat. Commun.* 6:6864. doi: 10.1038/ncomms7864

Dickison, M. E., Magnani, M., and Rossi, L. (2016). *Multilayer Social Networks*. Cambridge: Cambridge University Press.

Ibrahim, R., Elbagoury, A., Kamel, M. S., and Karray, F. (2018). Tools and approaches for topic detection from twitter streams: survey. *Knowl. Inf. Syst.* 54, 511–539. doi: 10.1007/s10115-017-1081-x

Jutla, I. S., Jeub, L. G. S., and Mucha, P. J. (2017). *A Generalized Louvain Method for Community Detection Implemented in Matlab*. Technical report.

Kwak, H., Lee, C., Park, H., and Moon, S. (2010). "What is twitter, a social network or a news media?," in *Proceedings of the 19th International Conference on World Wide Web* (Raleigh, NC: ACM), 591–600.

Litt, E. (2012). Knock, knock. Who's there? The imagined audience. *J. Broadcast. Electron. Media* 56, 330–345. doi: 10.1080/08838151.2012.705195

Papadopoulos, S., Kompatsiaris, Y., Vakali, A., and Spyridonos, P. (2012). Community detection in social media. *Data Min. Knowl. Discov.* 24, 515–554. doi: 10.1007/s10618-011-0224-z

Silva, W., Santana, A., Lobato, F., and Pinheiro, M. (2017). "A methodology for community detection in twitter," in *Proceedings of the International Conference on Web Intelligence* (New York, NY: ACM), 1006–1009.

Yang, J., and Counts, S. (2010). "Predicting the speed, scale, and range of information diffusion in twitter," in *Proceedings of the Fourth International Conference on Weblogs and Social Media* (Washington, DC), 355–358.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The handling editor and reviewer (SG) declared their involvement as co-editors in the Research Topic, and confirm the absence of any other collaboration

Bibliography

[1] Evelien Otte and Ronald Rousseau. Social network analysis: a powerful strategy, also for the information sciences. *Journal of Information Science*, 28(6):441–453, 2002. doi: 10.1177/016555150202800601. URL https://doi.org/10.1177/016555150202800601.

[2] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*, page 15. Structural Analysis in the Social Sciences. Cambridge University Press, 1994. doi: 10.1017/CBO9780511815478.

[3] Linton Freeman. The development of social network analysis—with an emphasis on recent events. *The SAGE Handbook of Social Network Analysis*, 01 2011.

[4] T. A. A. Broadbent. *Excursions into Mathematics. By A. Beck, M. N. Bleicher and D. W. Crowe. Pp. xxi, 489. 1969. (Worth Publishers, Inc., New York.)*, volume 54. Cambridge University Press, 1970. doi: 10.2307/3612118.

[5] Linton C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977. ISSN 00380431. URL http://www.jstor.org/stable/3033543.

[6] H.M. Blalock. *Social Statistics*. International edition. McGraw-Hill, 1979. ISBN 9780070057524. URL https://books.google.it/books?id=JjiCAAAAIAAJ.

[7] Kai Hwang. *Cloud Computing for Machine Learning and Cognitive Applications*, page 259. The MIT Press, 2017. ISBN 026203641X.

[8] NOAH E. FRIEDKIN. Structural cohesion and equivalence explanations of social homogeneity. *Sociological Methods & Research*, 12(3):235–261, 1984. doi: 10.1177/0049124184012003001. URL https://doi.org/10.1177/0049124184012003001.

[9] Randall. Collins. *Theoretical Sociology*. 1988.

[10] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences. Cambridge University Press, 1994. doi: 10.1017/CBO9780511815478.

[11] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486, 06 2009. doi: 10.1016/j.physrep.2009.11.002.

[12] Michele Coscia. Discovering communities of community discovery. In *Proceedings of the 2019 IEEE/ACM International Conference onAdvances in Social Networks Analysis and MiningFAB 2019, FOSINT-SI 2019, HI-BI-BI 2019*, pages 513–518, Vancouver, Canada, 2019. ISBN 10.1145/3341161.3342860.

[13] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 2004. doi: 10.1103/PhysRevE.69.026113.

[14] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 2006. ISSN 0027-8424. doi: 10.1073/pnas.0601602103. URL https://www.pnas.org/content/103/23/8577.

[15] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, oct 2008. doi: 10.1088/1742-5468/2008/10/p10008. URL https://doi.org/10.1088%2F1742-5468%2F2008%2F10%2Fp10008.

[16] Lucas G. S. Jeub, Marya Bazzi, Inderjit S. Jutla, and Peter J. Mucha. A generalized louvain method for community detection implemented in matlab. http://netwiki.amath.unc.edu/GenLouvain,https://github.com/GenLouvain/GenLouvain, (2011-2019).

[17] Martin Rosvall and Carl T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008. ISSN 0027-8424. doi: 10.1073/pnas.0706851105. URL https://www.pnas.org/content/105/4/1118.

[18] Ludvig Bohlin, Daniel Edler, Andrea Lancichinetti, and Martin Rosvall. *Community Detection and Visualization of Networks with the Map Equation Framework*, pages 3–34. 09 2014. ISBN 978-3-319-10376-1. doi: 10.1007/978-3-319-10377-8_1.

[19] Paul W. Holland, Kathryn B. Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. 1983.

[20] Tiago P. Peixoto. Efficient monte carlo and greedy heuristic for the inference of stochastic block models. *Phys. Rev. E*, 89:012804, Jan 2014. doi: 10.1103/PhysRevE.89.012804. URL https://link.aps.org/doi/10.1103/PhysRevE.89.012804.

[21] Thorben Funke and Till Becker. Stochastic block models: A comparison of variants and inference methods. *PloS one*, 14(4):e0215296–e0215296, Apr 2019. ISSN 1932-6203. doi: 10.1371/journal.pone.0215296. URL https://www.ncbi.nlm.nih.gov/pubmed/31013290.

[22] Thorben Funke and Till Becker. Stochastic block models: A comparison of variants and inference methods. https://github.com/funket/pysbm.

[23] Leto Peel, Daniel B. Larremore, and Aaron Clauset. The ground truth about metadata and community detection in networks. *Science Advances*, 3(5), 2017. doi: 10.1126/sciadv.1602548. URL https://advances.sciencemag.org/content/3/5/e1602548.

[24] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80:016118, Jul 2009. doi: 10.1103/PhysRevE.80.016118. URL https://link.aps.org/doi/10.1103/PhysRevE.80.016118.

[25] Marya Bazzi, Lucas G. S. Jeub, Alex Arenas, Sam D. Howison, and Mason A. Porter. A Framework for the Construction of Generative Models for Mesoscale Structure in Multilayer Networks. *arXiv e-prints*, art. arXiv:1608.06196, Aug 2016.

[26] Obaida Hanteer and Luca Rossi. The meaning of dissimilar: An evaluation of various similarity quantification approaches used to evaluate community detection solutions. In *Proceedings of the 2019 IEEE/ACM International Conference on-Advances in Social Networks Analysis and MiningFAB 2019, FOSINT-SI 2019, HI-BI-BI 2019*, pages 513–518, Vancouver, Canada, 2019. ISBN 978-1-4503-6868-1.

[27] Nicolas Dugué and Anthony Perez. Directed Louvain : maximizing modularity in directed networks. Research report, Université d'Orléans, Nov 2015. URL https://hal.archives-ouvertes.fr/hal-01231784.

[28] Arzum Karataş and Serap Şahin. Application areas of community detection: A review. pages 65–70, 12 2018. doi: 10.1109/IBIGDELFT.2018.8625349.

[29] Arzum Karataş and Serap Şahin. A review on social bot detection techniques and research directions. 10 2017.

[30] Javier O Garcia, Arian Ashourvan, Sarah F Muldoon, Jean M Vettel, and Danielle S Bassett. Applications of community detection techniques to brain graphs: Algorithmic considerations and implications for neural function. *Proceedings of the IEEE. Institute of Electrical and Electronics Engineers*, 106(5):846–867, 05 2018. doi:

10.1109/JPROC.2017.2786710. URL https://www.ncbi.nlm.nih.gov/pubmed/30559531.

[31] Danielle S Bassett, Nicholas F Wymbs, Mason A Porter, Peter J Mucha, Jean M Carlson, and Scott T Grafton. Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences of the United States of America*, 108(18):7641–7646, 05 2011. URL https://www.ncbi.nlm.nih.gov/pubmed/21502525.

[32] Ullas Gargi, Wenjun Lu, Vahab Mirrokni, and Sangho Yoon. Large-scale community detection on youtube for topic discovery and exploration. 01 2011.

[33] Obaida Hanteer and Luca Rossi. An innovative way to model twitter topic-driven interactions using multiplex networks. *Frontiers in Big Data*, 2:9, 2019. ISSN 2624-909X. doi: 10.3389/fdata.2019.00009. URL https://www.frontiersin.org/article/10.3389/fdata.2019.00009.

[34] Jorge Valverde-Rebaza and Alneu Lopes. Link prediction in online social networks using group information. 06 2014. doi: 10.1007/978-3-319-09153-2_3.

[35] P. Diaz. DEFINITIONS OF COMMUNITY. http://uregina.ca/~sauchyn/socialcohesion/definitions%20of%20community.htm. Accessed: 2020-01-31.

[36] D. Lee and H. Newby. *The Problem of Sociology: An Introduction to the Discipline*. Hutchinson university library. Routledge, 1983. ISBN 9780415094535. URL https://books.google.it/books?id=Scz_eCTeecUC.

[37] F. Tönnies, J. Harris, R. Geuss, M. Hollis, and Q. Skinner. *Gemeinschaft und Gesellschaft*. Cambridge Texts in the History of Political Thought. Cambridge University Press, 2001. ISBN 9780521567824. URL https://books.google.it/books?id=1doGaCgMhQ0C.

[38] G Hillery. Definitions of community: Areas of agreement. 20:111–123, 1955.

[39] C. Bell and H. Newby. *Community studies: an introduction to the sociology of the local community*. Studies in sociology. Allen and Unwin, 1971. ISBN 9780043000328. URL https://books.google.it/books?id=Z0pHAAAAMAAJ.

[40] Linton C. Freeman. Turning a profit from mathematics: The case of social networks. *The Journal of Mathematical Sociology*, 10(3-4):343–360, 1984. doi: 10.1080/0022250X.1984.9989975. URL https://doi.org/10.1080/0022250X.1984.9989975.

[41] John C. Turner. Towards a cognitive redefinition of the social group. 1982.

[42] M.E. Shaw. *Group dynamics: the psychology of small group behavior*. McGraw-Hill series in psychology. McGraw-Hill, 1976. ISBN 9780070565012. URL https://books.google.it/books?id=r0S3AAAAIAAJ.

[43] S. Reicher and J. Drury. *The Determination of Collective Behaviour*. Psychology of collective action. University of Bristol, 1982. URL https://books.google.it/books?id=UW25oAEACAAJ.

[44] M. Sherif. *An outline of social psychology*. Harper, 1948.

[45] Jenny Preece and Diane Maloney-Krichmar. Online Communities: Design, Theory, and Practice. *Journal of Computer-Mediated Communication*, 10(4), 07 2017.

[46] N.K. Baym. *Personal Connections in the Digital Age*. Digital Media and Society. Wiley, 2010. ISBN 9780745643311. URL https://books.google.it/books?id=N4jk7DZbR1oC.

[47] Constance Elise Porter. A Typology of Virtual Communities: a Multi-Disciplinary Foundation for Future Research. *Journal of Computer-Mediated Communication*, 10(1), 07 2017. ISSN 1083-6101. doi: 10.1111/j.1083-6101.2004.tb00228.x. URL https://doi.org/10.1111/j.1083-6101.2004.tb00228.x. JCMC1011.

[48] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005. doi: 10.1038/nature03607. URL https://doi.org/10.1038/nature03607.

[49] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002. ISSN 0027-8424. doi: 10.1073/pnas.122653799. URL https://www.pnas.org/content/99/12/7821.

[50] Balabhaskar Balasundaram, Sergiy Butenko, and Illya V. Hicks. Clique relaxations in social network analysis: The maximum k-plex problem. *Operations Research*, 59 (1):133–142, 2011. doi: 10.1287/opre.1100.0851. URL https://doi.org/10.1287/opre.1100.0851.

[51] Soumajit Pramanik, Qinna Wang, Maximilien Danisch, Jean-Loup Guillaume, and Bivas Mitra. Modeling cascade formation in Twitter amidst mentions and retweets. *Social Network Analysis and Mining*, 7(1), December 2017. doi: 10.1007/s13278-017-0462-1. URL https://hal.archives-ouvertes.fr/hal-02363892.

[52] Mark E. Dickison, Matteo Magnani, and Luca Rossi. *Multilayer Social Networks*. Cambridge University Press, 2016. doi: 10.1017/CBO9781139941907.

[53] G. Bianconi. *Multilayer Networks: Structure and Function*, page 4. Oxford University Press, 2018. ISBN 9780198753919. URL https://books.google.it/books?id=6v5cDwAAQBAJ.

[54] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P. Gleeson, Yamir Moreno, and Mason A. Porter. Multilayer networks. *Journal of Complex Networks*, 2(3): 203–271, 07 2014. ISSN 2051-1310. doi: 10.1093/comnet/cnu016. URL https://doi.org/10.1093/comnet/cnu016.

[55] Peter J. Mucha, Thomas Richardson, Kevin Macon, Mason A. Porter, and Jukka-Pekka Onnela. Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 2010. ISSN 0036-8075. doi: 10.1126/science.1184819.

[56] Han Zhang, Chang-Dong Wang, Jian-Huang Lai, and Philip Yu. Modularity in complex multilayer networks with multiple aspects: A static perspective. *Applied Informatics*, 2016. doi: 10.1186/s40535-017-0035-4.

[57] Schneidhofer, Bernhard and Wolthusen, Stephen. Multigraph critical infrastructure model. In Mason Rice and Sujeet Shenoi, editors, *Critical Infrastructure Protection X*, pages 149–167, Cham, 2016. Springer International Publishing. ISBN 978-3-319-48737-3.

[58] Federico Battiston, Vincenzo Nicosia, and Vito Latora. Structural measures for multiplex networks. *Phys. Rev. E*, 89:032804, Mar 2014. doi: 10.1103/PhysRevE.89.032804. URL https://link.aps.org/doi/10.1103/PhysRevE.89.032804.

[59] Vincenzo Nicosia, John Tang, Cecilia Mascolo, Mirco Musolesi, Giovanni Russo, and Vito Latora. Graph metrics for temporal networks. *Temporal Networks*, page 15–40, 2013. ISSN 1860-0840. doi: 10.1007/978-3-642-36461-7_2. URL http://dx.doi.org/10.1007/978-3-642-36461-7_2.

[60] Sonia Kéfi, Vincent Miele, Evie A. Wieters, Sergio A. Navarrete, and Eric L. Berlow. How structured is the entangled bank? the surprisingly simple organization of multiplex ecological networks leads to increased persistence and resilience. *PLOS Biology*, 14:e1002527, 08 2016. doi: 10.1371/journal.pbio.1002527.

[61] W. Fan and K.H. Yeung. Similarity between community structures of different online social networks and its impact on underlying community detection. *Communications in Nonlinear Science and Numerical Simulation*, 20(3):1015 – 1025, 2015. ISSN 1007-5704. doi: https://doi.org/10.1016/j.cnsns.2014.07.002. URL http://www.sciencedirect.com/science/article/pii/S1007570414003165.

[62] Richard F. Betzel and Danielle S. Bassett. Multi-scale brain networks. *NeuroImage*, 160:73 – 83, 2017. ISSN 1053-8119. doi: https://doi.org/10.

1016/j.neuroimage.2016.11.006. URL http://www.sciencedirect.com/science/article/pii/S1053811916306152. Functional Architecture of the Brain.

[63] Darko Hric, Kimmo Kaski, and Mikko Kivelä. Stochastic block model reveals maps of citation patterns and their evolution in time. *Journal of Informetrics*, 12(3):757 – 783, 2018. ISSN 1751-1577. doi: https://doi.org/10.1016/j.joi.2018.05.004. URL http://www.sciencedirect.com/science/article/pii/S1751157717301499.

[64] Ulf Aslak, Martin Rosvall, and Sune Lehmann. Constrained information flows in temporal networks reveal intermittent communities. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, (6), 2018. ISSN 2470-0045. doi: 10.1103/PhysRevE.97.062312.

[65] Manlio De Domenico, Andrea Lancichinetti, Alex Arenas, and Martin Rosvall. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X*, 5:011027, Mar 2015. doi: 10.1103/PhysRevX.5.011027. URL https://link.aps.org/doi/10.1103/PhysRevX.5.011027.

[66] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. URL http://ilpubs.stanford.edu:8090/422/. Previous number = SIDL-WP-1999-0120.

[67] Tiago P. Peixoto. Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Phys. Rev. E*, 92:042807, Oct 2015. doi: 10.1103/PhysRevE.92.042807. URL https://link.aps.org/doi/10.1103/PhysRevE.92.042807.

[68] Amir Ghasemian, Pan Zhang, Aaron Clauset, Cristopher Moore, and Leto Peel. Detectability thresholds and optimal algorithms for community structure in dynamic networks. *Physical Review X*, 6, 06 2015. doi: 10.1103/PhysRevX.6.031005.

[69] Marta Sarzynska, Elizabeth A. Leicht, Gerardo Chowell, and Mason A. Porter. Null models for community detection in spatially embedded, temporal networks. *Journal of Complex Networks*, 4(3):363–406, 11 2015. ISSN 2051-1310. doi: 10.1093/comnet/cnv027. URL https://doi.org/10.1093/comnet/cnv027.

[70] Clara Granell, Richard K. Darst, Alex Arenas, Santo Fortunato, and Sergio Gómez. Benchmark model to assess community structure in evolving networks. *Phys. Rev. E*, 92:012805, Jul 2015. doi: 10.1103/PhysRevE.92.012805. URL https://link.aps.org/doi/10.1103/PhysRevE.92.012805.

[71] Xiao Zhang, Cristopher Moore, and Mark E. J. Newman. Random graph models for dynamic networks. *The European Physical Journal B*, 90(10), Oct 2017. ISSN

1434-6036. doi: 10.1140/epjb/e2017-80122-8. URL http://dx.doi.org/10.1140/epjb/e2017-80122-8.

[72] Yuriy Hulovatyy and Tijana Milenkovic. Scout: simultaneous time segmentation and community detection in dynamic networks. 2016.

[73] Muhammad Pasta and Faraz Zaidi. Network generation model based on evolution dynamics to generate benchmark graphs. 06 2016.

[74] Caterina De Bacco, Eleanor Power, Daniel Larremore, and Cristopher Moore. Community detection, link prediction and layer interdependence in multilayer networks. *Physical Review E*, 95, 01 2017. doi: 10.1103/PhysRevE.95.042317.

[75] Subhadeep Paul and Yuguo Chen. Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel. *Electron. J. Statist.*, 10(2):3807–3870, 2016. doi: 10.1214/16-EJS1211. URL https://doi.org/10.1214/16-EJS1211.

[76] Axel Bruns and Jean E. Burgess. The use of twitter hashtags in the formation of ad hoc publics. In *6th European Consortium for Political Research General Conference*, University of Iceland, Reykjavik, August 2011. URL https://eprints.qut.edu.au/46515/.

[77] Michele Berlingerio, Michele Coscia, and Fosca Giannotti. Finding redundant and complementary communities in multidimensional networks. pages 2181–2184, 10 2011. doi: 10.1145/2063576.2063921.