COMMUNITY DETECTION IN NODE-ATTRIBUTED SOCIAL NETWORKS: A SURVEY

PETR CHUNAEV

ABSTRACT. Community detection is a fundamental problem in social network analysis consisting, roughly speaking, in dividing social actors (modelled as nodes in a social graph) with certain social connections (modelled as edges in the social graph) into densely knitted and highly related groups with each group well separated from the others. Classical approaches for community detection usually deal only with the structure of the network and ignore features of the nodes, although major real-world networks provide additional actors' information such as age, gender, interests, etc., traditionally called node attributes. It is known that the attributes may clarify and enrich the knowledge about the actors and give sense to the detected communities. This has led to a relatively novel direction in community detection — constructing algorithms that use both the structure and the attributes of the network (modelled already via a node-attributed graph) to yield more informative and qualitative results.

During the last decade many methods based on different ideas and techniques have appeared in this direction. Although there exist some partial overviews of them, a recent survey is a necessity as the growing number of the methods may cause uncertainty in practice.

In this paper we aim at clarifying the overall situation by proposing a clear classification of the methods and providing a comprehensive survey of the available results. We not only group and analyse the corresponding methods but also focus on practical aspects, including the information which methods outperform others and which datasets and quality measures are used for evaluation.

CONTENTS

1. Introduction and the problem statement 2
1.1. Overview 2
1.2. Notation and the problem statement 2
1.3. Basis for structural closeness and attribute homogeneity. Fusing topology and semantics 3
1.4. Applications of community detection in attributed social networks 4
1.5. Note on multilayer networks 4
1.6. Note on subspace-based clustering 4
2. Related works and main problems in the area 5
3. Classification of community detection methods for attributed social networks 5
4. Most used attributed social networks and quality measures 7
4.1. Attributed social networks 7
4.2. Measures for community detection quality 8
5. Early fusion methods 10
5.1. Weight-based methods 10
5.2. Distance-based methods 13
5.3. Node-augmented graph distance-based methods 15
5.4. Embedding-based (early fusion) methods 17
5.5. Pattern mining-based (early fusion) methods 19
6. Simultaneous fusion methods 19
6.1. Methods modifying Louvain, Normalised Cut, k-means, k-medoids and kNN algorithms 19
6.2. Metaheuristic-based methods 21
6.3. Non-negative matrix factorisation and matrix compression 22
6.4. Pattern mining-based (simultaneous fusion) methods 24
6.5. Probabilistic model-based methods 25
6.6. Dynamical system-based and agent-based methods 27
7. Late fusion methods 27
7.1. Consensus-based methods 28
8. Conclusion 29

9. Competing Interests Statement 29
10. Acknowledgements 29
References 29

1. INTRODUCTION AND THE PROBLEM STATEMENT

1.1. **Overview.** *Community detection* is a fundamental problem in social network analysis consisting, roughly speaking, in (unsupervised) dividing social actors into densely knitted and highly related groups with each group well separated from the others. Classical approaches for community detection mainly deal only with the structure of social networks and ignore features of the social network actors. There exist a variety of different methods for this task (see [69]) which have shown their efficiency in multiple experiments (see [119, 124]). However, real-world networks clearly provide more information about social actors than just connections between them. Usually, the actors fulfil their profiles with age, gender, interests, etc., and other information that is traditionally called *node attributes*. According to [207], attributes form the second dimension, besides the structural one, in social network representation. There are other classical approaches such as k-means which may use only node attributes to detect communities but completely ignore links between social actors thus not exploiting all available information. A reasonable generalisation of the methods of both types are the ones that take into account both network structure and actors attributes. This is a relatively novel direction [24] in social network analysis which is quite promising as simultaneous usage of structure and attributes may clarify and enrich the knowledge about the social actors, to give meaning to the detected communities and describe the powers that form them.

During the last decade many methods based on different ideas and techniques have appeared in this direction. Although there exist some partial overviews of them, especially in Related Works sections of published papers and in the survey [24] published in 2015, a recent summary of the subject is a necessity as the growing number of the methods may cause uncertainty in practice.

In this paper we aim to clarify the overall situation by proposing a clear classification of the methods and providing a comprehensive survey of the available results in the area. We not only group and analyse the corresponding methods but also focus on practical aspects, including the information which methods outperform others and which datasets and quality measures are used for evaluation.

To make the exposition more formal, we first provide the reader with necessary notation and state the problem of community detection in social networks whose actors are equipped with attributes.

1.2. **Notation and the problem statement.** Traditionally social networks such as online social networks or citation networks are modelled as graphs whose vertices (nodes) represent social actors (users or authors) and edges the relations between the actors (friendships, subscriptions or co-authorships). Actors' attributes (also known as node features or semantic vectors) may be thought as multidimensional node-attached vectors whose elements contain certain features describing the actors. In what follows, we call *node-attributed* or simply *attributed* the networks whose actors' features are available. Clearly, edges as relations between actors may be of different type in real networks. Such networks can be represented via multilayer graphs (each layer containing certain relation type) but in this paper, for the sake of simplicity, we confine ourselves only to one-layer networks (graphs), i.e. to those with edges of one type. We however mention some papers considering community detection in attributed multilayer networks in remarks below.

Let us move to the required definitions. We represent a node-attributed social network as the triple (called a *(node-)attributed graph*) $G = (\mathscr{V}, \mathscr{E}, \mathscr{A})$, where $\mathscr{V} = \{v_i\}_{i=1}^n$ is the set of nodes (vertices) representing social actors, $\mathscr{E} = \{e_{ij}\}_{i,j=1}^n$ the set of edges representing the existing relations between the actors (e_{ij} is an edge between nodes v_i and v_j) such that $\mathscr{E} \subset \mathscr{V} \times \mathscr{V}$, and $\mathscr{A}$ the set of attribute vectors $A(v_i) = \{a_k(v_i)\}_{k=1}^d$ associated with nodes in $\mathscr{V}$ and describing their features. The size of $\mathscr{V}$ is denoted by n or $|\mathscr{V}|$. The size of $\mathscr{E}$ is denoted by m or $|\mathscr{E}|$. The dimension of attribute vectors $A = \{a_k\}$ is d. The domain of a_k, i.e. the set of possible values of this attribute, is denoted by $dom(a_k)$. In these terms, kth attribute of node v_i is referred to as $a_k(v_i)$. The notation introduced is summarised in Figure 1.

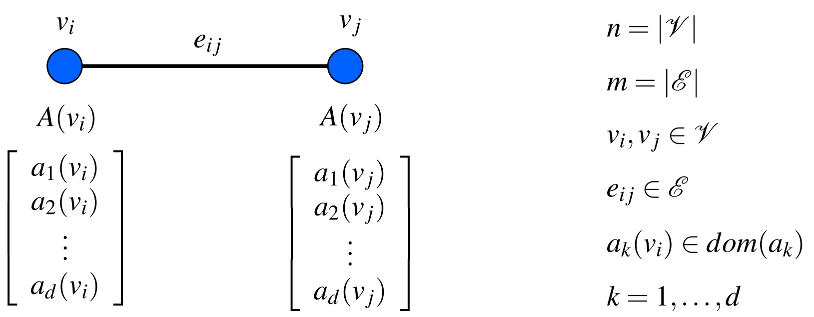

FIGURE 1. Notation used for a node-attributed network in the paper.

By *community detection* in a node-attributed network (graph) (also known as *attributed graph clustering*) we mean *unsupervised* dividing the attributed graph G into N disjoint or overlapping *subgraphs* (equivalently, *clusters* or *communities*) $C_k = (\mathscr{V}_k, \mathscr{E}_k)$, with $C := \{C_k\}_{k=1}^N$, such that all vertices are included in the resulting division, i.e.

$$\mathscr{V} = \bigcup_{k=1}^{N} \mathscr{V}_k,$$

where a certain balance between the following two properties is achieved:

1. **structural closeness**, i.e. nodes within a community are *structurally close* to each other, while nodes in different communities are not;
2. **attribute homogeneity**, i.e. nodes within a community have *similar attributes*, while nodes in different communities do not.

The basis for these properties is discussed in the forthcoming subsection.

Measures for structural closeness and attribute homogeneity may vary from method to method. One can evaluate the quality of community detection via different structure- and attribute-aware measures if no ground truth is available or compare the results with the ground truth, otherwise. We will mention the corresponding measures below.

Structure-attributes fusion techniques and community detection methods for node-attributed social networks are the subject of this paper. Related datasets and quality measures are also of our interest.

In what follows, the structural (topological) information contained in G is referred to as *network structure* or *topology*. The attribute (semantic) information in G is referred to as *network attributes* or *semantics*.

1.3. **Basis for structural closeness and attribute homogeneity. Fusing topology and semantics.** Structural closeness is related to the classical concept of (structural) community in terms of structural connections density. According to [76], communities are thought as subsets of nodes with dense connections within the subsets and sparse in between. [156] adopts the intuition that nodes within the same community should be better connected than they would be by chance to create the famous Modularity measure that became an influential tool for topology-based community detection in social networks [24]. Multiple Modularity modifications and other structural measures have been proposed to overcome several Modularity limitations [34], but Modularity is still a de facto standard in community detection. [156] observes that in social networks Modularity generally belongs to $[0.3, 0.7]$ but there is no particular value for good or bad community structure. In fact, any positive Modularity may indicate the presence of a structural community [41], oppositely to zero Modularity related to a random graph. High Modularity implies the structural closeness of the nodes within communities.

Attribute homogeneity requirement is based on the social science founding (see e.g. [68, 116, 138, 140]) that node attributes in social networks can reflect and affect community structure. The well-known principle of homophily in social networks states that like-minded social actors have a higher likelihood to be connected [140]. Thus community detection process taking into account the attribute homogeneity may provide results of better quality [24].

According to many experiments, e.g. [43, 75, 146, 180, 224] and many other papers cited in this survey, topology and semantics often provide complementary information and thus combining them usually leads to achieving better performance in community detection. For example, the semantics may compensate the sparseness of a real network [106]. At the same time, topological information may be helpful if there are missing or noisy attributes [180]. As observed by [59], topology-only or semantic-only community detection is often not as effective as when both sources of information are used. From the other side, some experiments (see e.g. [5, 231]) suggest that this is not always true and topology and semantics may be orthogonal and contradictory in some cases. Moreover, the relations between network topology and semantics may be highly non-linear [206]. Consequently, the way how one should use network topology and semantics together is a challenging problem.

1.4. **Applications of community detection in attributed social networks.** Community detection in node-attributed networks has not only obvious applications in marketing (recommender systems, targeted advertisements and user profiling) [8], but also can effectively support other multiple advanced applications. First of all, it may be used for search engine optimization and spam detection [149, 172]. Furthermore, community detection methods may help in counter-terrorist activities and disclosing fraudulent schemes [149]. There also exist applications related to the analysis of networks of different nature: protein-protein interactions, genes, epidemics and other biological networks [149].

Another area where the ideas of community detection in attributed networks are generally applied is document network clustering. Note that this direction is historically preceding to the community detection and is rich methodologically. For example, in [153], one of the first papers on community detection in attributed social networks, the following document clustering methods are mentioned: HyPursuit [145, 208, 212], PLSA-PHITS [43], Community-User-Topic model [230] and Link-PLSA-LDA [151]. From that time many others have appeared, see e.g. the surveys [3, 150, 174].

Clearly, methods from document network clustering can be adapted for community detection in attributed social networks, however social communities although have similar formal description with document clusters, have inner and more complicated forces to be formed and act. What is more, it has been shown that some methods for community detection in attributed social networks outperform preceding methods for document network clustering. In particular, **Inc-Cluster**[1] [232] has been shown to outperform k-SNAP [195], **PCL-DC** [222] to outperform PLSA-PHITS [43], LDA-Link-Word [63] and Link-Content-Factorization [233], **CESNA** [220] and **ASCD** [171] to outperform Block-LDA [14]. Taking this into account, we do not consider methods focused on document network clustering in the present survey.

1.5. **Note on multilayer networks.** Generally speaking, we do not aim at considering community detection methods for attributed multilayer networks (see e.g. [115]), where different types of vertexes and edges may present at different layers. However, we mention some of such methods from time to time in corresponding remarks. Although node-attributed single-layer networks may be considered as a particular case of the multilayer ones (or, generally, feature-rich networks [103]), the latter require special analysis to take into account the heterogeneity of attributes, edges and vertices on different layers. A separate survey and an extensive comparable study of such methods is an independent and useful task (see partial overviews e.g. in [25, 103, 115]).

1.6. **Note on subspace-based clustering.** According to the above-mentioned definition of community detection in attributed social networks, we mainly confine ourselves in the survey to the methods that can use the full attribute space and find communities covering the whole network. However, there is a big class of special methods that explore subspaces of attributes and/or find significant subgraphs of the network graph, e.g. GAMer [83, 87], DB-CSC [86], SSCG [88], FocusCO [166] and ACM [210]. The main idea behind the subspace-based (also known as projection-based) attributed graph clustering is that not all available semantic information is relevant to obtain good-quality communities [84, 85], therefore one has somehow choose the appropriate attribute subspace to avoid the so-called *curse of dimensionality* (see [24, Section 3.2]) and reveal significant communities that would not be detected if all available attributes were considered.

[1]Throughout the text, methods and datasets covered by the survey are written in **bold**.

To be precise, some of the methods that we discuss below partly use this idea, e.g. **WCru** [50,51] (cf. the definition of a *point of view* in the papers), **DVil** [201], **SCMAG** [101], **UNCut** [224], **DCM** [170], etc., but still can work with the full attribute space. In any case, a separate survey on the subspace-based attributed graph clustering methods would be very a valuable complement to the current survey.

2. Related works and main problems in the area

There is a variety of surveys and comparative studies considering community detection in social networks without attributes, in particular, [46,69,178,223]. In opposite, the survey [24] seems to be the only one on community detection in attributed social networks. Obviously, since it was published in 2015, many new methods adapting different techniques have appeared in the area. Furthermore, a big amount of the methods that had been available before 2015 are not covered by [24], in particular, some based on objective function modification, non-negative matrix factorisation, probabilistic models, ensembles, etc. In a sense, the technique-based classification of attributed graph clustering methods in [24] is also sometimes confusing. For example, **CODICIL** [172], a method based on assigning attribute-aware weights on graph edges, is not included in [24, Section 3.2. Weight modification according to node attributes], but to [24, Section 3.7. Other methods]. Although [24] is a nice highly cited survey in the area, a recent survey of community detection methods for attributed social networks is clearly required.

Besides [24], almost every paper on the topic contains a Related Works section. It typically has a short survey on preceding approaches and an attempt to classify them. We observed that many authors are just partly aware of the corresponding bibliography and this sometimes leads to repetitions in approaches. Furthermore, multiple classifications (usually technique-based) are mainly not full and even contradictory.

Another big problem in the area is a comparative study of known methods (by means of scalability, complexity and quality). Separate papers provide a limited impact on this (as usually compare their own method with few known ones), see Figures 2 and 3, and the whole picture is unclear. In fact, we are unaware of any comprehensive unified comparison of different attributed graph clustering methods. One more issue, related to the previous one, is that authors use different datasets (of various size and nature) and quality measures to evaluate their methods so that any direct comparison becomes impossible. What is more, datasets and code sources stay unavailable for comparison experiments in the majority of cases.

Facing the above-mentioned problems, in the current survey we not only collect the existing methods but also proposed their unified classification based on the moment when topology and semantics of the network are fused and used in the corresponding algorithm. We also focus on the experimental part so that one can see which networks (with the corresponding dataset link) and quality measures are used in each paper and which methods were compared in each study. Besides this, we also provide the reader with a short description of the most influential and interesting methods for community detection in attributed social networks.

The survey covers the papers published in journals and conference proceedings before the middle of 2019. Exceptionally we sometimes note preprints available on arxiv.

3. Classification of community detection methods for attributed social networks

In previous works, the classification of methods for community detection in attributed social networks was done mostly with respect to the techniques used (e.g. distance-based or random walk-based). We partly follow this methodology at a lower level but at the upper level we group the methods by the moment when topology and semantics are used and fused in the method (with respect to the community detection (clusterisation) step), see Figure 4. Namely, we distinguish

- **early fusion methods** that fuse topology and semantics before the clusterisation step,
- **simultaneous fusion methods** that fuse topology and semantics during the clusterisation step, and
- **late fusion methods** that fuse topology and semantics after the clusterisation step.

Within each fusion type, we also divide the methods into technique-used subclasses.

A subclassification that is applied to some subclasses of early fusion methods is by the modification of the initial network topology (structure). In fact, the existing topology may be saved or modified depending on the heuristics used, therefore we distinguish

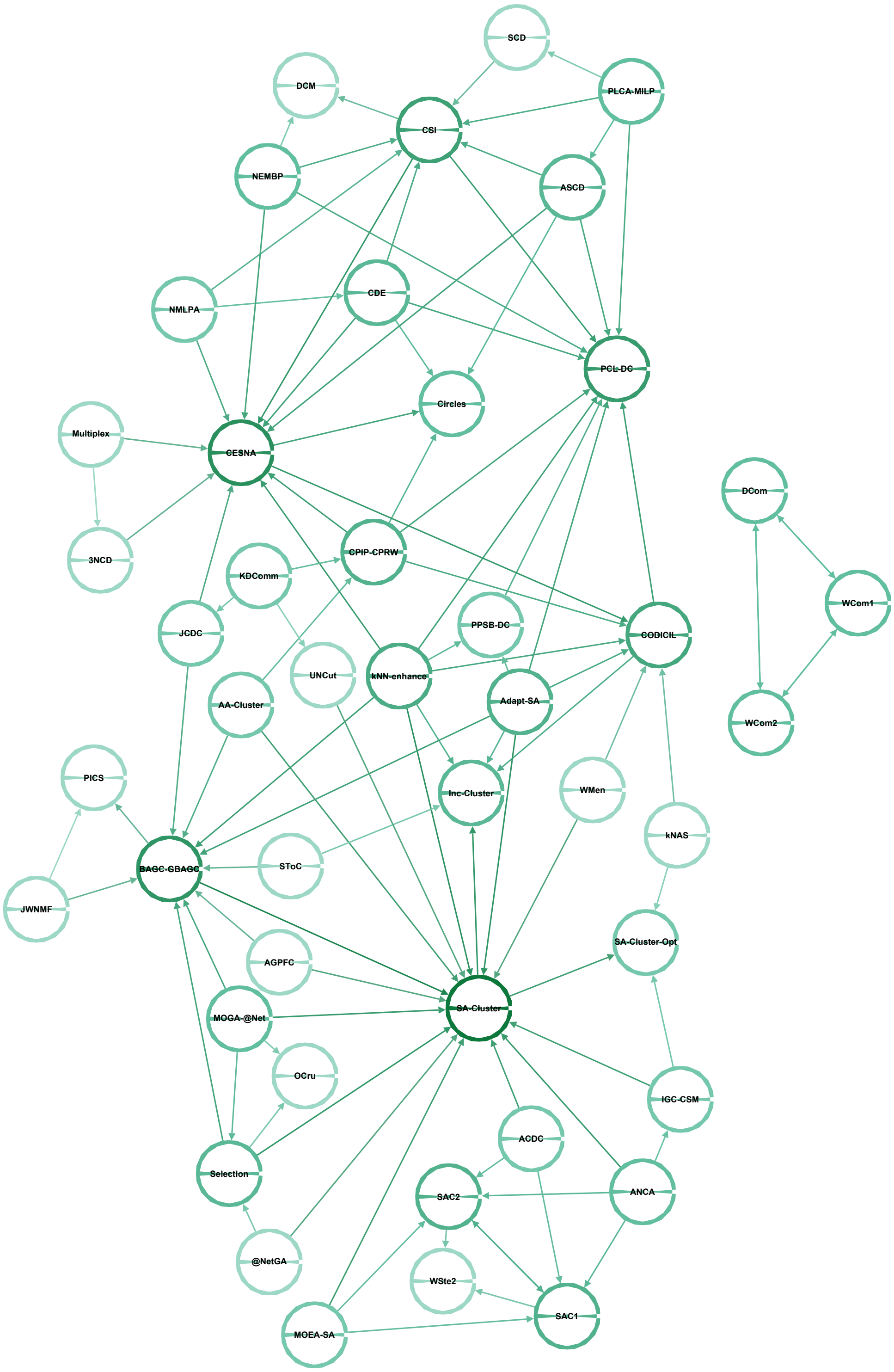

FIGURE 2. The directed graph of existing method-method comparisons. Nodes (shown only those with degree $\geqslant 2$) represent the methods classified in the present survey.

- **fixed topology methods** that use the existing network topology without modifying it with respect to the semantics,
  and
- **non-fixed topology methods** that modify the existing network topology with respect to the semantics, in particular, add/erase edges and/or vertices.

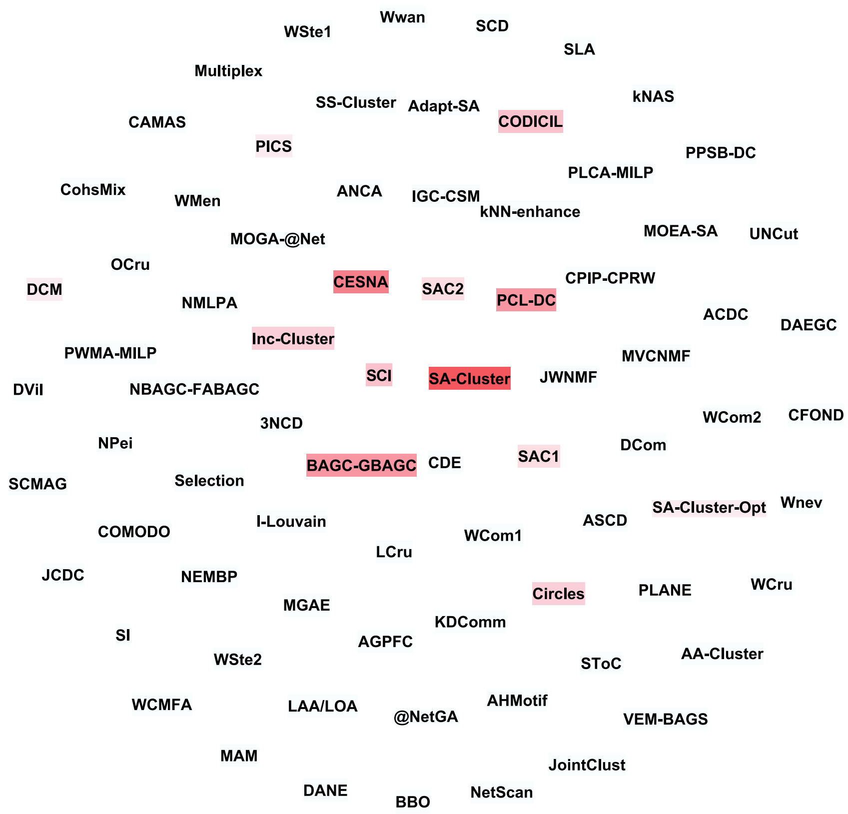

FIGURE 3. The most influential methods (i.e. the ones newer methods most compared with) among the methods classified in the present survey are shown in red.

It is important to distinguish the cases as each one leads to certain advantages or disadvantages. For example, if one assigns edge weights between all nodes in the network, even if there is no structural connections (i.e. considers non-fixed topology) and further removes edges with tiny weights, then in the social network settings this may lead to the following: (a) nodes representing social actors who are highly related in terms of semantics may have vanishing social connections so that the resulting connection may seem unrealistic, (b) one may erase too many important connections. At the same time, the initial “fixed” topological structure may be sparse or noisy in a network and some kind of its enrichment is required. In any case, a proper balance between pure non-fixed and fixed topologies is usually necessary.

As we have already mentioned, the lowest level of classification is by fusion technique. For example, by “weight-based methods” we mean those which form a weighted graph while fusing topology and semantics. Some of the methods further use weighted graph clusterisation algorithms (and this is reasonable) but some may still transform the graph into a distance matrix and use distance-based methods for clusterisation, though are still called “weight-based”. On the other hand, “distance-based methods” are called in this way as produce a distance matrix at the fusion step.

4. MOST USED ATTRIBUTED SOCIAL NETWORKS AND QUALITY MEASURES

4.1. **Attributed social networks.** It can be observed that “social networks” in many papers mean not only real social networks (like Google+, Facebook, Twitter) but also citation networks (like DBLP and CiteSeer). In fact, citations and blogs are the most popular examples in experiments, while real social networks (say, with friendship connections between users) are not.

By **small**, **medium** and **large networks** we mean those with $< 10^3$, $10^3 \dots 10^5$ and $> 10^5$ nodes. The most popular datasets used in experiments on community detection in node-attributed social networks

FIGURE 4. All the methods classified in the present survey. Early fusion (41%), simultaneous fusion (51%) and late fusion (8%) methods are shown in green, blue and red boxes, respectively. Edges indicate existing method-method comparisons.

are collected [2] in Tables 1, 2 or 3. Below, datasets used for evaluation of each method are shown in Dataset columns[3]. Note also that other versions of the networks from Tables 1, 2 or 3 can be used in fact in different papers, and to show this we mark such datasets by *. For example, a DBLP dataset with the number of nodes and edges different from the described **DBLP10K** and **DBLP84K** is denoted by **DBLP***.

In most cases, the attributes suitable for the methods discussed in the survey are represented by continuous numerical vectors. If one deals, say, with nominal, textual or graphical attributes, it is common to use TF-IDF or other similar frameworks to obtain continuous numerical vectors instead.

4.2. **Measures for community detection quality.** Given the set of detected communities (overlapping or not), one needs to evaluate the quality of the communities. There are two possible options depending on the network under consideration. If the network has no ground truth, one can measure structural closeness and attribute homogeneity directly. According to our observations, the most popular quality measures in this case are Modularity and Density for the former and Entropy for the latter. Many others such as Conductance, Within Cluster Sum of Squares, Intra-cluster distance, etc., are also possible. If there is ground truth, it is sometimes reasonable to compare the detected communities with the known ones. This can be done, for instance, with the following popular measures: Accuracy, Normalised Mutual Information (denoted below by NMI), Adjusted Rand Index or Rand Index (denoted below by ARI and RI, correspondingly) and F-measure.

[2] An interested reader can find other attributed network dataset at Mark Newman page, HPI Information Systems Group, LINQS Statistical Relational Learning Group, Stanford Large Network Dataset Collection, Laboratory of Cell Trafficking and Signal Transduction, University of Verona, Marc Plantevit page, Tore Opsahl page, UCINET networks, Interactive Scientific Network Data Repository, Citation Network Dataset.

[3] Recall that if a dataset name is written in **bold**, its description can be found in Tables 1, 2 or 3

TABLE 1. Most popular small real-world attributed social networks

Network	Description	Source
Political Books	All books in this dataset were about U.S. politics published during the 2004 presidential election and sold by Amazon.com. Edges between books means two books are always bought together by customers. Each book has only one attribute termed as political persuasion, with three values: 1) conservative; 2) liberal; and 3) neutrality	Link
WebKB	A classified network of 877 webpages (nodes) and 1608 hyperlinks (edges) gathered from four different universities Web sites (Cornell, Texas, Washington, and Wisconsin). Each web page is associated with a binary vector, whose elements take the value 1 if the corresponding word from the vocabulary is present in that webpage, and 0 otherwise. The vocabulary consists of 1703 unique words. Nodes are classified into five classes: course, faculty, student, project, or staff.	Link [47]
Twitter	A collection of several tweet networks: 1) Politics-UK dataset is collected from Twitter accounts of 419 Members of Parliament in the United Kingdom in 2012. Each user has 3614-dimensional attributes, including a list of words repeated more than 500 times in their tweets. The accounts are assigned to five disjoint communities according to their political affiliation. 2) Politics-IE dataset is collected from 348 Irish politicians and political organizations, each user has 1047-dimensional attributes. The users are distributed into seven communities. 3) Football dataset contains 248 English Premier League football players active on Twitter which are assigned to 20 disjoint communities, each corresponding to a Premier League club. 4) Olympics dataset contains users of 464 athletes and organizations involved in the London 2012 Summer Olympics. The users are grouped into 28 disjoint communities, corresponding to different Olympic sports.	Link 1 Link 2 [77]
Lazega	A corporate law partnership in a Northeastern US corporate law firm; possible attributes: (1: partner; 2: associate), office (1: Boston; 2: Hartford; 3: Providence); 71 nodes and 575 edges	[121]
Research	A research team of employees in a manufacturing company; possible attributes: location (1: Paris; 2: Frankfurt; 3: Warsaw; 4: Geneva), tenure (1: 1–12 months; 2: 13–36 months; 3: 37–60 months; 4: 61+ months); 77 nodes and 2228 edges	[48]
Consult	the relationship between employees in a consulting company; possible attributes: organisational level (1: Research Assistant; 2: Junior Consultant; 3: Senior Consultant; 4: Managing Consultant; 5: Partner), gender (1: male; 2: female); 46 nodes and 879 edges	[48]

TABLE 2. Most popular medium real-world attributed social networks

Network	Description	Source
Political Blogs	A non-classified network of 1,490 webblogs (nodes) on US politics with 19,090 hyperlinks (edges) between the webblogs. Each node has an attribute describing its political leaning as either liberal or conservative (represented by 0 and 1).	Link [2]
DBLP10K	A non-classified co-author network extracted from DBLP Bibliography (four research areas of database, data mining, information retrieval and artificial intelligence) with 10,000 authors (nodes) and their co-author relationships (edges). Each author is associated with two relevant categorical attributes: prolific and primary topic. For attribute “prolific”, authors with $\geqslant 20$ papers are labelled as highly prolific; authors with > 10 and < 20 papers are labelled as prolific and authors with $\leqslant 10$ papers are labelled as low prolific. Node-attribute values for “primary topic” (100 research topics) are obtained via topic modelling. Each extracted topic consists of a probability distribution of keywords which are most representative of the topic.	Link [232]
DBLP84K	A larger non-classified co-author network extracted from DBLP Bibliography (15 research areas of database, data mining, information retrieval, artificial intelligence, machine learning, computer vision, networking, multimedia, computer systems, simulation, theory, architecture, natural language processing, human-computer interaction, and programming language) with 84,170 authors (nodes) and their co-author relationships (edges). Each author is associated with two relevant categorical attributes: prolific and primary topic, defined in a similar way as in DBLP10.	Link [232]
Cora	A classified network of machine learning papers with 2,708 papers (nodes) and 5,429 citations (edges). Each node is attributed with a 1433-dimension binary vector indicating the absence/presence of words from the dictionary of words collected from the corpus of papers. The papers are classified into 7 subcategories: case-based reasoning,genetic algorithms, neural networks, probabilistic methods, reinforcement learning, rule learning and theory.	Link 1 Link 2 [179]
CiteSeer	A classified citation network in the field of machine learning with 3,312 papers (nodes) and 4,732 citations (edges). Each node is attributed with a binary vector indicating the absence/presence of the corresponding words from the dictionary of the 3,703 words collected from the corpus of papers. Papers are classified into 6 classes.	Link 1 Link 2 [179]
Sinanet	A classified microblog user relationship network extracted from the sina-microblog website (http://www.weibo.com) with 3,490 users (nodes) and 30,282 relationships (edges). Each node is attributed with 10-dimensional numerical attributes describing the interests of the user.	Link [106]
PubMed Diabetes	A classified citation networks extracted from the PubMed database pertaining to diabetes. It contains 19,717 publications (nodes) and 44,338 citations (edges). Each node is attributed by a TF-IDF weighted word vector from a dictionary that consists of 500 unique words.	Link
Facebook100	A non-classified Facebook users network with 6,386 users (nodes) and 435,324 friendships (edges). The network is gathered from Facebook users of 100 colleges and universities (e.g. Caltech, Princeton, Georgetown and UNC Chapel Hill) in September 2005. Each user has the following attributes: ID, a student/faculty status flag, gender, major, second major/minor (if applicable), dormitory(house), year and high school.	Link [197, 198]
ego-Facebook	Dataset consists of ’circles’ (’friends lists’) from Facebook with 4039 nodes and 88234 edges. Facebook data was collected from survey participants using a Facebook app. The dataset includes node features (profiles), circles, and ego networks.	Link [125]
LastFM	A network gathered from the online music system Last.fm with 1,892 users (nodes) and 12,717 friendships on Last.fm (edges). Each node has 11,946-dimensional attributes, including a list of most listened music artists, and tag assignments.	Link
Delicious	A network of 1,861 nodes, 7,664 edges and 1,350 attributes. This is a publicly available dataset from the HetRec 2011 workshop that has been obtained from the Delicious social bookmarking system. Its users are connected in a social network generated from Delicious mutual fan relations. Each user has bookmarks, tag assignments, that is, [user, tag, bookmark] tuples, and contact relations within the social network. The tag assignments were transformed to attribute data by taking all tags that a user ever assigned to any bookmark and assigning those to the user.	Link
Wiki	A network with nodes as web pages. The link among different nodes is the hyperlink in the web page. 2,405 nodes, 12,761 edges, 4,973 attributes, 17 labels	Link
ego-Twitter	This dataset consists of ’circles’ (or ’lists’) from Twitter. Twitter data was crawled from public sources. The dataset includes node features (profiles), circles, and ego networks. Nodes 81306, Edges 1768149	Link [125]

Due to space limitations, we refer the reader to the comprehensive survey [34] and to [24, Sections 2.2 and 4], where all the above-mentioned evaluation metrics and many others are precisely defined and discussed in detail.

TABLE 3. Most popular large real-world attributed social networks

Network	Description	Source
Flickr	A network with 100,267 nodes, 3,781,947 edges and 16,215 attributes collected from the internal database of the popular Flickr photo sharing platform. The social network is defined by the contact relation of Flickr. Two vertices are connected with an undirected edge if at least one undirected edge exists between them. Each user has a list of tags associated that he/she used at least five times. Tags are limited to those used by at least 50 users. Users are limited to those having a vocabulary of more than 100 and less than 5,000 tags.	[172] A version of the dataset
Patents	A patent citation network with vertices representing patents and edges depicting the citations between. A subgraph containing all the patents from the year 1988 to 1999. Each patent has six attributes, grant year, number of claims, technological category, technological subcategory, assignee type, and main patent class. There are 1,174,908 vertices and 4,967,216 edges in the network.	Link Larger dataset
ego-G+	This dataset consists of 'circles' from Google+. Google+ data was collected from users who had manually shared their circles using the 'share circle' feature. The dataset includes node features (profiles), circles, and ego networks. Nodes 107,614, Edges 13,673,453. Each node has four features: job title, current place, university, and workplace. A user-pair(edge) is compared using knowledge graphs based on, Category: Occupations, Category:Companies by country and industry, Category: Countries, Category:Universities and colleges by country.	link [125]

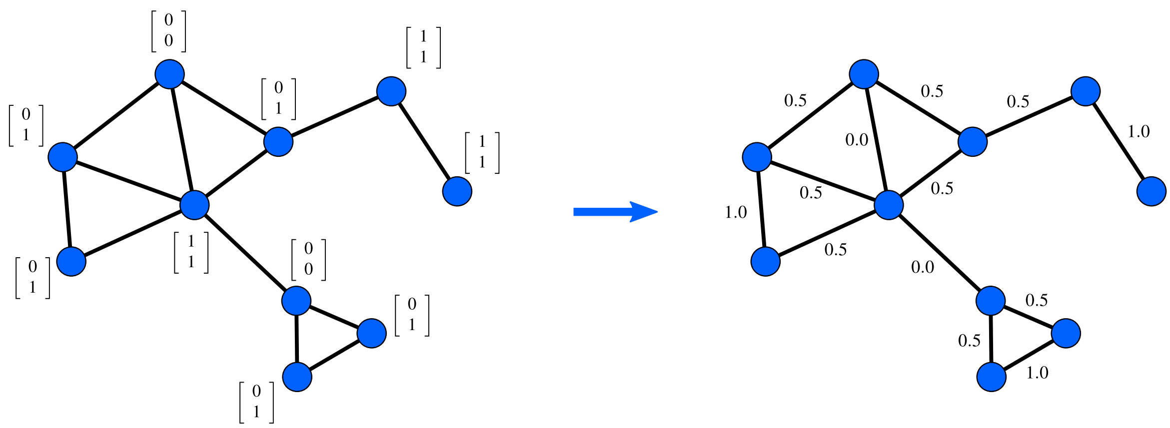

FIGURE 5. The scheme of the weight-based methods (the attribute-aware weights are here calculated with the normalised matching coefficient).

5. EARLY FUSION METHODS

These methods aim to fuse topological and semantic information before the clusterisation step so that the data obtained at the fusion step is suitable for conventional clusterisation methods.

5.1. **Weight-based methods.** The main characteristic of these methods is that the semantics is used to assign weights on edges of the network graph G (the topology may be fixed or non-fixed), see Figure 10, so that the resulting weighted graph G_W can be further clustered e.g. by a clusterisation algorithm for weighted graphs such as Weighted Louvain [21] (requiring the adjacency matrix for the weighted graph as an input). There are also several algorithms that still find the distance matrix for G_W and apply distance-based clusterisation algorithms such as k-means and k-medoids. In other words, weight-based methods remove attribute information by storing it inside the structure, namely, on the edges of the graph.

The weights are usually assigned on edges e_{ij} as follows:

$$W(e_{ij};\alpha) = \alpha w_T(e_{ij}) + (1-\alpha) w_S(e_{ij}), \qquad \alpha \in [0,1], \tag{5.1}$$

where w_T and w_S are chosen *topological similarity function* and *semantic similarity function* for nodes v_i and v_j, respectively. The parameter α controls the balance between topological and semantic components so that $\alpha = 1$ corresponds to the pure topological case and $\alpha = 0$ to the semantic one. Generally speaking, one may introduce a non-linear fusing function instead of (5.1), however such a choice clearly complicates the model and thus requires reasonable justification.

The fixed topology assumes that only existing edges e_{ij} are assigned with the weight (5.1), while the non-fixed one assigns weights on the edges of the complete graph $\bar{G}$ based on G. In the former case, usually $w_T(e_{ij}) \equiv 0$ for e_{ij} that did not exist in the initial graph G, and $W(e_{ij};\alpha)$ is thus generated only by the semantic similarity component.

A very popular approach in fixed-topology case is assuming $\alpha = 0$ in (5.1), see Table 4. This effect may be also achieved by assuming $w_T(e_{ij}) \equiv 0$ for all e_{ij}. Actually this means that the weights in G_W

are based only on the semantic similarity. Clearly, this may lead to the dominance of the semantics component and the break of initial structural connection between nodes with dissimilar attributes.

The approach with $\alpha \in [0,1]$ in (5.1), see Table 5, seems more adequate with respect to $\alpha = 0$ as explicitly allows controlling the impact of both components. For unweighted graphs G, it is common to put $w_T(e_{ij}) = 1$ if edge e_{ij} exist in G, and $w_S(e_{ij}) = 0$ otherwise.

As for w_S, there are several popular measures. Assume that we are given two attribute vectors $x = (x_k)_{k=1}^d$ and $y = (y_k)_{k=1}^d$. One can define w_S basing on

- the (normalised) matching coefficient

$$\mathsf{MC}(x,y) = \frac{1}{d}\sum_{k=1}^{d} \mathbb{1}(x_k = y_k) \in [0,1], \tag{5.2}$$

- the cosine similarity

$$\mathsf{CS}(x,y) = \frac{x \cdot y}{\|x\|_2 \|y\|_2} \in [-1,1], \tag{5.3}$$

- Jaccard similarity coefficient

$$\mathsf{J}(x,y) = \frac{|x \cap y|}{|x \cup y|} \in [0,1] \qquad \text{or} \qquad \mathsf{J}(a,b) = \frac{\sum_{k=1}^d \min(x_k,y_k)}{\sum_{k=1}^d \max(x_k,y_k)} \in [0,1], \tag{5.4}$$

  where x and y are thought as sets in the former case and as vectors with non-negative real values in the latter one,
- Minkowski similarity

$$\mathsf{MS}(x,y) = \frac{1}{1 + \|x-y\|_p} \in [0,1], \tag{5.5}$$

  where one gets the city block norm L_1 in the denominator if $p = 1$ and the Euclidean norm L_2 if $p = 2$.

The choice of w_S is usually unclear and is determined by author's preferences. Moreover, we are unaware of any systematic comparison of the above-mentioned measures for semantic similarity.

TABLE 4. Weight-based approaches with $\alpha = 0$ in (5.1)

Algorithm	Input for / Method of Clusterisation	Number of clusters as input/ Clusters overlap	Network size	Evaluation	Topology	Databases	Other attributed network clusterisation algorithms compared with
WNev [153]	**Weighted graph** MinCut [111] MajorClust [184] Spectral [107]	No/No	Small	Accuracy	Fixed	Synthetic	—
WSte1 [185]	**Weighted graph** Threshold	No/No	Large	Modularity	Fixed	Phone Network [136]	—
WSte2 [186]	**Similarity matrix (via Weighted graph and random walks)** Hierarchical clustering [70, 109]	No/No	Large	Modularity	Fixed	Phone Network [136]	—
WCom1 [44]	**Weighted graph** Weighted Louvain [21]	Yes/No	Small	Accuracy	Fixed	**DBLP***	**WCom2** [44] **DCom** [44]
WCom2 [44]	**Distance matrix (via weighted graph)** Hierarchical agglomerative clustering	Yes/No	Small	Accuracy	Fixed	**DBLP***	**WCom1** [44] **DCom** [44]
AA-Cluster [5, 6]	**Node embeddings (via weighted graph)** k-medoids	Yes/No	Small Medium Large	Density Entropy	Fixed	**Political Blogs** **DBLP*** **Patents*** Synthetic	**SA-Cluster** [231] **BAGC** [217] **CPIP** [133]
PWMA-MILP [9]	**Weighted graph** Linear programming MILP [9]	No/No	Small	RI NMI	Fixed	**WebKB**	—
KDComm [19]	**Weighted graph** Iterative Weighted Louvain	No/No	Small Medium Large	F-measure Jaccard measure Rank Entropy measure	Fixed	**ego-G+** **Twitter*** **DBLP*** [106] Reddit link	**CPIP** [133] **JCDC** [229] **UNCut** [224] **SI** [154]

Now let us describe the most influential weight-based methods **CODICIL** [172] and **SAC2** [54], according to Figure 3.

TABLE 5. Weight-based approaches with $\alpha \in [0,1]$ in (5.1)

Algorithm	α in (5.1)	Input for / Method of Clusterisation	Number of clusters as input/ Clusters overlap	Network size	Evaluation	Topology	Databases	Other attributed network clusterisation algorithms compared with
WWan [205]	$[0,1]$ in theory $\frac{1}{2}$ in experiments	**Edge similarity matrix (via weighted graph)** EdgeCluster [192] (k-means variant)	Yes	Small	NMI Micro-F1 Macro-F1	Non-fixed: removing edges	Synthetic BlogCatalog **Delicious***	Non-overlapping co-clustering [57]
SAC2 [54]	$[0,1]$	**kNN (unweighted) graph (via weighted graph)** (Unweighted) Louvain [21]	No/ No	Small Medium	Density Entropy	Non-fixed: removing edges	**Political Blogs Facebook100 DBLP10K**	**SAC1** [54] **WSte2** [186] Fast greedy [41] for weighted graph
WCru [50, 51]	$[0,1]$ in theory Not specified in experiments	**Weighted graph** Weighted Louvain [21]	No	Medium	Modularity Intracluster distance	Fixed	**Twitter***	—
CODICIL [172]	$[0,1]$ in theory $1/2$ in some experiments	**Weighted graph** Metis [112] Markov Clustering [177]	No	Small Medium Large	F-measure	Non-fixed: adding and removing edges	**CiteSeer* Flickr*** Wikipedia*	**Inc-Cluster** [232] **PCL-DC** [222] Link-PLSA-LDA [151]
WMen [141]	Not specified	**Weighted graph/Distance matrix for the weighted graph** SLPA [213] Weighted Louvain [21] K-medoids [226]	Yes-No/ Yes-No	Small Medium	NMI F-measure Accuracy	Fixed	**Lazega Research Consult** LFR benchmark [120]	**CODICIL** [172] **SA-Cluster** [231]
PLCA-MILP [9]	$[0,1]$	**Weighted graph** Linear programming MILP [9]	No/No	Small	RI NMI	Non-fixed: adding and removing edges	**WebKB**	**SCD** [131] **ASCD** [171] **SCI** [206] PCL-DC [222] Block-LDA [14]
kNN-enhance [106]	May be thought as $\alpha = 1/2$, kNN by semantics	**Distance matrix (of the augmented graph)** kNN k-means	No/No	Medium	Accuracy NMI F-Measure Modularity Entropy	Non-fixed: adding edges	**Cora Citeseer Sinanet PubMed Diabetes DBLP***	**PCL-DC** [222] PPL-DC [221] **PPSB-DC** [33] **CESNA** [220] **cohsMix** [227] **BAGC** [214] **GBAGC** [217]) **SA-Custer** [231] **Inc-Cluster** [232] **CODICIL** [172] GLFM [127])
IGC-CSM [152] source	$[0,1]$ in theory $1/2$ in comparison experiments	**Distance matrix for the weighted graph** k-Medoids	Yes/ No	Medium	Density Entropy	Fixed	**Political Blogs DBLP10K**	**SA-Cluster** [231] **SA-Cluster-Opt** [40]
AGPFC [93]	$[0,1]$ in theory, manually tuned in experiments	Fuzzy equivalent matrix λ-cut set method	No/Yes	Small Medium	Density Entropy	Fixed	**Political Blogs CiteSeer Cora WebKB**	**SA-Cluster** [231] **BAGC** [214]
NMLPA [98]	$1/2$	**Weighted graph** A multi-label propagation algorithm	Yes/ Yes	Medium	$F1$-score Jaccard Similarity	Fixed	**ego-Facebook Flickr*** [172] **ego-Twitter**	**CESNA** [220] **SCI** [206] **CDE** [129]

5.1.1. *CODICIL.* The method **CODICIL** [172] assigns semantic weights w_S between all the nodes in G, i.e. employs the non-fixed topology scheme. To decrease complexity, k nodes with highest cosine similarity values with v_i are selected as the top-k neighbours of v_i so that the semantic similarity w_S for v_i and v_j is essentially (5.3). The topological similarity weight w_T for two nodes is defined through the relative overlap of their respective structural neighbours. Approximations of (5.3) and (5.4) are used for this purpose. Then the topological and semantic weights are combined similar to (5.1). After that, a biased edge sampling procedure that retains edges being locally relevant to each node is applied (in other words, the edges with highest similarity values are retained) to make the weighted graph G_W sparse and enable both better runtime performance and lower memory usage in the subsequent community detection step that is performed by Metis [112] or Multi-layer Regularised Markov Clustering [177, 199]. The complexity of **CODICIL** is $O(n^2 \log n)$.

5.1.2. *SAC2.* The method **SAC2** [54] uses (5.2) as w_S for discrete attributes and (5.5) with $p = 2$ if they are continuous. Textual ones are first transformed into numeric values by TF-IDF procedure. Furthermore, the corresponding w_S is (5.3) or (5.4). The w_S obtained are then used to assign weights (5.1), where $w_T(e_{ij}) = 1$ if v_i and v_j are directly connected and 0, otherwise. After that, the weight W is used to construct an (unweighted) k-nearest neighbour graph $\tilde{G}$ as a directed graph in which each node has exactly k edges, connecting to its k most similar neighbours in G (thus the topology is non-fixed). The parameter k is set to equal to the average node degree in G. The version [60] of kNN algorithm with the empirical cost $O(n^{1.14})$ is applied to reduce complexity. At the community detection step, Louvain algorithm [21] is applied to find communities in $\tilde{G}$.

Remark 1. The authors of [17] consider attributed *multilayer* networks with different types of edges and use a similarity measure similar to (5.2) to flatten the network and put corresponding weights on single-layer network. After this fusion, any weighted graph clustering algorithm, e.g. Weighted Louvain [21], is actually suitable for community detection. In [162], the method called CAMIR is proposed for clustering attributed *multilayer* networks and assigns different weights to each attribute and edge-type. In particular, it ranks vertex properties by exploiting the information from edge-types and attributes and further constructs a unified similarity matrix (taking into account all edge types and attributes). The clusterisation step is performed via spectral clustering.

Remark 2. SANS [163] works with weighted *directed* graphs (that are out of scope of the present survey) using the matching coefficient (5.2) for semantic similarity and the so-called Weight Index (the sum of weights of incoming and outgoing edges) for topological similarity in a version of (5.1). SANS automatically determines the number of clusters via centroids and use the threshold algorithm for clustering.

Remark 3. Edge weighting similar to (5.1) is also applied in FocusCO [166]. Although it is not a purely unsupervised clustering approach (it requires user's preferences on focus attributes), it allows to solve simultaneously two interesting problems: the extraction of focused local clusters and the detection of outliers in an attributed network.

Remark 4. Let us mention that there exist approaches similar ideologically (attributes $\to$ edge weights $\to$ embeddings $\to$ k-means) but preceding to the recent algorithm **AA-Cluster** [5, 6]. For example, for a given network with numerical vector attributesrs GraphEncoder [194] and GraRep [31] first obtain edge weights (5.1) with $\alpha = 0$ and W_S being the cosine similarity (5.3) and then apply different techniques (sparse autoencoder in [194] and matrix factorization in [31] partly based on skip-gram [142] and DeepWalk [167] ideas) to obtain embeddings (low-dimensional vector representations) for the nodes of the weighted graph. The resulting embeddings are further fed to k-means algorithm to detect communities. However, in opposite to [5, 6], [194] and [31] mostly focus on embedding techniques suitable for a weighted graph and consider their different applications e.g. to classification and visualisation.

5.2. **Distance-based methods.** Methods considered in the precious subsection exchange the node attributes for edge weights so that one obtains a weighted graph with semantic information incorporated. Thus the topology of the network is somehow saved at the fusion step. Methods from this subsection intentionally remove the network so that the topological and semantic information is fused by a distance function between nodes and stored in a distance matrix, see Figure 6. Distance-based clusterisation methods such as k-means and k-medoids then can be applied. The user of such methods has to be aware of that in general the resulting clusters may contain disconnected portions of the initial graph as the graph structure is removed at the fusion step [5, Section 3.3].

The usual form of the distance fusion function is

$$D(v_i, v_j; \alpha) = \alpha d_T(v_i, v_j) + (1-\alpha) d_S(v_i, v_j), \qquad \alpha \in [0, 1], \tag{5.6}$$

where d_T and d_S is a *topological distance function* and a *semantic distance function* for nodes v_i and v_j, correspondingly. Clearly, one can introduce a more complicated fusion function based on distances. The parameter α influences the balance between topological and semantic information so that $\alpha = 1$ corresponds to the pure topological case and $\alpha = 0$ to the semantic one. It is common to define $d_T(v_i, v_j)$ as short path length distance between v_i and v_j. The possible options for d_S are as follows if we are given attribute vectors $x = (x_k)_{k=1}^d$ and $y = (y_k)_{k=1}^d$:

TABLE 6. Distance-based methods

Algorithm	α in (5.6)	Input for / Method of clusterisation	Number of clusters as input/Clusters overlap	Network size	Evaluation	Topology	Databases	Community detection methods for attributed graphs compared with
DCom [44]	$[0,1]$	**Distance matrix** Hierarchical agglomerative clustering	Yes/No	Small	Accuracy	Non-fixed: added edges	**DBLP***	**WCom1** [44] **WCom2** [44]
DVil [159, 201]	$[0,1]$	**Distance (or similarity) matrix** Stochastic kernel SOM algorithm [159, 201]	No/No	Small Medium	NMI	Non-fixed: added edges	Synthetic Medieval Notarial Deeds	—
SToC [15]	Formally $1/2$ but controlled via d_T and d_S	**Distance matrix** τ-close clustering [15]	No/No	Medium Large	Modularity Within-Cluster Sum of Squares	Non-fixed: added edges	**DBLP10K** DIRECTORS* DIRECTORS-gcc*	**Inc-Cluster** [232] **GBAGC** [217]
@NetGA [169]	$\alpha \in [0,1]$ in general $\alpha = 1/2$ in experiments	**Distance matrix** Genetic algorithm	No/No	Medium	NMI	Non-fixed: added edges	Synthetic	**SA-Cluster** [231] CSPA [62, 187]) **Selection** [62]
ANCA [65, 66]	Maybe thought as $\alpha = 1/2$ for summing eigenvectors of distance and similarity matrices	**Distance and similarity matrices** k-means for the sum of eigenvectors of the distance and similarity matrices	Yes/No	Medium	Adjusted Rand Index NMI Density Modularity Conductance Entropy	Fixed	Synthetic **DBLP10K** Anonymized Enron email corpus	**SA-Cluster** [231] **SAC1-SAC2** [54] **IGC-CSM** [152] **WSte1** [185] **ILouvain** [45].

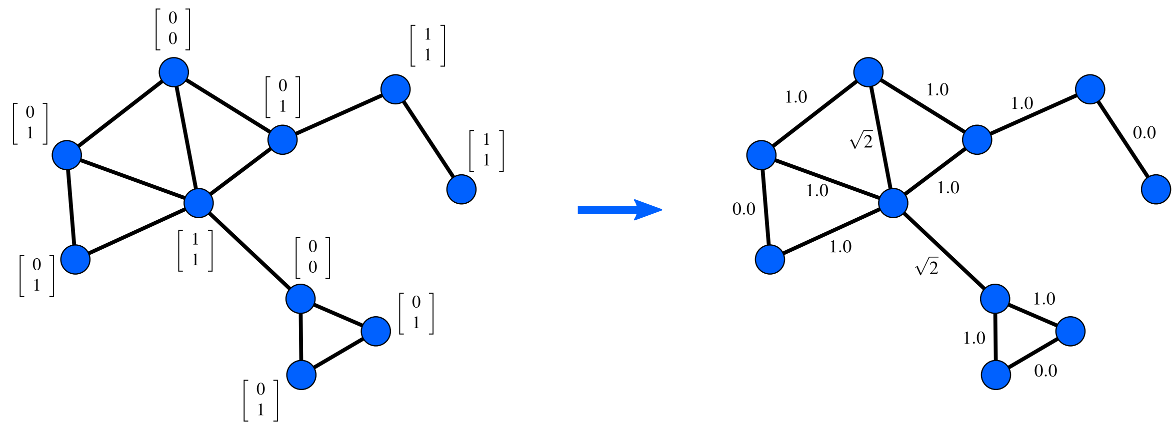

FIGURE 6. Distance-based approach (attributes distance is calculated with the Euclidean norm).

- Jaccard distance

$$\rho_{\mathrm{J}}(x,y) = 1 - \frac{|x \cap y|}{|x \cup y|} \in [0,1] \qquad \text{or} \qquad \rho_{\mathrm{J}}(a,b) = 1 - \frac{\sum_{k=1}^{d} \min(x_k,y_k)}{\sum_{k=1}^{d} \max(x_k,y_k)} \in [0,1], \tag{5.7}$$

  where x and y are thought to be sets in the former case and vectors with non-negative real values in the latter one,
- Minkowski distance

$$\rho_{\mathrm{M}}(x,y) = \|x-y\|_p, \tag{5.8}$$

  where one gets the city block norm L_1 if $p=1$ and the Euclidean norm L_2 if $p=2$.

The distance-based methods are summarised in Table 6. Note that **ANCA** [65, 66] employs a bit different approach than in (5.6) but nevertheless still deals with distance matrices (with respect to certain chosen seed-nodes).

There are no highly influential methods among the distance-based ones according to Figure 3, so we are going to describe the one most interesting to us.

5.2.1. *DVil.* In **DVil** [201], (5.6) with d_T and d_S being different *kernel functions* is used to combine semantic information of different types (graph, numerical variables, factors, textual variables, etc.) and network topology between all the nodes in the network. In one of the experiments, d_T is the shortest path length between two nodes. What is more, the topology is non-fixed there. The obtained distance matrix is then used in a stochastic kernel SOM[4] algorithm at the community detection step. The usage of SOM allows to simultaneously solve the problem of visualisation by projecting the nodes onto a grid of small dimension. The method **DVil** [201] is later developed in [159], where the balance between topological and semantic information is tuned automatically.

5.2.2. *SToC.* Semantic-Topological Clustering **SToC** [15] has time complexity $O(m \log n)$, where n and m are the number of nodes and edges in the network, respectively. **SToC** uses a fusing function different from (5.6), namely,

$$D(v_i, v_j; \alpha) = \max\{d_T(v_i, v_j), d_S(v_i, v_j)\}. \tag{5.9}$$

One can formally think that $\alpha = \frac{1}{2}$ in this scheme, however the impact of the semantic and topological components is still controlled by the parameters involved in d_T and d_S (see below).

The topological distance d_T in **SToC** is defined via (5.7) and the notion of l-neighbourhood:

$$d_T(v_i, v_j) = \rho_J(N_l(v_i), N_l(v_j)),$$

where the l-neighbourhood $N_l(v)$ of v is the set of nodes reachable from v with a path of length at most l (being a parameter), see [86]. To reduce complexity, the Jaccard distance in d_T is approximated with a bounded error $\varepsilon > 0$ (being a one more parameter) by bottom-k sketch vectors [42], i.e. compressed representations of l-neighbourhood in this case. The semantic distance d_S for quantitative attributes (normalised to $[0,1]$) is calculated using the Euclidean distance (5.8), and for categorical attributes using the Jaccard distance (5.7). The resulting distance D as in (5.9) is defined to be in $[0,1]$.

Using D, a cluster is defined by considering nodes that are within a maximum distance $\tau > 0$ from a given node. Namely, for a given threshold τ, a τ-close cluster C is a subset of the nodes in $\mathscr{V}$ such that there exists a node $v \in C$ such that for all $v' \in C$, $D(v, v') \leqslant \tau$. A τ-close clustering of G is defined as a partition of its nodes into τ-close clusters. At the clusterisation step, **SToC** iteratively extracts τ-close clusters from G starting from random seeds s (chosen through a select node function) by partial traversal of G. Take into account that C is contained in the set of nodes v such that $D(s, v) \leqslant \tau$. Nodes assigned to a cluster are not used in further iterations, thus the clusters formed are not overlapping. Moreover, the approach does not require the number of clusters as input.

As the choice of the parameters l and τ in **SToC** can be non-trivial, the authors propose an autotuning procedure. It computes optimal l and τ via approximating the cumulative distribution of d_S and d_T, taking into account parameters α_S and α_T, provided by the user and controlling the importance of semantic and topological component, respectively.

Remark 5. There exist distance-based methods for *multilayer* networks. For example, CLAMP (CLustering Attributed Multi-graPhs) [161] is an approach for clustering attributed networks with heterogeneous (numerical and categorical) attributes and multiple types of edges that uses a unified distance measure similar to (5.6), in a sense. The distance measure takes into account the importance of the node properties and the balance between the sets of attributes and edges, by assigning different weight to each of them. The clustering process adopts the gradient descent to produce fussy clusters. It is also worth mentioning that CLAMP is highly parallelisable.

5.3. **Node-augmented graph distance-based methods.** Methods from this class transform the initial graph G into another node-augmented graph G_A with new semantic nodes representing distinct node attributes, see Figure 7 and Table 7. Edges between structural and semantic nodes are added according to the node attributes in G (thus the topology is non-fixed). Take into account that the resulting graph G_A is much larger than G (especially if the dimension d and the sets of possible attribute values $dom(a_k)$ of node attributes A are large) and this extremely increases the time complexity of the methods.

According to Figure 3, **SA-Cluster** family is one of the most influential methods for community detection in attributed graphs and therefore we now give a short description of it.

[4] Throughout the text, SOM stand for self-organising maps.

TABLE 7. Node-augmented graph distance-based methods

Algorithm	Graph augmentation	Input for / Method of clusterisation	Number of clusters as input/Clusters overlap	Network size	Evaluation	Topology	Databases	Community detection methods for attributed graphs compared with
SA-Cluster [231] **Inc-Cluster** [39,232] **SA-Cluster-Opt** [40]	Semantic vertexes and structure-semantics edges	**Distance matrix (via neighbourhood random walks)** Modified k-medoids [231]	Yes/No	Small Medium	Density Entropy	Non-fixed: adding edges	**Political Blogs DBLP10K DBLP84K**	W-Cluster [231] (based on (5.6)) **SA-Cluster** [231] **Inc-Cluster** [39,232] **SA-Cluster-Opt**
SCMAG [101]	Semantic vertexes and structure-semantics edges	**Distance matrix (via neighbourhood random walks)** Subspace clustering algorithm based on ENCLUS [38]	No/Yes	Medium	Density Entropy	Non-fixed: adding edges	IMDB Arnetminer bibliography*	**SA-Custer** [231] GAMer [87]

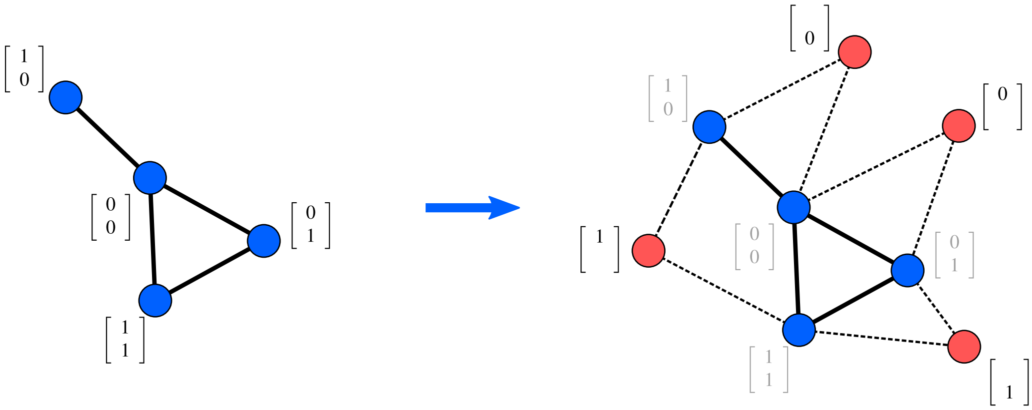

FIGURE 7. Distance approaches based on an node-augmented graph (attributes are converted to new attribute nodes thus forming an augmented graph together with the initial structural nodes).

5.3.1. *SA-Cluster.* The method **SA-Cluster** [231] transforms G into G_A with new attribute nodes $\tilde{v}_j^k$ representing distinct node attribute values. Namely, an attribute node $\tilde{v}_j^k$ represents an attribute-value pair (a_j, a_j^k). If $a_j^k \in dom(a_j)$ for v_i, then an attribute edge is added between v_i and $\tilde{v}_j^k$ (this however cannot be applied to continuous attributes). In G_A, two vertices $v_i, v_j \in G$ are close if they are connected through many structural and/or attribute edges. The neighbourhood random walk model is further used in **SA-Cluster** to estimate the node closeness in G_A.

To proceed, we recall several definitions from [231]. Let P be the (one-step) transition probability matrix of a graph. Given L as the length of a random walk, $c \in (0,1)$ as the restart probability, the *neighbourhood random walk distance* from v_i to v_j from the graph is defined as

$$\sum_{\tau: v_i \to v_j; \ell(\tau) \leqslant L} p(\tau) c (1-c)^{\ell(\tau)},$$

where τ is a path from v_i to v_j whose length is $\ell(\tau)$ with transition probability $p(\tau)$. Moreover,

$$R^L = \sum_{l=1}^{L} c(1-c)^l P^l,$$

where R is the *neighbourhood random walk distance matrix*. One can measure then the closeness between vertices v_i and v_j as

$$d(v_i, v_j) = R^l(i,j). \tag{5.10}$$

If $L \to \infty$, the neighbourhood random walk is the same as the random walk with restart defined in [196].

To combine the structural closeness and attribute similarity in G, [231] constructs the transition probability matrix of the graph G_A and computes the corresponding distances (5.10). Notice that at this step

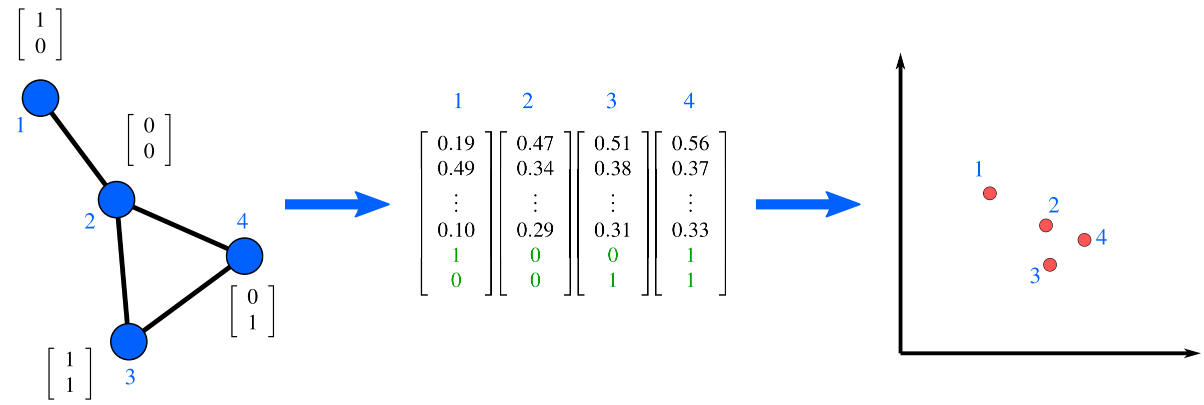

FIGURE 8. Embedding-based methods (in the simplest case, attribute vectors can be concatenated with the node embeddings and further fed to k-means).

weights on edges in G_A are assigned: topological edge has a weight of w_0, semantic edges corresponding to $a_1, a_2, \dots, a_d$ have an edge weight of $w_1, w_2, \dots, w_d$, respectively.

The resulting distance matrix, based on (5.10) for G_A, is then fed in a k-medoids type clustering algorithm. First, good initial centroids from the density point of view [95] are chosen. Furthermore, a converging iterative process for the optimisation of an objective function (in order to maximize intra-cluster similarity and minimize inter-cluster similarity) is performed, with the corresponding adjustments of the edge weights $w_0, \dots, w_d$.

As expected from the construction of G_A, **SA-Cluster** is computationally expensive, namely, its time complexity is $O(n^3)$. In order to improve the efficiency and scalability of **SA-Cluster**, the methods **Inc-Cluster** [39,232] and **SA-Cluster-Opt** [40] have been proposed. The main idea behind them is to reduce the number and the complexity of random walk distance computations.

5.4. **Embedding-based (early fusion) methods.** As is well-known, a graph as a traditional representation of a network brings several difficulties to network analysis. As mentioned in [53], graph algorithms suffer from high computational complexity, low parallelisability and inapplicability of machine learning methods. Novel network embedding techniques aim to tackle this by learning low-dimensional continuous vector representations (also known as *embeddings*) for the network nodes so that main network information is efficiently encoded[5]. Additionally, the embeddings not only aim at reconstructing the initial network but also at supporting network inference such as predicting links, classification and clustering nodes (for more details, see [30, 53]).

In the context of node-attributed social networks, the objective of network embedding is efficient low-dimensional encoding and combining both the network topology and semantics preserving proximities of different orders [31, 72, 191]. Having an embedding representation for the nodes, one can theoretically use traditional distance-based clusterisation methods such as k-means and k-medoids to further tackle the clusterisation problem, see Table 8.

Undoubtedly, there exists a rich bibliography on embedding techniques for networks with side information (node- and edge-attributed, heterogeneous in node and edge types) [30, 53] but in fact not all of them are reliable for the community detection task. It is worth mentioning that the task of classification (i.e. a supervised learning task) is typically considered. At the same time, some authors use embedding techniques for clusterisation in performance experiments that have been used only for classification in the original papers, e.g. in [72] the comparison is between attributed network embedding methods include TADW [218], LANE [99], GAE [114], VGAE [114], and GraphSAGE [90]. Taking all these fact into account, we confine ourselves in this survey only to the methods that work with node-attributed social

[5]The embedding approach is an algorithmic framework for learning continuous feature representations for nodes in networks, initially proposed as node2vec [78]. Node2vec learns a mapping of nodes to low-dimensional space of features by maximizing the likelihood of preserving network neighbourhoods of nodes. As a result, embeddings reflect the structural equivalence or homophily between network nodes [78].

or citation networks, have been applied to community detection and compared with other clusterisation methods.

TABLE 8. Embedding-based (early fusion) methods

Algorithm	Embeddings	Input for / Method of clusterisation	Number of clusters as input/Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
PLANE [122]	Via a generative model and EM [55]	**Node embeddings** *k*-means	Yes/No	Small Medium	Accuracy	**Cora***	Relational Topic Model [36]+Topic Distributions Embedding [105]
DANE [72]	Autoencoder	**Node Embeddings** *k*-means	Yes/No	Medium	Accuracy	**Cora** **Citeseer** **PubMed Diabetes** **Wiki**	Embeddings obtained via TADW [218] LANE [99] GAE [114] VGAE [114] GraphSAGE [90]
CDE [129]	Topology embedding matrix	**Topology embedding matrix and attribute matrix** Non-negative matrix factorisation	Yes/(Yes/No)	Small Medium	Accuracy NMI Jaccard similarity F1-score	**Cora** **Citeseer** **WebKB** **Flickr*** Philosophers [4] **ego-Facebook**	**PCL-DC** [222] **Circles** [125] **CESNA** [220] **SCI** [206]
MGAE [203]	Autoencoder	**Node embeddings** Spectral clustering	Yes/No	Medium	Accuracy NMI F-score Precision Recall Average Entropy Adjusted Rand Index	**Cora** **CiteSeer** **Wiki**	**Circles** [125] RTM [36] RMSC [211] Embeddings obtained via TADW [218] VGAE [114]

There are no highly influential methods among the embedding-based ones according to Figure 3 but we provide a short description of each one from Table 8 due to the novelty and importance of embedding techniques in the clusterisation task.

5.4.1. *PLANE.* Probabilistic LAtent Document Network Embedding **PLANE** [122] is a topic-based embedding method that aims to combine the following representations of each node with text attributes (e.g. in a citation network): the high-dimensional representations based on word occurrences and network topology, the representation in terms of a topic distribution (based on the Relational Topic Model [36]) and the low-dimension representation for nodes. The representations are joint through a generative model, with the estimation of the parameters (including the corresponding node embeddings) via the maximum a posteriori estimation with EM algorithm [55]. It is interesting that not only observed positive links are incorporated but also virtual negative ones. For each node, the authors form a 2-dimensional embedding to simultaneously solve the visualisation problem. To perform community detection, the embeddings obtained are fed to *k*-means.

5.4.2. *DANE.* Deep Attributed Network Embedding **DANE** [72] is an embedding-based algorithm using a deep model to preserve the first-order, high-order and semantic proximities in the attributed network. There are two branches composed of a multi-layer non-linear function and capturing the network topology and semantics with further mapping them into a low-dimensional space. Each branch is an auto-encoder, i.e. an unsupervised deep model widely used in machine learning [108]. The auto-encoders aim to minimize the reconstruction loss between the input vectors and the output embeddings to preserve the above-mentioned proximities. At that, the consistency and complementary of topology and semantics are preserved simultaneously at some point in order to obtain better structure-attribute fusion. Note that the loss function exploits an efficient most negative sampling strategy (with complexity $O(n^2)$). The resulting output is the concatenation of the embeddings obtained by each branch. Typically for the methods from this class, community detection is performed by *k*-means on the embeddings.

5.4.3. *CDE.* The method of **CDE** (Community Detection in attributed graphs: an Embedding approach) [129] uses a special function to measure community membership similarity. Its values further are input for a procedure based on skip-gram with negative sampling [142] to obtain a community structure embedding matrix that encodes the latent densely-connected subgraphs and explore inherent community structures. After this, the embedding matrix is used instead of the adjacency matrix for the network. Having the structure embedding matrix and attribute matrix at hand, the actual community detection is further performed via a nonnegative matrix factorization procedure (with a unified topology- and semantics-aware objective function) that optimizes community membership with suitable iterative updating rules based on Majorization-Minimization framework [4] (cf. [206]). The impact of topology and semantics may be varied. The resulting k communities (the number k is an input) may overlap or not depending on the community membership rule chosen.

5.4.4. *MGAE.* Marginalized Graph Autoencoder for Graph Clustering **MGAE** [203] takes an attributed graph as input and learns a topology and semantics with an augmented autoencoder upon them, with the graph convolutional network as a base. The authors propose to corrupt the semantics with noise with further marginalization in order to obtain a better representation from the autoencoder. By stacking multiple layers of the autoencoder, **MGAE** results in a deep representation for network nodes that is later fed into the spectral clustering algorithm.

Remark 6. Community detection in heterogeneous and multilayer networks is considered e.g. in [37, 102, 165]. Other embedding approaches for different heterogeneous networks (in particular, node-attributed) which are used mostly for classification but theoretically can be applied for clusterisation are discussed e.g. in the comprehensive surveys [30, 53].

TABLE 9. Pattern mining-based methods

Algorithm	Attribute types	Patterns	Number of clusters as input/Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
AHMotif [126]	Binary Numerical	Motif	Yes/No	Medium	NMI Accuracy	**Cora** **WebKB**	—

5.5. **Pattern mining-based (early fusion) methods.** Recall that a motif is a pattern of the interconnection occurring in real-world networks at numbers that are significantly higher than those in random networks [143] (note that a spanning tree pattern and a clique are representatives of motifs). Motifs are considered as building blocks for complex networks [143], and they may help to uncover useful information hidden in the network topology and semantics. We found just one community detection method for attributed social networks based on this idea, namely, **AHMotif** (Attribute Homogenous Motif-based method) [126], see Table 9. This method equips structural motifs identified for the network with the so-called homogeneity value based on attributes of the nodes involved in the motif. This information is then stored in a special adjacency matrix. Subsequently, the matrix is the input to the existing community detection algorithms such as Permanence [35] and Affinity Propagation [71].

6. SIMULTANEOUS FUSION METHODS

Oppositely to the early fusion methods, the simultaneous fusion ones use and fuse topology and semantics in a unified process with community detection. Some of them are based on modifications of known clusterisation algorithms such as Louvain, Normalised Cut, k-means, k-medoids and kNN, or attribute-aware adaptations of heuristic approaches such as evolutionary and genetic algorithms. A big subclass of simultaneous fusion methods use non-negative matrix factorisation framework to detect communities in attributed social networks, while another subclass — generative probabilistic models — aim to statistically infer a model of the attributed network under assumption that topology and semantics are generated accordingly to some parametric distributions.

6.1. **Methods modifying Louvain, Normalised Cut, k-means, k-medoids and kNN algorithms.** The list of the methods is given in Table 10. According to Figure 3, **SAC1** is one of the most influential methods for community detection in attributed social networks. Besides **SAC1**, we will also provide short descriptions of several other interesting methods.

TABLE 10. Methods that modify Louvain, Normalised Cut, k-means, k-medoids and kNN algorithms

Algorithm	Modified method	Number of clusters as input/ Clusters overlap	Network size	Evaluation	Databases	Other attributed network clusterisation methods compared with
OCru [51]	Louvain [21] Added attribute Entropy minimisation	No/No	Medium	Modularity Entropy	**Facebook100**	—
SAC1 [54]	Louvain [21] Added attribute similarity maximisation	No/ No	Small Medium	Density Entropy	**Political Blogs** **Facebook100** **DBLP10K**	**SAC2** [54] **WSte2** [186] Fast greedy [41] for weighted graph
I-Louvain [45] (code)	Louvain [21] Added maximisation of attribute-based measure Inertia	No/ No	Small Medium	NMI Accuracy	DBLP+Microsoft Academic Search* Synthetic	ToTeM [44][6]
LAA/LOA [11]	Louvain [21] Modularity gain depends on attributes	No/No	Small	Density Modularity	London gang [79] Italy gang Polbooks Adjnoun [155] Football [76]	—
UNCut [224]	Normalised Cut Added attribute homogenuity-aware measure Unimodality Compactness	Yes/No	Small Medium	NMI ARI	Disney [148] DFB [88] ARXIV [88] **Political Blogs** 4area [166] **Patents**	**SA-cluster** [231] SSCG [88] **NNM** [181]
DAEGC [202]	Graph attention network [200]+k-means for node embeddings+Stochastic Gradient Descent	Yes/No	Medium	ACC NMI F-measure ARI	**Cora** **Citeseer** **Pubmed**	RMSC [211] TADW [218] +k-means VGAE and GAE [114] +k-means
NetScan [64, 74]	An approximation algorithm for the connected k-Center optimization problem	Yes/Yes	Small Medium	Accuracy	Professors* Synthetic **DBLP*** BioGRID+Spellman	—
JointClust [147]	An approximation algorithm for the Connected X Clusters problem	No/No	Medium	Accuracy	**DBLP*** **CiteSeer*** Corel stock photo collection	—
MAM [176] (code)	Louvain-type algorithm with attribute-aware Modularity+Outlier detection	No/No	Small Medium Large	F1-score Attribute-aware Modularity	Synthetic Disney [148] DFB [88] ARXIV [88] IMDB [88] **DBLP*** **Patents*** Amazon [175]	CODA [73]
SS-Cluster [67]	k-Medoid based clustering algorithm with structural and attribute objective functions	Yes/No	Medium	Density Entropy	**Political Blogs** **DBLP10K**	**SA-cluster** [40, 231] W-cluster [40] kSNAP [195]
Adapt-SA [128]	Weighted k-means for d-dimensional representations of structure and attributes	Yes/No	Medium	Accuracy NMI F-measure Modularity Entropy	Synthetic **WebKB** **Cora** **Political Blogs** **CiteSeer** **DBLP10K**	**CODICIL** [172] **SA-Cluster** [230] **Inc-Cluster** [232] **PPSB-DC** [33] **PCL-DC** [222] **BAGC** [214]
kNAS [22]	kNN with added Semantic Similarity Score	Yes/Yes	Medium	Density Tanimoto Coefficient	**DBLP*** **Facebook*** **Twitter***	**SA-Cluster-Opt** [40] **CODICIL** [172] NISE [209]

6.1.1. *SAC1.* The method **SAC1** [54] is based on the modification of Newman's Modularity [41] for a given partition of G into k clusters:

$$Q_S = \sum_{l=1}^{k} \sum_{i \in C_l, j \in C_l} S(v_i, v_j),$$

where the normalised link strength $S(v_i, v_j)$ between nodes v_i and v_j is measured by comparing the existing network connection (v_i, v_j) with the expected number of connections $(d_i d_j)/2m$ (d_i is the degree of v_i). To deal with the attributes, **SAC1** uses the attribute modularity Q_A of a partition:

$$Q_A = \sum_{C} \sum_{i,j \in C} sim_A(v_i, v_j),$$

where sim_A is an attribute similarity function. As for the above-mentioned **SAC2** [54], if attributes are discrete, (5.2) is used for sim_A, while if they are continuous, (5.5) with $p = 2$ is applied. Textual ones are

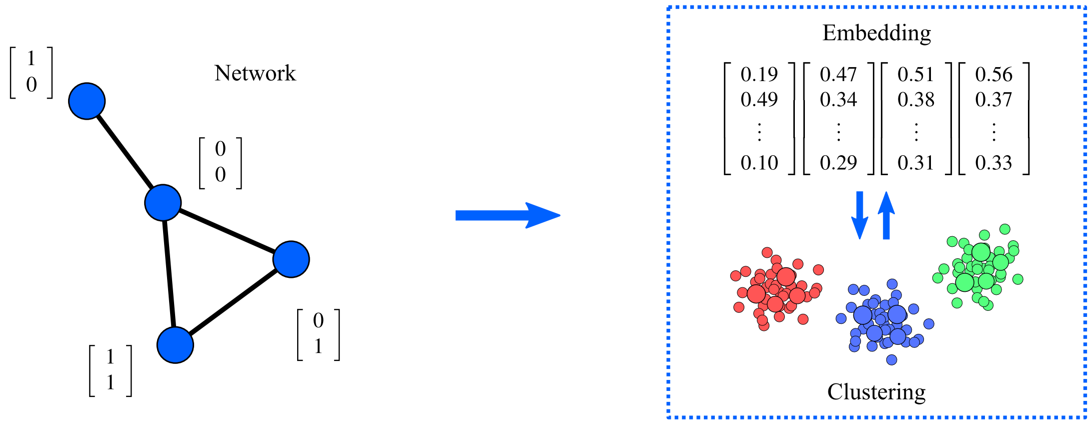

FIGURE 9. The scheme of **DAEGC** [202].

first transformed into numeric values by TF-IDF procedure. The similarity sim_A between the resulting representations is (5.3) or (5.4).

Next, a composite modularity is introduced as a weighted combination of structure modularity and attribute modularity

$$Q_C = \sum_C \sum_{i,j \in C} (\alpha S(v_i, v_j) + (1-\alpha)\, sim_A(v_i, v_j)), \qquad \alpha \in [0,1],$$

where α is a fusion parameter. This function is then maximised in a way similar to that in Louvain [21].

6.1.2. *I-Louvain.* The method **I-Louvain** [45] (source code and datasets) is based on a local optimization of a global criterion that includes Modularity Q [155] and a new measure called Inertia. The measure is defined by the sum of euclidean distances between attribute vectors and its centre of gravity, an average attribute vector over attribute vectors in the network. Using this notion, the authors define Inertia-based modularity $Q_{Inertia}$ for a partition that allows to compare, for each pair of elements from the same community, the expected distance with the observed distance between attributes. While Q considers the strength of the link between nodes in order to cluster strongly connected nodes, $Q_{Inertia}$ aims at clustering nodes whose attributes are the most similar. Like in [54], the community detection process consists in the optimisation of a linear combination of Q and $Q_{Inertia}$ similar to Louvain's.

6.1.3. *DAEGC.* Deep Attentional Embedded Graph Clustering **DAEGC** [202], in opposite to the embeddings-based early fusion methods, uses a goal-directed deep learning approach with a unified framework for producing embeddings and clustering. Namely, **DAEGC** fuses network topology and semantics via an attentional autoencoder (a variant of the graph attention network [200] taking into account high-order proximity) to obtain node embeddings. Furthermore, basing on the embeddings, soft labels are generated to guide a self-training graph clustering component. These two procedures are joint and performed iteratively to benefit both embedding and clusterisation quality. **DAEGC** produces k non-overlapping clusters where k is an input.

6.1.4. *kNAS.* The method **kNAS** [22] starts with identification of centroids for k clusters (k is an input) as nodes with high Local Outlier Factor [26] meaning that the node is core (low Local Outlier Factor refers to outliers). Initial clusters are formed by the kNN algorithm (i.e. topological similarity is achieved). Furthermore, the so-called Similarity Score responsible for nodes' semantic similarity within clusters is measured. Taking into account the Similarity Score obtained, the clusters are merged and centroids updated in a certain way. The process of achieving topological and semantic similarity repeats until the Similarity Score is maximized.

6.2. **Metaheuristic-based methods.** These methods adapt metaheuristic algorithms (in particular, evolutionary algorithms and tabu search) for optimisation of an objective function that quantifies the structural closeness and attribute homogeneity of an attributed network partition. The list of the methods is given in Table 11.

TABLE 11. Metaheuristic-based methods

Algorithm	Modified method	Number of clusters as input/ Clusters overlap	Network size	Evaluation	Databases	Other attributed network clusterisation methods compared with
MOEA-SA [130]	Multiobjective evolutionary algorithm (Modularity and Attribute Similarity are maximized)	No/No	Small Medium	Density Entropy	**Political Books** **Political Blogs** **Facebook100** **ego-Facebook**	**SAC1-SAC2** [54] **SA-Cluster** [231]
MOGA-@Net [168]	Multiobjective genetic algorithm (optimizing Modularity, Community score, Conductance, attribute similarity)	No/No	Small Medium	NMI Cumulative NMI Density Entropy	Synthetic **Cora** **Citeseer** **Political books** **Political Blogs** **ego-Facebook**	**SA-cluster** [231], **BAGC** [214] **OCru** [52] **Selection** [62] HGPA-CSPA [62, 187]
JCDC [229]	Tabu search and gradient ascent for a structure-attribute-aware loss function	Yes/No	Small Medium	NMI	Synthetic World trade network [158] **Lazega**	CASC [20] **CESNA** [220] **BAGS** [214]

6.3. **Non-negative matrix factorisation and matrix compression.** Non-negative matrix factorization (NMF) is a family of algorithms that aim to approximate a non-negative matrix with high rank by a product of non-negative matrices with lower ranks so that the approximation error by means of the Frobenius norm, denoted below by F, is minimal. As well known, NMF has an inherent clustering property, i.e. is able to find clusters in the input data [123]. The approximating product of matrices usually contains two factors but some algorithms [58] propose to include three or more. Often NMF is regularised (e.g. by a Lasso type conditions) to avoid bad behaviour of the approximating matrices.

As for node-attributed social networks, NMF requires a proper adaptation to fuse both topology and semantics and this has been done in several papers, see Table 12. To proceed, we need additional notation. In what follows, $\mathbf{S}_{n\times n}$ denotes the adjacency matrix for the initial network topology (as before, n is the number of nodes), $\mathbf{A}_{n\times d}$ the node attribute matrix for the initial network semantics ($d = \dim A$ is the dimension of attribute vector A), k the number of required clusters (it is an input in NMF approaches), $\mathbf{U}_{n\times k}$ the cluster membership matrix whose elements indicate the association of nodes with communities and finally $\mathbf{V}_{d\times k}$ denotes the cluster membership matrix whose elements indicate the association of the attributes with the communities. Other auxiliary matrices will be introduced below.

The general idea of NMF methods for attributed networks is to use known matrices $\mathbf{S}$, $\mathbf{A}$ and the number of clusters k in order to determine the unknown matrices $\mathbf{U}$ and $\mathbf{V}$ in an iterative optimisation procedure, and thus to obtain simultaneously a community partition and the corresponding semantic description for each community. Note that each element of normalised $\mathbf{U}$ and $\mathbf{V}$ in fact contains the probability of a node to belong to a particular community (communities may overlap in these settings). One can instead assign a node to the community with the highest probability to obtain non-overlapping communities [206].

"Matrix compression" technique will be discussed below while describing **PICS** algorithm [7].

Note that **SCI** [206] is one of the most influential methods for community detection in attributed social networks according to Picture 3. We will give a short description of **SCI** and several other NMF-based methods below.

6.3.1. *NPei.* The method **NPei** [164] uses a constrained nonnegative matrix tri-factorization framework [58] to cluster Twitter users and messages by fusing the relations between users (i.e. topology) and content (i.e. semantics). The initial point is a user-word-tweet tripartite network represented by several adjacency matrices similar to $\mathbf{S}$ and $\mathbf{A}$. The optimisation problem proposed by the authors includes however not only user-user, user-word and word-tweet adjacency matrices but also three types of network regularization [182] to model user similarity, message similarity and user interaction. The similarities are measured by a version of PageRank [160] based on the cosine similarity of messages and the adjacency matrix for users. The optimisation is further performed by an iterative update algorithm [58] to obtain user cluster matrix $\mathbf{U}$ and message cluster matrix $\mathbf{V}$. According to the authors, the complexity of their approach is $O(n^3)$ with respect to the number n of nodes in the network.

6.3.2. *SCI.* Semantic Community Identification **SCI** [206] adopts NMF for fusing topology and semantics as follows. The consistency in topology is modelled as $\min_{\mathbf{U}\geqslant 0} \|\mathbf{S}-\mathbf{U}\mathbf{U}^T\|_F^2$, while the consistency

TABLE 12. Non-negative matrix factorisation and matrix compression approaches

Algorithm	Factorisation/ compression type	Number of clusters as input / Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
NPei [164]	3-factor NMF	Yes/Yes	Small Medium	Purity	**Twitter** DBLP*	Relational Topic Model [36] (for documents)
3NCD [157]	2-factor NMF	Yes/Yes	Medium Large	F1-score Jaccard similarity	**ego-Facebook** **ego-Twitter** **ego-G+**	**CESNA** [220]
SCI [206]	2-factor NMF	Yes/Yes	Medium	ACC NMI GNMI F-measure Jaccard similarity	**Citeseer** **Cora** **WebKB** **LastFM**	**PCL-DC** [222] **CESNA** [220] **DCM** [170]
JWNMF [101]	2-factor NMF	Yes/Yes	Small Medium	Modularity Entropy NMI	Amazon Fail dataset Disney dataset Enron dataset DBLP-4AREA dataset **WebKB** **Citeseer** **Cora**	**BAGC** [214] **PICS** [7] SANS [163]
SCD [131]	2- and 3-factor NMF	Yes/Yes-No	Small Medium	Accuracy NMI	**Twitter** **WebKB**	**SCI** [206]
ASCD [171]	2-factor NMF	Yes/Yes-No	Small Medium	ACC NMI F-measure Jaccard similarity	**LastFM** **WebKB** **Cora** **Citeseer** **ego-Twitter*** **ego-Facebook***	Block-LDA [14] **PCL-DC** [222] **SCI** [206] **CESNA** [220] **Circles** [139]
CFOND [89]	2- and 3-factor NMF	Yes/(Yes/No)	Medium	Accuracy NMI	**Cora** **CiteSeer** **PubMed** **Attack** Synthetic	GNMF [29] DRCC [81] LP-NMTF [204] iTopicModel [188]
MVCNMF [92]	2-factor NMF	Yes/Yes	Small Medium	Density Entropy	**Political Blogs** **CiteSeer** **Cora** **WebKB** ICDM (DBLP*)	FCAN [97] SACTL [216] **kNAS** [22]
PICS [7]	Matrix compression (finding rectangular blocks)	No/No	Small Medium	Anecdotal and visual study	Youtube [144] **Twitter*** Phonecall [61] Device [61] **Political Books** (link) **Political Blogs**	—

in semantics as $\min_{\mathbf{V}\geqslant 0}\|\mathbf{U}-\mathbf{AV}\|_F^2$. The authors also propose to select the most relevant attributes for each community by adding an l_1 norm sparsity term to each column of matrix $\mathbf{V}$. This together with the models for topology and semantics leads to the following unified optimisation problem:

$$\min_{\mathbf{U}\geqslant 0,\mathbf{V}\geqslant 0}\left(\alpha_1\|\mathbf{S}-\mathbf{U}\mathbf{U}^T\|_F^2+\|\mathbf{U}-\mathbf{AV}\|_F^2+\alpha_2\sum\nolimits_j\|\mathbf{V}(\cdot,j)\|_1^2\right),$$

where $\alpha_1>0$ controls the topology impact and $\alpha_1\geqslant 0$ the sparsity penalty. Within **SCI**, a local minima is found by Majorization-Minimization framework [4]. In particular, the algorithm iteratively updates $\mathbf{U}$ with $\mathbf{V}$ fixed and then $\mathbf{V}$ with $\mathbf{U}$ fixed so that the process is guaranteed to converge. Note that, instead of using $\mathbf{U}$ directly, the authors consider $\mathbf{AV}$ as the final community membership matrix.

6.3.3. *JWNMF.* Joint Weighted Nonnegative Matrix Factorization method for clustering attributed graphs **JWNMF** [101] follows the same way to model topology as in [206] but with a weighted factorization for semantics, where the weights are automatically determined and updated to reduce the influence of uninformative attributes. Namely, a normalised diagonal matrix $\Lambda_{d\times d}$ is introduced to assign a weight for each attribute and to be further used in the approximation $\mathbf{A}\Lambda\approx\mathbf{U}\mathbf{V}^T$ by means of the F-norm, inspired by SymNMF [117]. The corresponding optimisation problem thus takes the form:

$$\min_{\mathbf{V},\mathbf{U},\Lambda\geqslant 0}\left(\|\mathbf{S}-\mathbf{U}\mathbf{U}^T\|_F^2+\alpha_1\|\mathbf{A}\Lambda-\mathbf{U}\mathbf{V}^T\|_F^2\right),$$

where $\alpha_1>0$ is the fusion parameter. The optimisation is performed iteratively [58]. Finally, a k-means variant is performed on $\mathbf{U}$ to identify k clusters. The complexity of **JWNMF** is $O(n^2)$.

6.3.4. *SCD.* The Semantic Community Detection method **SCD** [131] introduces an additional community relationship indicator matrix $\mathbf{R}_{k\times k}$ whose elements describe the relationships between the corresponding communities, and set regularisation condition on it that aim to ensure the consistency of the community structure with respect to topology and semantics. The optimisation problem obtained is

$$\min_{\mathbf{U},\mathbf{R},\mathbf{V}\geqslant 0}\left(\|\mathbf{S}-\mathbf{U}\mathbf{R}\mathbf{U}^T\|_F^2+\alpha_1\|\mathbf{U}-\mathbf{A}\mathbf{V}\|_F^2+\alpha_2\|\mathbf{R}-\mathbf{V}^T\mathbf{V}\|_F^2\right),$$

where $\alpha_{1,2}\geqslant 0$ are the fusion parameters. The problem is further solved iteratively [58].

6.3.5. *ASCD.* Adaptive Semantic Community Detection **ASCD** [171] follows the general line of NMF modelling discussed above but additionally employs an adaptive parameter to control the mismatch between topology and semantics components. According to the authors, the mismatch, i.e. the effect occurring when topology is not compatible with semantics, may happen for some networks and negatively affect the clustering performance (several their experiments confirm it). For this reason, they deal with the following optimisation problem

$$\min_{\mathbf{U},\mathbf{V}\geqslant 0}\left(\|\mathbf{S}-\mathbf{U}_t\mathbf{U}_t^T\|_F^2+f(\mathbf{U}_{t-1},\mathbf{V}_{t-1})\|\mathbf{U}_t-\mathbf{A}\mathbf{V}_t\|_F^2+\alpha_1\sum_j\|\mathbf{V}_t(\cdot,j)\|_1^2\right),$$

where t indicates the iteration number and f is the matching coefficient that controls the trade-off between topology and semantics according to the mismatch degree. There are two versions of f, namely, one is based on arctan functions and another on the NMI between the network topology and semantics. In particular, the former matching coefficient is defined as

$$f(\mathbf{U},\mathbf{V})=-\frac{2}{\pi}\arctan\left(\delta\sqrt{\frac{\|\mathbf{A}-\mathbf{U}\mathbf{V}^T\|_F^2}{dn}}\right),$$

where $\delta>0$ is a parameter. The optimisation problem is solved by the two-step block coordinate descent ($\mathbf{U}$ is updated while $\mathbf{V}$ is fixed, then vice versa).

Remark 7. In [104], NMF-based community detection in *multilayer* attributed networks is considered. Let us also mention the method from [137] that captures the complicated relationship between topology and semantics using a nonlinear projection function between the different cluster assignments for topology and semantics and adopts the positive unlabelled learning [132] to take the effect of partially observed positive edges into the cluster assignment.

6.3.6. *PICS.* The method **PICS** [7] (source) is a parameter-free algorithm that not only finds clusters but also detects anomalies and bridges. It is worth mentioning however that the nodes in a cluster found by **PICS** may be not necessarily densely connected due to the definition of clusters in [7]. As for the community detection process, **PICS** simultaneously "compresses" the network adjacency matrix $\mathbf{S}$ and the binary attribute matrix $\mathbf{A}$ by finding homogeneous rectangular blocks (considered further as clusters) of low and high densities in the matrices. The MDL principle [80], a criterion based on lossless compression principles, is adapted for this procedure.

6.4. **Pattern mining-based (simultaneous fusion) methods.** Pattern mining in attributed social networks focuses on fining and extraction of patterns, e.g. subsets of specific attributes or connections, in network topology and semantics [13]. This in turn helps to make sense of a network and to understand why the corresponding connections could be formed. Pattern mining methods for community detection typically use local patterns and optimisation criteria for finding informative communities not in the whole network but in its part only (e.g. [12, 170]). Note that there are many papers devoted to pattern and semantic subgraph mining in social networks (see the survey in [13]) but the majority of them do not deal with the task of community detection.

Community detection may be also based on cliques, according to a natural assumption that a community is a subset of well-connected nodes [24, 113]. Recall that in graph theory, a clique is a subset of nodes in an undirected graph such that every two nodes are adjacent, i.e. the corresponding subgraph is complete. A clique is called maximal if there is no other clique that contains it.

The list of the corresponding method is presented in Table 13.

Note that **DCM** [170] is a rather influential method according to Picture 3 and therefore we provide its main ideas below.

TABLE 13. Pattern mining-based (simultaneous fusion) methods

Algorithm	Patterns/Cliques	Number of clusters as input/Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
DCM [170]	Semantic patterns (*queries*)	Yes/Yes	Small Medium	Evaluation	**Delicious** **LastFM** **Flickr**	—
COMODO [12]	Semantic patterns	Yes/Yes	Small Medium	Description complexity Community size	BibSonomy [16] **Delicious** **LastFM**	**DCM** [170]
ACDC [113]	Maximal cliques	Yes/Yes	Medium	Density	**Political Blogs**	**SA-Cluster** [231] **SAC1-SAC2** [54]

6.4.1. *DCM.* Description-Driven Community Detection **DCM** [170] (code source) searchers for patterns in binary attributes to form k overlapping communities together with their proper descriptions. More precisely, each iteration of the algorithm consists of two steps aiming at reshaping the community via optimising first a certain structural quality function (the so-called community score based on local topology) and secondly a description complexity function that is based on mined concise patterns in attributes best describing the community (the patterns are called *queries*). Mining patterns is based on the ReMine algorithm [234] that recursively splits the data into the most informative patterns. The authors underline that **DCM** is able to grow communities starting from small seeds of nodes or from preliminary descriptions (depending on what information is available at the beginning). At the same time, **DCM** is not initially created for the complete coverage of the network.

Remark 8. In [18], ABACUS (frequent pAttern mining-BAsed Community discoverer in mUltidimensional networkS) is proposed to extract communities based on the extraction of patterns from *multi-layer* attributed social networks.

6.5. **Probabilistic model-based methods.** Methods from this class statistically infer a model of a clustered attributed network under the assumption that its structure and attributes are generated according to certain parametric distribution. The generative or stochastic block model are mainly used [9]. Note that it is a non-trivial task to properly choose a priori distributions for topology and semantics [5].

According to [222], there are many probabilistic models combining both topology and semantics: PHITS-PLSA combines PHITS with PLSA for community detection [43]), [63] combines LDA with LDA-Link for network analysis to have the LDA-Link-Word model, [151] combine the mixed membership stochastic block model with LDA, and extend the LDA-Link-Word model by separating the citing documents and cited documents with LDA-Link-Word model on the citing documents and PLSA model on the cited documents. However, the majority of the methods appeared before [222] focused on document clustering which is generally out of scope of the present survey. For this reason we consider only community detection methods published after the seminal paper [222], see Table 14. We will also describe **PCL-DC** [222], **BAGC** [214], **GBAGC** [217], **CESNA** [220] and **Circles** [139] as the most influential methods for community detection in attributed social networks according to Picture 3.

6.5.1. *PCL-DC.* The method **PCL-DC** (Popularity-based Conditional Link Model-Discriminative Content) [222] is based on a discriminative model of combining topology and semantics for community detection. It adapts a conditional model for network structure analysis taking into account the popularity of the nodes. The impact of irrelevant attributes is reduced by the usage of a discriminative content model where attributes are automatically assigned with proper weights, depending on their discriminative power. The above-mentioned models are further combined in a unified framework with the maximum likelihood inference performed in a two-stage EM-based optimization algorithm.

6.5.2. *BAGC-GBAGC.* The method **BAGC** (Bayesian Attributed Graph Clustering) [214] employs a Bayesian probabilistic model for detecting non-overlapping communities in networks with categorical attributes. **BAGC** uses a generative process similar to that in **CohsMix** [227], in particular, community labels for the nodes are modelled via a multinomial distribution independently, then attributes are modelled by a multinomial distribution and edges by a Bernoulli one basing on the labels modelled earlier. However, oppositely to **CohsMix** [227], **BAGC** works with categorical attributes and does not treat the parameters of distributions as fixed values. More precisely, **BAGC** takes a Bayesian treatment on the

TABLE 14. Probabilistic model-based methods

Algorithm	Model features	Number of clusters as input/Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
PCL-DC [222]	Conditional Link Model Discriminative Content model	Yes/No	Medium	NMI Pairwise F-measure Modularity Normalized cut	**Cora** **Siteseer**	PHITS-PLSA [43] LDA-Link-Word [63] Link-Content-Factorization [233]
CohsMix [227]	MixNet model [183]	Yes/No	Small	Rand Index	Synthetic Exalead.com search engine dataset	Multiple view learning [228] Hidden Markov Random Field [10]
BAGC [214] **GBAGC** [217]	a Bayesian treatment on distribution parameters	Yes/No	Medium	Modularity Entropy	**Political Blogs** **DBLP10K** **DBLP84K**	**Inc-Cluster** [232] **PICS** [7]
VEM-BAGC [32]	Based on **BAGC** [214]	Yes/No	Medium	Modularity Entropy	**Political Blogs** Synthetic networks	**BAGC** [214]
PPSB-DC [33]	Popularity-productivity stochastic block model and discriminative content model	Yes/No	Medium	normalized mutual information (NMI) Pairwise F measure (PWF) Accuracy	**Cora** **CiteSeer** **WebKB**	**PCL-DC** [222] PPL-DC [221]
CESNA [220]	A probabilistic generative model assuming communities generate network structure and attributes	No/Yes	Medium Large	Evaluation	**ego-Facebook** **ego-G+** **ego-Twitter** **Wikipedia* (philosophers)** **Flickr**	**CODICIL** [172] **Circles** [139] Block-LDA [14]
Circles [139]	A generative model for friendships in social circles	Yes/Yes	Medium Large	Balanced Error Rate	**ego-Facebook** **ego-G+** **ego-Twitter**	Block-LDA [14] Adapted Low-Rank Embedding [225]
SI [154]	A modified version of a stochastic block model [96]	Yes/No	Small Medium	Normalized mutual information (NMI)	Synthetic High school friendship network Food web of marine species in the Weddell Sea Harvard Facebook friendship network malaria HVR 5 and 6 gene recombination network	—
NEMBP [94]	A generative model with learning method using a nested EM algorithm with belief propagation	Yes/(Yes/No)	Small Medium	Accuracy NMI GNMI F-score Jaccard	**WebKB** **ego-Twitter*** **ego-Facebook*** **CiteSeer** **Cora** **Wikipedia*** **Pubmed**	Block-LDA [14] **PCL-DC** [222] **CESNA** [220] **DCM** [170] **SCI** [206]
NBAGC-FABAGC [215]	A nonparametric and asymptotic Bayesian model selection method based on **BAGC** [214]	No/No	Medium	NMI Modularity Entropy	Synthetic **Political Blogs** **DBLP10K** **DBLP84K**	**PICS** [7]

parameters and thus considers all their possible values that leads, according to the authors, to better community detection quality. The probabilistic inference is further performed by the variational approach from [110] together with a certain approximating procedure. **GBAGC** (General Bayesian framework to Attributed Graph Clustering) [217], a generalisation of **BAGC** for weighted attribute networks, is further proposed by the same authors.

6.5.3. *CESNA.* The method **CESNA** (Communities from Edge Structure and Node Attributes) [220] simultaneously uses the probabilistic generative model of BIGCLAM [219] for generating connections and the logistic model for attributes to infer the distribution of community memberships. The resulting communities are overlapping. Furthermore, a block-coordinate ascent method is used to update all model parameters in $O(m)$-time, where $m = |E|$, that makes **CESNA** robust for large attributed networks.

6.5.4. *Circles.* The method **Circles** [139] detects users' social circles in attributed user's ego networks via a multimembership node clustering. Its generative model is based on hard assignment of a node to multiple circles and learns the circle-specific user profile similarity metric. To maximize the corresponding likelihood, the coordinate ascent by [135] is used.

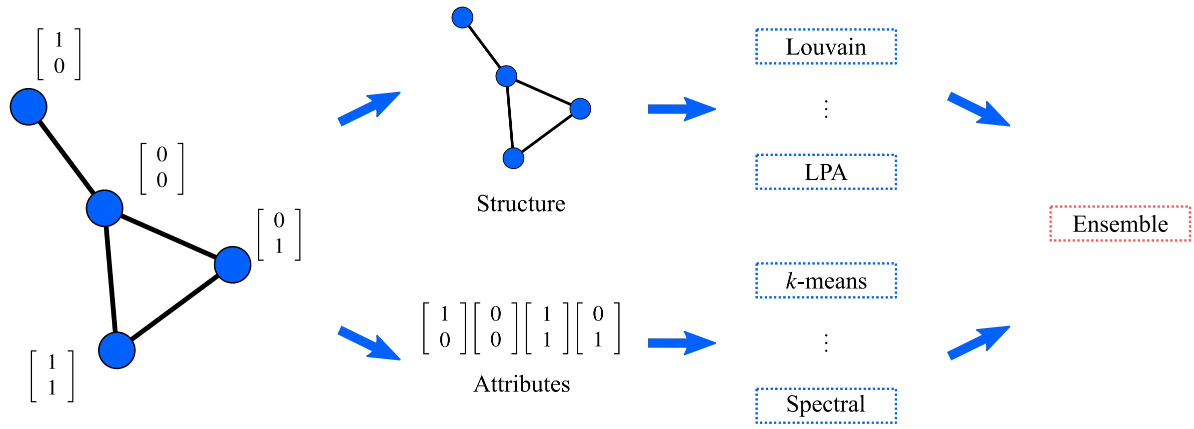

FIGURE 10. The scheme of late fusion methods.

Remark 9. TUCM (Topic User Community Model) [173] proposer generative Bayesian models for detecting overlapping communities in *multilayer* attributed networks where different types of interactions between users are possible.

6.6. **Dynamical system-based and agent-based methods.** Methods from this class treat a network as a dynamic system and assume that its community structure is a consequence of certain interactions among nodes, see Table 15. Some methods assume that the interactions occur in an information propagation process, i.e. while information is sent to or received from every node. Others comprehend each node as an autonomous agent and develop a multiagent system to detect communities. In fact, these methods are not among the most influential in Picture 3 but this is probably due to their novelty. In any case, these contemporary approaches seem to be very efficient for large attributed social networks as can be easily parallelised.

TABLE 15. Dynamical system-based and agent-based methods

Algorithm	Description	Number of clusters as input / Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
CPIP-CPRW [133]	Content (information) propagation models: a linear approximate model of influence propagation (**CPIP**) and content propagation with the random walk principle (**CPRW**)	Yes/Yes	Medium	F-score, Jaccard Similarity, Normalized Mutual Information (NMI)	**CiteSeer** **Cora** **ego-Facebook** **PubMed Diabetes**	Adamic Adar [1] **PCL-DC** [222] **Circles** [125] **CODICIL** [172] **CESNA** [220]
CAMAS [28]	Each node with attributes as an autonomous agent with influence in a cluster-aware multiagent system	No/Yes	Medium Large	Coverage Rate Normalized Tightness Normalized Homogeneity F1-Score Jaccard Adjusted Rand Index	Synthetic **ego-Facebook** **ego-Twitter*** **ego-G+**	**CESNA** [220] EDCAR [85]
SLA [27]	A dynamic cluster formation game played by all nodes and clusters in a discrete-time dynamical system	Yes/No	Medium Large	Density Entropy F1-score	**Delicious** **LastFM** **ego-Facebook** **ego-Twitter*** **ego-G+**	**CESNA** [220] EDCAR [85]

7. LATE FUSION METHODS

Late fusion methods intend to fuse topology and semantics after the clusterisation step. Usually clusterings produced separately for topological (e.g. by the Louvain method [21]) and semantic (e.g. by k-means [91]) information are further fused via consensus (ensemble-based) clustering techniques [82, 118, 187, 189, 190].

7.1. **Consensus-based methods.** Given an ensemble of clusterings, the goal is to perform a consensus clustering, i.e., a single, prototypical clustering solution that optimizes a certain objective function properly defined over information available from the clusterings in the ensemble. A recent survey on general-purpose ensemble-based clustering methods can be found in [23]. Besides the above-mentioned general-purpose approaches that actually have not been compared with the methods discussed in this survey, we could only find the ones in Table 16 that particularly focus on community detection in attributed social networks. Definitely, further study in this direction and comparison with other attributed network clustering methods is necessary.

According to Picture 3, there are no consensus-based methods among the most influential for community detection in attributed social networks but we will nevertheless provide some details on some of them, namely, **Selection** [62] and **WCMFA** [134].

7.1.1. *Selection.* The **Selection** method [62] switches from topology-based to semantics-based clustering when the graph structure is ambiguous. More precisely, the method relies on topology-based clusters when the so-called estimated mixing parameter

$$\mu = \frac{1}{n}\sum_{i=1}^{n}\frac{d_i^{\mathsf{ext}}}{d_i}, \qquad (d_i^{\mathsf{ext}} \text{ is the external degree of and } d_i \text{ the degree of } v_i),$$

for the topology-based clustering is less then the experimental value of the mixing parameter μ_{limit} in LFR benchmark with ground truth [120] when the NMI corresponding to the topology-based method significantly drops (the graph structure is then called ambiguous). For instance, for the Louvain method $\mu_{limit} \approx 0.6$ as shown in [62, 120]. If $\mu \geqslant \mu_{limit}$, then the semantics-based clustering (obtained e.g. by k-means) is used. The performance of **Selection** is particularly compared with that of HGPA (HyperGraph Partitioning Algorithm) and CSPA (Cluster-based Similarity Partitioning Algorithm), general-purposed ensemble clustering methods from [187], in combining the topology-based Louvain clustering and the semantics-based k-means clustering of the network. It is observed that **Selection** is able to outperform (in some sense) the tested methods by switching from the Louvain clustering to the k-means one.

7.1.2. *WCMFA.* The Weighted Co-association Matrix-based Fusion Algorithm **WCMFA** [134] takes as an input an ensemble of several clusterings based separately on topology and semantics with weights depending on topological and semantic similarity of the initial nodes. Furthermore, a weighted co-association matrix is constructed so that the co-occurrence of two nodes in the same cluster and the degree of its similarity, if the pair is indeed in the same cluster, is taken into account. The matrix is then treated as a similarity matrix for the node set that can be input for Single Link, Complete Link or Average Link clustering algorithms to find a consensus community structure.

Remark 10. We refer the interested reader to [82, 189, 190] for general-purpose clustering methods for *multilayer* networks.

TABLE 16. Consensus-based methods

Algorithm	Combining the partitions	Number of clusters as input / Clusters overlap	Network size	Evaluation	Databases	Community detection methods for attributed graphs compared with
LCru [49]	Row-manipulation in the contingency matrix for the clusterings	No/No	Small Medium	ARI Density Entropy	Facebook* **DBLP10K**	—
Selection [62]	Switching between the clusterings	Depends on the partitions	Medium	NMI Modularity	Synthetic LFR benchmark [120] **DBLP84K**	**BAGC** [214] **OCru** [51] **SA-Cluster** [231] HGPA-CSPA [187]
Multiplex [100]	Multiplex representation scheme (attributes and structure are clustered separately as layers and then combined via consensus [193])	No/Yes	Medium Large	F1-score	Synthetic **ego-Twitter** **ego-Facebook** **ego-G+**	**CESNA** [220] **3NCD** [157]
WCMFA [134]	Association matrix with weighting based on topology and semantics similarity	Depends on the partitions	Small	Rand index Adjusted RI NMI	**Consult** [48] London Gang [79] Montreal Gang [56]	**WMen** [141]

8. Conclusion

It is shown in the survey that there exist a large amount of methods for community detection in node-attribute social networks based on different fusion techniques. In particular, 77 methods are grouped and analysed using the proposed classification criterion and much more are mentioned as relative to the topic under consideration. Moreover, we indicated the most influential methods and gave their short descriptions.

According to our analysis, several essential problems exist in the area. For example, an comprehensive comparative study is an emergency problem as the existing partial contributions to this do not allow to see the overall picture of methods' community detection quality. Figure 2 confirms the fact that the method-method comparison graph is very sparse. What is more, even if some methods are compared with others, it does not generally mean that one *purely* outperforms another as hyperparameters of the methods under consideration are usually not tuned for a particular task. Moreover, different authors use for experiments different datasets and quality metrics and this does not add clarity to the question. Another issue is that many authors are just unaware of the state-of-the-art methods and continue to compare their approaches with rather inefficient pioneering methods.

We hope that our survey is an important step in resolving the above-mentioned methodological and experimental problems.

9. Competing Interests Statement

There are no competing interests in publication of this survey paper.

10. Acknowledgements

This research is financially supported by Russian Science Foundation, Agreement 17-71-30029 with co-financing of Bank Saint Petersburg.

References

[1] Lada A Adamic and Eytan Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211 – 230, 2003.
[2] Lada A. Adamic and Natalie Glance. The political blogosphere and the 2004 u.s. election: Divided they blog. In *Proceedings of the 3rd International Workshop on Link Discovery*, LinkKDD '05, pages 36–43, New York, NY, USA, 2005. ACM.
[3] Charu C. Aggarwal and ChengXiang Zhai. *A Survey of Text Clustering Algorithms*, pages 77–128. Springer US, Boston, MA, 2012.
[4] Yong-Yeol Ahn, James P. Bagrow, and Sune Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466:761–764, 2010.
[5] Esra Akbas and Peixiang Zhao. Attributed graph clustering: An attribute-aware graph embedding approach. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, ASONAM '17, pages 305–308, New York, NY, USA, 2017. ACM.
[6] Esra Akbas and Peixiang Zhao. *Graph Clustering Based on Attribute-Aware Graph Embedding*, pages 109–131. Springer International Publishing, Cham, 2019.
[7] Leman Akoglu, Hanghang Tong, Brendan Meeder, and Christos Faloutsos. Pics: Parameter-free identification of cohesive subgroups in large attributed graphs. In *Proceedings of the 12th SIAM International Conference on Data Mining, SDM 2012*, pages 439–450, 2012.
[8] Andry Alamsyah, Budi Rahardjo, and Kuspriyanto. Community detection methods in social network analysis. *Advanced Science Letters*, 20(1):250–253, 2014.
[9] Esmaeil Alinezhad, Babak Teimourpour, Mohammad Mehdi Sepehri, and Mehrdad Kargari. Community detection in attributed networks considering both structural and attribute similarities: two mathematical programming approaches. *Neural Computing and Applications*, Feb 2019.
[10] C. Ambroise, M. Dang, and G. Govaert. Clustering of spatial data by the em algorithm. In Amílcar Soares, Jaime Gómez-Hernandez, and Roland Froidevaux, editors, *geoENV I — Geostatistics for Environmental Applications*, pages 493–504, Dordrecht, 1997. Springer Netherlands.
[11] Yousra Asim, Rubina Ghazal, Wajeeha Naeem, Abdul Majeed, Basit Raza, and Ahmad Kamran Malik. Community detection in networks using node attributes and modularity. *International Journal of Advanced Computer Science and Applications*, 8(1), 2017.
[12] Martin Atzmueller, Stephan Doerfel, and Folke Mitzlaff. Description-oriented community detection using exhaustive subgroup discovery. *Information Sciences*, 329:965 – 984, 2016. Special issue on Discovery Science.
[13] Martin Atzmueller, Henry Soldano, Guillaume Santini, and Dominique Bouthinon. Minerlsd: efficient mining of local patterns on attributed networks. *Applied Network Science*, 4(1):43, Jun 2019.

[14] Ramnath Balasubramanyanm and William W. Cohen. *Block-LDA: Jointly modeling entity-annotated text and entity-entity links*, pages 450–461. 2011.

[15] Alessandro Baroni, Alessio Conte, Maurizio Patrignani, and Salvatore Ruggieri. Efficiently clustering very large attributed graphs. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, ASONAM '17, pages 369–376, New York, NY, USA, 2017. ACM.

[16] Dominik Benz, Andreas Hotho, Robert Jäschke, Beate Krause, Folke Mitzlaff, Christoph Schmitz, and Gerd Stumme. The social bookmark and publication management system bibsonomy. *The VLDB Journal*, 19(6):849–875, Dec 2010.

[17] M. Berlingerio, M. Coscia, and F. Giannotti. Finding and characterizing communities in multidimensional networks. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 490–494, July 2011.

[18] 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, Nov 2013.

[19] Shreyansh Bhatt, Swati Padhee, Amit Sheth, Keke Chen, Valerie Shalin, Derek Doran, and Brandon Minnery. Knowledge graph enhanced community detection and characterization. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, pages 51–59, New York, NY, USA, 2019. ACM.

[20] N. Binkiewicz, J. T. Vogelstein, and K. Rohe. Covariate-assisted spectral clustering. *Biometrika*, 104(2):361–377, 03 2017.

[21] 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.

[22] M. Parimala Boobalan, Daphne Lopez, and X.Z. Gao. Graph clustering using k-neighbourhood attribute structural similarity. *Appl. Soft Comput.*, 47(C):216–223, October 2016.

[23] Tossapon Boongoen and Natthakan Iam-On. Cluster ensembles: A survey of approaches with recent extensions and applications. *Computer Science Review*, 28:1 – 25, 2018.

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

[25] Oualid Boutemine and Mohamed Bouguessa. Mining community structures in multidimensional networks. *ACM Trans. Knowl. Discov. Data*, 11(4):51:1–51:36, June 2017.

[26] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, SIGMOD '00, pages 93–104, New York, NY, USA, 2000. ACM.

[27] Z. Bu, H. Li, J. Cao, Z. Wang, and G. Gao. Dynamic cluster formation game for attributed graph clustering. *IEEE Transactions on Cybernetics*, 49(1):328–341, Jan 2019.

[28] Zhan Bu, Guangliang Gao, Hui-Jia Li, and Jie Cao. Camas: A cluster-aware multiagent system for attributed graph clustering. *Information Fusion*, 37:10 – 21, 2017.

[29] Deng Cai, Xiaofei He, Xiaoyun Wu, and Jiawei Han. Non-negative matrix factorization on manifold. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 63–72, Washington, DC, USA, 2008. IEEE Computer Society.

[30] H. Cai, V. W. Zheng, and K. C. Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1616–1637, Sep. 2018.

[31] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 891–900, New York, NY, USA, 2015. ACM.

[32] Xiangyong Cao, Xiangyu Chang, and Zongben Xu. Community detection for clustered attributed graphs via a variational em algorithm. In *Proceedings of the 2014 International Conference on Big Data Science and Computing*, BigDataScience '14, pages 28:1–28:1, New York, NY, USA, 2014. ACM.

[33] Bian-fang Chai, Jian Yu, Cai-yan Jia, Tian-bao Yang, and Ya-wen Jiang. Combining a popularity-productivity stochastic block model with a discriminative-content model for general structure detection. *Phys. Rev. E*, 88:012807, Jul 2013.

[34] Tanmoy Chakraborty, Ayushi Dalmia, Animesh Mukherjee, and Niloy Ganguly. Metrics for community analysis: A survey. *ACM Comput. Surv.*, 50(4):54:1–54:37, August 2017.

[35] Tanmoy Chakraborty, Sriram Srinivasan, Niloy Ganguly, Animesh Mukherjee, and Sanjukta Bhowmick. On the permanence of vertices in network communities. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1396–1405, New York, NY, USA, 2014. ACM.

[36] Jonathan Chang and David Blei. Relational topic models for document networks. In David van Dyk and Max Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 81–88, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR.

[37] Shiyu Chang, Wei Han, Jiliang Tang, Guo-Jun Qi, Charu C. Aggarwal, and Thomas S. Huang. Heterogeneous network embedding via deep architectures. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 119–128, New York, NY, USA, 2015. ACM.

[38] Chun-Hung Cheng, Ada Waichee Fu, and Yi Zhang. Entropy-based subspace clustering for mining numerical data. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, pages 84–93, New York, NY, USA, 1999. ACM.

[39] Hong Cheng, Yang Zhou, Xin Huang, and Jeffrey Xu Yu. Clustering large attributed information networks: an efficient incremental computing approach. *Data Mining and Knowledge Discovery*, 25(3):450–477, Nov 2012.

[40] Hong Cheng, Yang Zhou, and Jeffrey Xu Yu. Clustering large attributed graphs: A balance between structural and attribute similarities. *ACM Trans. Knowl. Discov. Data*, 5(2):12:1–12:33, February 2011.
[41] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70:066111, Dec 2004.
[42] Edith Cohen and Haim Kaplan. Summarizing data using bottom-k sketches. In *Proceedings of the Twenty-sixth Annual ACM Symposium on Principles of Distributed Computing*, PODC '07, pages 225–234, New York, NY, USA, 2007. ACM.
[43] David A. Cohn and Thomas Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 430–436. MIT Press, 2001.
[44] David Combe, Christine Largeron, Elod Egyed-Zsigmond, and Mathias Gery. Combining relations and text in scientific network clustering. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, ASONAM '12, pages 1248–1253, Washington, DC, USA, 2012. IEEE Computer Society.
[45] David Combe, Christine Largeron, Mathias Géry, and Előd Egyed-Zsigmond. I-louvain: An attributed graph clustering method. In Elisa Fromont, Tijl De Bie, and Matthijs van Leeuwen, editors, *Advances in Intelligent Data Analysis XIV*, pages 181–192, Cham, 2015. Springer International Publishing.
[46] Michele Coscia, Fosca Giannotti, and Dino Pedreschi. A classification for community discovery methods in complex networks. *Stat. Anal. Data Min.*, 4(5):512–546, October 2011.
[47] Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew McCallum, Tom Mitchell, Kamal Nigam, and Sean Slattery. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the fifteenth national/tenth conference on artificial intelligence/innovative applications of artificial intelligence*, pages 509–516, Menlo Park, 1998. American Association for Artificial Intelligence.
[48] R. Cross and A. Parker. *The Hidden Power of Social Networks*. Harvard Business School Press, Boston, MA, USA, 2004.
[49] J. D. Cruz and C. Bothorel. Information integration for detecting communities in attributed graphs. In *2013 Fifth International Conference on Computational Aspects of Social Networks*, pages 62–67, Aug 2013.
[50] Juan Cruz, Cécile Bothorel, and François Poulet. Détection et visualisation des communautés dans les réseaux sociaux. *Revue d'intelligence artificielle*, 26:369 – 392, 08 2012.
[51] Juan David Cruz Gomes, Cécile Bothorel, and François Poulet. Semantic clustering of social networks using points of view. In *CORIA: conférence en recherche d'information et applications 2011*, Avignon, France, March 2011.
[52] Juan David Cruz Gomez, Cécile Bothorel, and François Poulet. Entropy based community detection in augmented social networks. In *International Conference on Computational Aspects of Social Networks*, pages 163–168, Salamanca, Spain, October 2011.
[53] P. Cui, X. Wang, J. Pei, and W. Zhu. A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*, 31(5):833–852, May 2019.
[54] The Anh Dang and Emmanuel Viennet. Community detection based on structural and attribute similarities. In *International Conference on Digital Society (ICDS)*, pages 7–14, January 2012. ISBN: 978-1-61208-176-2. Best paper award.
[55] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
[56] Karine Descormiers and Carlo Morselli. Alliances, conflicts, and contradictions in montreal's street gang landscape. *International Criminal Justice Review*, 21(3):297–314, 2011.
[57] Inderjit S. Dhillon, Subramanyam Mallela, and Dharmendra S. Modha. Information-theoretic co-clustering. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, pages 89–98, New York, NY, USA, 2003. ACM.
[58] Chris Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 126–135, New York, NY, USA, 2006. ACM.
[59] Ying Ding. Community detection: Topological vs. topical. *Journal of Informetrics*, 5(4):498–514, 2011.
[60] Wei Dong, Charikar Moses, and Kai Li. Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 577–586, New York, NY, USA, 2011. ACM.
[61] Nathan Eagle, Alex (Sandy) Pentland, and David Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274–15278, 2009.
[62] Haithum Elhadi and Gady Agam. Structure and attributes community detection: Comparative analysis of composite, ensemble and selection methods. In *Proceedings of the 7th Workshop on Social Network Mining and Analysis*, SNAKDD '13, pages 10:1–10:7, New York, NY, USA, 2013. ACM.
[63] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5220–5227, 2004.
[64] Martin Ester, Rong Ge, Byron J. Gao, Zengjian Hu, and Boaz Ben-Moshe. Joint cluster analysis of attribute data and relationship data: the connected k-center problem. In *SDM*, 2006.
[65] Issam Falih, Nistor Grozavu, Rushed Kanawati, and Younès Bennani. Anca : Attributed network clustering algorithm. In Chantal Cherifi, Hocine Cherifi, Márton Karsai, and Mirco Musolesi, editors, *Complex Networks & Their Applications VI*, pages 241–252, Cham, 2018. Springer International Publishing.
[66] Issam Falih, Nistor Grozavu, Rushed Kanawati, and Younes Bennani. Community detection in attributed network. In *WWW '18 Companion Proceedings of the The Web Conference 2018*, pages 1299–1306, 2018.

[67] Saeed Farzi and Sahar Kianian. A novel clustering algorithm for attributed graphs based on k-medoid algorithm. *Journal of Experimental & Theoretical Artificial Intelligence*, 30(6):795–809, 2018.
[68] Andrew Fiore and Judith Donath. Homophily in online dating: When do you like someone like yourself? pages 1371–1374, 01 2005.
[69] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75 – 174, 2010.
[70] A. L. N. Fred and A. K. Jain. Data clustering using evidence accumulation. In *Object recognition supported by user interaction for service robots*, volume 4, pages 276–280 vol.4, Aug 2002.
[71] Brendan J. Frey and Delbert Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.
[72] Hongchang Gao and Heng Huang. Deep attributed network embedding. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 3364–3370. International Joint Conferences on Artificial Intelligence Organization, 7 2018.
[73] Jing Gao, Feng Liang, Wei Fan, Chi Wang, Yizhou Sun, and Jiawei Han. On community outliers and their efficient detection in information networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 813–822, New York, NY, USA, 2010. ACM.
[74] Rong Ge, Martin Ester, Byron J. Gao, Zengjian Hu, Binay Bhattacharya, and Boaz Ben-Moshe. Joint cluster analysis of attribute data and relationship data: The connected k-center problem, algorithms and applications. *ACM Trans. Knowl. Discov. Data*, 2(2):7:1–7:35, July 2008.
[75] Lisa Getoor, Nir Friedman, Daphne Koller, and Benjamin Taskar. Learning probabilistic models of link structure. *J. Mach. Learn. Res.*, 3:679–707, March 2003.
[76] 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.
[77] Derek Greene and Pádraig Cunningham. Producing a unified graph representation from multiple social network views. In *Proceedings of the 5th Annual ACM Web Science Conference*, WebSci '13, pages 118–121, New York, NY, USA, 2013. ACM.
[78] Aditya Grover and Jure Leskovec. Node2vec: Scalable feature learning for networks. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 855–864, New York, NY, USA, 2016. ACM.
[79] Thomas U. Grund and James A. Densley. Ethnic homophily and triad closure: Mapping internal gang structure using exponential random graph models. *Journal of Contemporary Criminal Justice*, 31(3):354–370, 2015.
[80] P. D. Grünwald. *The minimum description length principle*. The MIT Press, 2007.
[81] Quanquan Gu and Jie Zhou. Co-clustering on manifolds. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 359–368, New York, NY, USA, 2009. ACM.
[82] Francesco Gullo, Carlotta Domeniconi, and Andrea Tagarelli. Projective clustering ensembles. *Data Mining and Knowledge Discovery*, 26(3):452–511, May 2013.
[83] S. Gunnemann, I. Farber, B. Boden, and T. Seidl. Subspace clustering meets dense subgraph mining: A synthesis of two paradigms. In *2010 IEEE International Conference on Data Mining*, pages 845–850, Dec 2010.
[84] Stephan Gunnemann. Subspace clustering for complex data. In Volker Markl, Gunter Saake, Kai-Uwe Sattler, Gregor Hackenbroich, Bernhard Mitschang, Theo Harder, and Veit Koppen, editors, *Datenbanksysteme fur Business, Technologie und Web (BTW) 2034*, pages 343–362, Bonn, 2013. Gesellschaft fur Informatik e.V.
[85] Stephan Günnemann, Brigitte Boden, Ines Färber, and Thomas Seidl. Efficient mining of combined subspace and subgraph clusters in graphs with feature vectors. In Jian Pei, Vincent S. Tseng, Longbing Cao, Hiroshi Motoda, and Guandong Xu, editors, *Advances in Knowledge Discovery and Data Mining*, pages 261–275, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
[86] Stephan Günnemann, Brigitte Boden, and Thomas Seidl. Db-csc: A density-based approach for subspace clustering in graphs with feature vectors. In *Proceedings of the 2011th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I*, ECMLPKDD'11, pages 565–580, Berlin, Heidelberg, 2011. Springer-Verlag.
[87] Stephan Günnemann, Ines Färber, Brigitte Boden, and Thomas Seidl. Gamer: a synthesis of subspace clustering and dense subgraph mining. *Knowledge and Information Systems*, 40(2):243–278, Aug 2014.
[88] Stephan Günnemann, Ines Färber, Sebastian Raubach, and Thomas Seidl. Spectral subspace clustering for graphs with feature vectors. In *2013 IEEE 13th International Conference on Data Mining*, pages 231–240, 2013.
[89] T. Guo, S. Pan, X. Zhu, and C. Zhang. Cfond: Consensus factorization for co-clustering networked data. *IEEE Transactions on Knowledge and Data Engineering*, 31(4):706–719, April 2019.
[90] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 1024–1034. Curran Associates, Inc., 2017.
[91] J. A. Hartigan and M. A. Wong. A k-means clustering algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):100–108, 1979.
[92] C. He, X. Fei, H. Li, Y. Tang, H. Liu, and Q. Chen. A multi-view clustering method for community discovery integrating links and tags. In *2017 IEEE 14th International Conference on e-Business Engineering (ICEBE)*, pages 23–30, Nov 2017.
[93] Chaobo He, Shuangyin Liu, Lei Zhang, and Jianhua Zheng. A fuzzy clustering based method for attributed graph partitioning. *Journal of Ambient Intelligence and Humanized Computing*, 10(9):3399–3407, Sep 2019.

[94] Dongxiao He, Zhiyong Feng, Di Jin, Xiaobao Wang, and Weixiong Zhang. Joint identification of network communities and semantics via integrative modeling of network topologies and node contents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 116–124. AAAI Press, 2017.

[95] Alexander Hinneburg and Daniel A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, KDD'98, pages 58–65. AAAI Press, 1998.

[96] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137, 1983.

[97] L. Hu and K.C.C. Chan. Fuzzy clustering in a complex network based on content relevance and link structures. *IEEE transactions on fuzzy systems*, 24(2):456–470, 2016.

[98] Bingyang Huang, Chaokun Wang, and Binbin Wang. Nmlpa: Uncovering overlapping communities in attributed networks via a multi-label propagation approach. *Sensors (Basel, Switzerland)*, 19(2):260, Jan 2019.

[99] Xiao Huang, Jundong Li, and Xia Hu. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, pages 731–739, New York, NY, USA, 2017. ACM.

[100] Y. Huang and H. Wangg. Consensus and multiplex approach for community detection in attributed networks. In *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 425–429, Dec 2016.

[101] Zhichao Huang, Yunming Ye, Xutao Li, Feng Liu, and Huajie Chen. Joint weighted nonnegative matrix factorization for mining attributed graphs. In Jinho Kim, Kyuseok Shim, Longbing Cao, Jae-Gil Lee, Xuemin Lin, and Yang-Sae Moon, editors, *Advances in Knowledge Discovery and Data Mining*, pages 368–380, Cham, 2017. Springer International Publishing.

[102] Zhipeng Huang and Nikos Mamoulis. Heterogeneous information network embedding for meta path based proximity. *ArXiv*, abs/1701.05291, 2017.

[103] Roberto Interdonato, Martin Atzmueller, Sabrina Gaito, Rushed Kanawati, Christine Largeron, and Alessandra Sala. Feature-rich networks: going beyond complex network topologies. *Applied Network Science*, 4(1):4, Jan 2019.

[104] Hiroyoshi Ito, Takahiro Komamizu, Toshiyuki Amagasa, and Hiroyuki Kitagawa. Community detection and correlated attribute cluster analysis on multi-attributed graphs. In *EDBT/ICDT Workshops*, 2018.

[105] Tomoharu Iwata, Kazumi Saito, Naonori Ueda, Sean Stromsten, Thomas L. Griffiths, and Joshua B. Tenenbaum. Parametric embedding for class visualization. *Neural Computation*, 19(9):2536–2556, 2007.

[106] Caiyan Jia, Yafang Li, Matthew B. Carson, Xiaoyang Wang, and Jian Yu. Node attribute-enhanced community detection in complex networks. *Scientific Reports*, 7:2626:1–15, 2017.

[107] Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, Aug 2000.

[108] Wenhao Jiang, Hongchang Gao, Fu-lai Chung, and Heng Huang. The $\ell_{2,1}$-norm stacked robust autoencoders for domain adaptation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 1723–1729. AAAI Press, 2016.

[109] Stephen C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, Sep 1967.

[110] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov 1999.

[111] D.R. Karger. Global min-cuts in rnc, and other ramifications of a simple min-cut algorithm. pages 21–30, 1993.

[112] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, December 1998.

[113] N. Khediri and W. Karoui. Community detection in social network with node attributes based on formal concept analysis. In *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, pages 1346–1353, Oct 2017.

[114] Thomas N. Kipf and Max Welling. Variational graph auto-encoders. *ArXiv*, abs/1611.07308, 2016.

[115] 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.

[116] Gueorgi Kossinets and Duncan J. Watts. Origins of homophily in an evolving social network. *American Journal of Sociology*, 115:405–450, 2009.

[117] Da Kuang, Sangwoon Yun, and Haesun Park. Symnmf: Nonnegative low-rank approximation of a similarity matrix for graph clustering. *J. of Global Optimization*, 62(3):545–574, July 2015.

[118] Andrea Lancichinetti and Santo Fortunato. Consensus clustering in complex networks. *Scientific Reports*, 2:336:1–7, Mar 2012.

[119] Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, mar 2009.

[120] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E*, 78:046110, Oct 2008.

[121] E. Lazega. *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*. Oxford University Press, Oxford, UK, 2001.

[122] T. M. V. Le and H. W. Lauw. Probabilistic latent document network embedding. In *2014 IEEE International Conference on Data Mining*, pages 270–279, Dec 2014.

[123] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001.

[124] Jure Leskovec, Kevin J. Lang, and Michael Mahoney. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 631–640, New York, NY, USA, 2010. ACM.

[125] Jure Leskovec and Julian J. Mcauley. Learning to discover social circles in ego networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 539–547. Curran Associates, Inc., 2012.

[126] P. Li, L. Huang, C. Wang, D. Huang, and J. Lai. Community detection using attribute homogenous motif. *IEEE Access*, 6:47707–47716, 2018.

[127] Wu-Jun Li, Dit-Yan Yeung, and Zhihua Zhang. Generalized latent factor models for social network analysis. In *IJCAI*, 2011.

[128] Y. Li, C. Jia, X. Kong, L. Yang, and J. Yu. Locally weighted fusion of structural and attribute information in graph clustering. *IEEE Transactions on Cybernetics*, 49(1):247–260, Jan 2019.

[129] Ye Li, Chaofeng Sha, Xin Huang, and Yanchun Zhang. Community detection in attributed graphs: An embedding approach. In *AAAI*, 2018.

[130] Z. Li, J. Liu, and K. Wu. A multiobjective evolutionary algorithm based on structural and attribute similarities for community detection in attributed networks. *IEEE Transactions on Cybernetics*, 48(7):1963–1976, July 2018.

[131] Zhen LI, Zhisong PAN, Guyu HU, Guopeng LI, and Xingyu ZHOU. Detecting semantic communities in social networks. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E100.A(11):2507–2512, 2017.

[132] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. Building text classifiers using positive and unlabeled examples. *Third IEEE International Conference on Data Mining*, pages 179–186, 2003.

[133] L. Liu, L. Xu, Z. Wangy, and E. Chen. Community detection based on structure and content: A content propagation perspective. In *2015 IEEE International Conference on Data Mining*, pages 271–280, Nov 2015.

[134] Sheng Luo, Zhifei Zhang, Yuanjian Zhang, and Shuwen Ma. Co-association matrix-based multi-layer fusion for community detection in attributed networks. *Entropy*, 21(1), 2019.

[135] David J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, New York, NY, USA, 2002.

[136] Gregory R. Madey, Albert-László Barabási, Nitesh V. Chawla, Marta Gonzalez, David Hachen, Brett Lantz, Alec Pawling, Timothy Schoenharl, Gábor Szabó, Pu Wang, and Ping Yan. Enhanced situational awareness: Application of dddas concepts to emergency and disaster management. In Yong Shi, Geert Dick van Albada, Jack Dongarra, and Peter M. A. Sloot, editors, *Computational Science – ICCS 2007*, pages 1090–1097, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[137] Seiji Maekawa, Koh Takeuchi, and Makoto Onizuka. Non-linear attributed graph clustering by symmetric nmf with pu learning. *ArXiv*, abs/1810.00946, 2018.

[138] Peter V. Marsden and Noah E. Friedkin. Network studies of social influence. *Sociological Methods & Research*, 22(1):127–151, 1993.

[139] Julian Mcauley and Jure Leskovec. Discovering social circles in ego networks. *ACM Trans. Knowl. Discov. Data*, 8(1):4:1–4:28, February 2014.

[140] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[141] Fanrong Meng, Xiaobin Rui, Zhixiao Wang, Yan Xing, and Longbing Cao. Coupled node similarity learning for community detection in attributed networks. *Entropy*, 20(6), 2018.

[142] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA, 2013. Curran Associates Inc.

[143] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[144] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, IMC '07, pages 29–42, New York, NY, USA, 2007. ACM.

[145] Dharmendra S. Modha and W. Scott Spangler. Clustering hypertext with applications to web searching. In *Proceedings of the Eleventh ACM on Hypertext and Hypermedia*, HYPERTEXT '00, pages 143–152, New York, NY, USA, 2000. ACM.

[146] Flavia Moser, Recep Colak, Arash Rafiey, and Martin Ester. Mining cohesive patterns from graphs with feature vectors. In *SDM*, pages 593–604. SIAM, 2009.

[147] Flavia Moser, Rong Ge, and Martin Ester. Joint cluster analysis of attribute and relationship data withouta-priori specification of the number of clusters. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 510–519, New York, NY, USA, 2007. ACM.

[148] E. Muller, P. I. Sanchez, Y. Mulle, and K. Bohm. Ranking outlier nodes in subspaces of attributed graphs. In *2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW 2013)*, pages 216–222, Los Alamitos, CA, USA, apr 2013. IEEE Computer Society.

[149] N. Muslim. A combination approach to community detection in social networks by utilizing structural and attribute data. *Social Networking*, 5:11–15, 2016.

[150] M. P. Naik, H. B. Prajapati, and V. K. Dabhi. A survey on semantic document clustering. In *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–10, March 2015.

[151] Ramesh M. Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 542–550, New York, NY, USA, 2008. ACM.

[152] Waqas Nawaz, Kifayat-Ullah Khan, Young-Koo Lee, and Sungyoung Lee. Intra graph clustering using collaborative similarity measure. *Distributed and Parallel Databases*, 33(4):583–603, Dec 2015.

[153] Jennifer Neville, Micah Adler, and David Jensen. Clustering relational data using attribute and link information. In *In Proceedings of the Text Mining and Link Analysis Workshop, 18th International Joint Conference on Artificial Intelligence*, pages 9–15, 2003.

[154] M. Newman and Aaron Clauset. Structure and inference in annotated networks. *Nature Communications*, 7, 07 2015.

[155] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74:036104, Sep 2006.

[156] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, Feb 2004.

[157] H. T. Nguyen and T. N. Dinh. Unveiling the structure of multi-attributed networks via joint non-negative matrix factorization. In *MILCOM 2015 - 2015 IEEE Military Communications Conference*, pages 1379–1384, Oct 2015.

[158] Wouter de Nooy, Andrej Mrvar, and Vladimir Batagelj. *Exploratory Social Network Analysis with Pajek*. Cambridge University Press, New York, NY, USA, 2004.

[159] Madalina Olteanu, Nathalie Villa-Vialaneix, and Christine Cierco-Ayrolles. Multiple kernel self-organizing maps. In Michel Verleysen, editor, *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, page 83, Bruges, Belgium, April 2013.

[160] 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. Previous number = SIDL-WP-1999-0120.

[161] Andreas Papadopoulos, George Pallis, and Marios D. Dikaiakos. Weighted clustering of attributed multi-graphs. *Computing*, 99(9):813–840, Sep 2017.

[162] Andreas Papadopoulos, Dimitrios Rafailidis, George Pallis, and Marios D. Dikaiakos. Clustering attributed multi-graphs with information ranking. In Qiming Chen, Abdelkader Hameurlain, Farouk Toumani, Roland Wagner, and Hendrik Decker, editors, *Database and Expert Systems Applications*, pages 432–446, Cham, 2015. Springer International Publishing.

[163] M. Parimala and Daphne Lopez. Graph clustering based on structural attribute neighborhood similarity (sans). *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–4, 2015.

[164] Yulong Pei, Nilanjan Chakraborty, and Katia Sycara. Nonnegative matrix tri-factorization with graph regularization for community detection in social networks. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 2083–2089. AAAI Press, 2015.

[165] Z. Pei, X. Zhang, F. Zhang, and B. Fang. Attributed multi-layer network embedding. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 3701–3710, Dec 2018.

[166] Bryan Perozzi, Leman Akoglu, Patricia Iglesias Sánchez, and Emmanuel Müller. Focused clustering and outlier detection in large attributed graphs. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1346–1355, New York, NY, USA, 2014. ACM.

[167] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 701–710, New York, NY, USA, 2014. ACM.

[168] C. Pizzuti and A. Socievole. Multiobjective optimization and local merge for clustering attributed graphs. *IEEE Transactions on Cybernetics*, pages 1–13, 2019.

[169] Clara Pizzuti and Annalisa Socievole. A genetic algorithm for community detection in attributed graphs. In Kevin Sim and Paul Kaufmann, editors, *Applications of Evolutionary Computation*, pages 159–170, Cham, 2018. Springer International Publishing.

[170] Simon Pool, Francesco Bonchi, and Matthijs van Leeuwen. Description-driven community detection. *ACM Trans. Intell. Syst. Technol.*, 5(2):28:1–28:28, April 2014.

[171] Meng Qin, Di Jin, Kai Lei, Bogdan Gabrys, and Katarzyna Musial-Gabrys. Adaptive community detection incorporating topology and content in social networks. *Knowledge-Based Systems*, 161:342 – 356, 2018.

[172] Yiye Ruan, David Fuhry, and Srinivasan Parthasarathy. Efficient community detection in large networks using content and links. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1089–1098, New York, NY, USA, 2013. ACM.

[173] Mrinmaya Sachan, Danish Contractor, Tanveer A. Faruquie, and L. Venkata Subramaniam. Using content and interactions for discovering communities in social networks. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 331–340, New York, NY, USA, 2012. ACM.

[174] N. Y. Saiyad, H. B. Prajapati, and V. K. Dabhi. A survey of document clustering using semantic approach. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 2555–2562, March 2016.

[175] P. I. Sanchez, E. Muller, F. Laforet, F. Keller, and K. Bohm. Statistical selection of congruent subspaces for mining attributed graphs. In *2013 IEEE 13th International Conference on Data Mining*, pages 647–656, Dec 2013.

[176] Patricia Iglesias Sánchez, Emmanuel Müller, Uwe Leo Korn, Klemens Böhm, Andrea Kappes, Tanja Hartmann, and Dorothea Wagner. Efficient algorithms for a robust modularity-driven clustering of attributed graphs. In *SDM*, 2015.

[177] Venu Satuluri and Srinivasan Parthasarathy. Scalable graph clustering using stochastic flows: Applications to community discovery. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 737–746, New York, NY, USA, 2009. ACM.

[178] Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27 – 64, 2007.

[179] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29:93–106, 2008.

[180] Nasrullah Sheikh, Zekarias Kefato, and Alberto Montresor. gat2vec: representation learning for attributed graphs. *Computing*, 101(3):187–209, Mar 2019.

[181] Motoki Shiga, Ichigaku Takigawa, and Hiroshi Mamitsuka. A spectral clustering approach to optimally combining numericalvectors with a modular network. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 647–656, New York, NY, USA, 2007. ACM.

[182] Alexander J. Smola and Risi Kondor. Kernels and regularization on graphs. In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Learning Theory and Kernel Machines*, pages 144–158, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[183] Tom A. B. Snijders and Krzysztof Nowicki. Estimation and prediction for stochastic block-structures for graphs with latent block structure. 1997.

[184] Benno Stein and Oliver Niggemann. On the nature of structure and its identification. In *Proceedings of the 25th International Workshop on Graph-Theoretic Concepts in Computer Science*, WG '99, pages 122–134, London, UK, UK, 1999. Springer-Verlag.

[185] Karsten Steinhaeuser and Nitesh V. Chawla. Community detection in a large real-world social network. In Huan Liu, John J. Salerno, and Michael J. Young, editors, *Social Computing, Behavioral Modeling, and Prediction*, pages 168–175, Boston, MA, 2008. Springer US.

[186] Karsten Steinhaeuser and Nitesh V. Chawla. Identifying and evaluating community structure in complex networks. *Pattern Recognition Letters*, 31(5):413 – 421, 2010.

[187] Alexander Strehl and Joydeep Ghosh. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, 3:583–617, March 2003.

[188] Y. Sun, J. Han, J. Gao, and Y. Yu. itopicmodel: Information network-integrated topic modeling. In *2009 Ninth IEEE International Conference on Data Mining*, pages 493–502, Dec 2009.

[189] 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.

[190] Aditya Tandon, Aiiad Albeshri, Vijey Thayananthan, Wadee Alhalabi, and Santo Fortunato. Fast consensus clustering in complex networks. *Phys. Rev. E*, 99:042301, Apr 2019.

[191] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 1067–1077, Republic and Canton of Geneva, Switzerland, 2015. International World Wide Web Conferences Steering Committee.

[192] Lei Tang and Huan Liu. Scalable learning of collective behavior based on sparse social dimensions. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 1107–1116, New York, NY, USA, 2009. ACM.

[193] Mariano Tepper and Guillermo Sapiro. From local to global communities in large networks through consensus. In Alvaro Pardo and Josef Kittler, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 659–666, Cham, 2015. Springer International Publishing.

[194] Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. Learning deep representations for graph clustering. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 1293–1299. AAAI Press, 2014.

[195] Yuanyuan Tian, Richard A. Hankins, and Jignesh M. Patel. Efficient aggregation for graph summarization. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 567–580, New York, NY, USA, 2008. ACM.

[196] H. Tong, C. Faloutsos, and J. Pan. Fast random walk with restart and its applications. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 613–622, Dec 2006.

[197] AL Traud, ED Kelsic, PJ Mucha, and MA Porter. Comparing community structure to characteristics in online collegiate social networks. *SIAM REVIEW*, 53(3):526–543, 2011.

[198] Amanda L. Traud, Peter J. Mucha, and Mason A. Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165 – 4180, 2012.

[199] S. van Dongen. *Graph clustering by flow simulation*. PhD thesis, University of Utrecht, 2000.

[200] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[201] Nathalie Villa-Vialaneix, Madalina Olteanu, and Christine Cierco-Ayrolles. Carte auto-organisatrice pour graphes étiquetés. In *Atelier Fouilles de Grands Graphes (FGG) - EGC'2013*, page Article numéro 4, Toulouse, France, January 2013.

[202] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. Attributed graph clustering: A deep attentional embedding approach. *arXiv:1906.06532*, 06 2019.

[203] Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang. Mgae: Marginalized graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pages 889–898, New York, NY, USA, 2017. ACM.

[204] Hua Wang, Feiping Nie, Heng Huang, and Fillia Makedon. Fast nonnegative matrix tri-factorization for large-scale data co-clustering. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, pages 1553–1558. AAAI Press, 2011.

[205] X. Wang, L. Tang, H. Gao, and H. Liu. Discovering overlapping groups in social media. In *2010 IEEE International Conference on Data Mining*, pages 569–578, Dec 2010.

[206] Xiao Wang, Di Jin, Xiaochun Cao, Liang Yang, and Weixiong Zhang. Semantic community identification in large attribute networks. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 265–271. AAAI Press, 2016.

[207] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[208] Ron Weiss, Bienvenido Vélez, and Mark A. Sheldon. Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering. In *Proceedings of the the Seventh ACM Conference on Hypertext*, HYPERTEXT '96, pages 180–193, New York, NY, USA, 1996. ACM.

[209] J. J. Whang, D. F. Gleich, and I. S. Dhillon. Overlapping community detection using neighborhood-inflated seed expansion. *IEEE Transactions on Knowledge and Data Engineering*, 28(5):1272–1284, May 2016.

[210] Peng Wu and Li Pan. Mining application-aware community organization with expanded feature subspaces from concerned attributes in social networks. *Knowledge-Based Systems*, 139:1 – 12, 2018.

[211] Rongkai Xia, Yan Pan, Lei Du, and Jian Yin. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 2149–2155. AAAI Press, 2014.

[212] Xiaofeng He, C. H. Q. Ding, Hongyuan Zha, and H. D. Simon. Automatic topic identification using webpage clustering. In *Proceedings 2001 IEEE International Conference on Data Mining*, pages 195–202, Nov 2001.

[213] Jierui Xie and Boleslaw K. Szymanski. Towards linear time overlapping community detection in social networks. In Pang-Ning Tan, Sanjay Chawla, Chin Kuan Ho, and James Bailey, editors, *Advances in Knowledge Discovery and Data Mining*, pages 25–36, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[214] Z. Xu, Y. Ke, Y. Wang, H. Cheng, and J. Cheng. A model-based approach to attributed graph clustering. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 505–516, 2012.

[215] Zhiqiang Xu, James Cheng, Xiaokui Xiao, Ryohei Fujimaki, and Yusuke Muraoka. Efficient nonparametric and asymptotic bayesian model selection methods for attributed graph clustering. *Knowl. Inf. Syst.*, 53(1):239–268, October 2017.

[216] Zhiqiang Xu and Yiping Ke. Effective and efficient spectral clustering on text and link data. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, pages 357–366, New York, NY, USA, 2016. ACM.

[217] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. Gbagc: A general bayesian framework for attributed graph clustering. *ACM Trans. Knowl. Discov. Data*, 9(1):5:1–5:43, August 2014.

[218] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. Network representation learning with rich text information. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 2111–2117. AAAI Press, 2015.

[219] Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: A nonnegative matrix factorization approach. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pages 587–596, New York, NY, USA, 2013. ACM.

[220] Jaewon Yang, Julian J. McAuley, and Jure Leskovec. Community detection in networks with node attributes. *2013 IEEE 13th International Conference on Data Mining*, pages 1151–1156, 2013.

[221] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. *Directed Network Community Detection: A Popularity and Productivity Link Model*, pages 742–753. 2010.

[222] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. Combining link and content for community detection: A discriminative approach. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 927–936, New York, NY, USA, 2009. ACM.

[223] Zhao Yang, René Algesheimer, and Claudio J. Tessone. A comparative analysis of community detection algorithms on artificial networks. In *Scientific reports*, 2016.

[224] Wei Ye, Linfei Zhou, Xin Sun, Claudia Plant, and Christian Böhm. Attributed graph clustering with unimodal normalized cut. In Michelangelo Ceci, Jaakko Hollmén, Ljupčo Todorovski, Celine Vens, and Sašo Džeroski, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 601–616, Cham, 2017. Springer International Publishing.

[225] T. Yoshida. Toward finding hidden communities based on user profile. In *2010 IEEE International Conference on Data Mining Workshops*, pages 380–387, Dec 2010.

[226] Donghua Yu, Guojun Liu, Maozu Guo, and Xiaoyan Liu. An improved k-medoids algorithm based on step increasing and optimizing medoids. *Expert Systems with Applications*, 92:464 – 473, 2018.

[227] Hugo Zanghi, Stevenn Volant, and Christophe Ambroise. Clustering based on random graph model embedding vertex features. *Pattern Recognition Letters*, 31(9):830 – 836, 2010.

[228] Tong Zhang, Alexandrin Popescul, and Byron Dom. Linear prediction models with graph regularization for web-page categorization. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 821–826, New York, NY, USA, 2006. ACM.

[229] Yuan Zhang, Elizaveta Levina, and Ji Zhu. Community detection in networks with node features. *Electron. J. Statist.*, 10(2):3153–3178, 2016.

[230] Ding Zhou, Eren Manavoglu, Jia Li, C. Lee Giles, and Hongyuan Zha. Probabilistic models for discovering e-communities. In *Proceedings of the 15th International Conference on World Wide Web*, WWW '06, pages 173–182, New York, NY, USA, 2006. ACM.

[231] Yang Zhou, Hong Cheng, and Jeffrey Xu Yu. Graph clustering based on structural/attribute similarities. *Proc. VLDB Endow.*, 2(1):718–729, August 2009.

[232] Yang Zhou, Hong Cheng, and Jeffrey Xu Yu. Clustering large attributed graphs: An efficient incremental approach. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pages 689–698, Washington, DC, USA, 2010. IEEE Computer Society.

[233] Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 487–494, New York, NY, USA, 2007. ACM.

[234] Albrecht Zimmermann, Björn Bringmann, and Ulrich Rückert. Fast, effective molecular feature mining by local optimization. In *ECML/PKDD*, 2010.

NATIONAL CENTER FOR COGNITIVE TECHNOLOGIES, ITMO UNIVERSITY (SAINT PETERSBURG, RUSSIA)