

Article

TaCD: Team-Aware Community Detection Based on Multi-View Modularity

Chengzhou Fu ^{1,†} , Feiyi Tang ^{2,†}, Lingzhi Hu ^{3,4,*}, Chengzhe Yuan ⁵ and Ronghua Lin ⁶

¹ College of Medical Information Engineering, Guangdong Pharmaceutical University, Guangzhou 510006, China; fucz@gdpu.edu.cn

² School of Information Engineering, Guangzhou Polytechnic University, Guangzhou 511483, China; tangfy@gzypu.edu.cn

³ School of Modern Information Industry, Guangzhou College of Commerce, Guangzhou 511363, China

⁴ Faculty of Applied Sciences, Macao Polytechnic University, Macau 999078, China

⁵ School of Electronics and Information, Guangdong Polytechnic Normal University, Guangzhou 510665, China; ycz@gpnu.edu.cn

⁶ School of Computer Science, South China Normal University, Guangzhou 510631, China; rhlin@m.scnu.edu.cn

* Correspondence: hulingz@hotmail.com

† These authors contributed equally to this work.

Abstract

Community detection in social networks is one of the most important topics of network science. Researchers have developed numerous methods from various perspectives. However, the existing methods often overlook the team information encoded as a special type of user relation in the social network, which plays an important role in community formation and evolution. In this paper, we propose a novel community detection algorithm called Team-aware Community Detection (TaCD). Our model constructs a multi-view network by encoding the user interaction information as the user view and the team information as the team view. To measure the consistency across the two views, we use the Jaccard similarity to establish a cross-view coupling. Based on the constructed 2-view network, we use multi-view modularity to discover team-aware community structure, and solve the optimization problem using the well-known Generalized Louvain approach. Another contribution of this paper is the collection of a new SCHOLAT dataset, which consists of several social networks with team information and is publicly available for testing purposes. Our experimental results on several SCHOLAT networks with team information demonstrate that TaCD outperforms the existing community detection algorithms.

Keywords: team-aware; community detection; multi-view; Generalized Louvain; SCHOLAT



Academic Editor: Hocine Cherifi

Received: 20 November 2025

Revised: 14 December 2025

Accepted: 20 December 2025

Published: 24 December 2025

Copyright: © 2025 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Community detection is to discover both hidden and defined communities from the distributed and disordered structure of the internet and complex social systems [1–3]. Identifying communities can provide information about how the network is organized. It allows us to focus on areas of the graph that have a degree of autonomy. It also helps to classify vertices according to their roles relative to their communities [4]. Community detection has various applications in different fields, which can be used to uncover potential relationships between users in the field of social software development [5,6]. It can also be used to detect the structure of protein–protein interaction in the field of biology [7–9]. Even for the real internet, community detection can also be used to discover related

websites [10–13]. In the era of big data, it is critical to discover the meaningful community structure when dealing with numerous huge networks.

During the past few decades, more and more efforts have been made on community detection [14–17]. However, most of them only consider the user interaction or attribute information, but ignore the relationship with the teammates in the social network service platform. To demonstrate the importance of team information in community detection in social networks, we collect a dataset from SCHOLAT (SCHOLAT: <https://www.scholat.com>) [18] (a well-known academic social networking service platform in China). This dataset contains eight networks and will be made publicly available to all of the scholars for academic research usage. Different from the existing social network datasets, the social networks on the SCHOLAT dataset contain not only the user–user interaction information but also the team information. In SCHOLAT, the relationship between users is complex. There are many teams such as the Academic-team and Class-team which are created by users. As shown in Figure 1, there are three teams such as $T_1 = \{u_2, u_3, u_4, u_6, u_9, u_{12}, u_{13}, u_{17}\}$, $T_2 = \{u_1, u_3, u_7, u_{13}, u_{15}\}$, and $T_3 = \{u_1, u_3, u_5, u_8, u_9, u_{10}, u_{11}, u_{14}, u_{16}, u_{17}\}$. Some users are in the same team, but they may not have friendship (e.g., in the team T_1 : u_4 and u_9 , u_2 and u_6 , etc.). Similarly, the users have friendship, but they may not be in the same team (e.g., u_6 in the team T_1 and u_1 in the team T_2 / T_3 , etc.), and others may not be in the same team or have a friendship (e.g., u_4 in the team T_1 and u_1 in the team T_2 and T_3 , etc.).

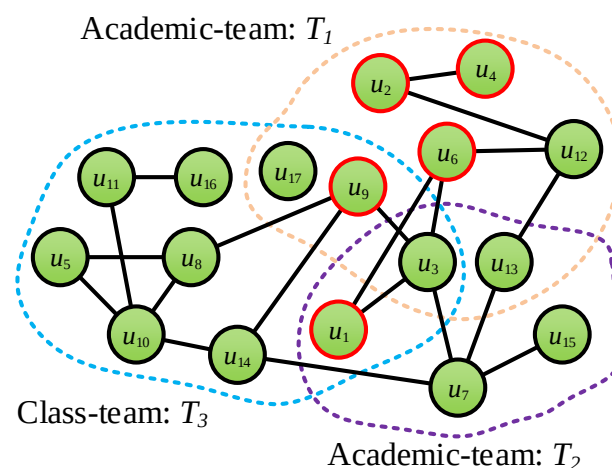


Figure 1. Illustration of network structure in SCHOLAT.

Unlike traditional multi-layer networks that typically aggregate homogeneous relationships (e.g., merging Twitter and Facebook ties), TaCD introduces a heterogeneous coupling mechanism. It integrates the explicit “Affiliation Layer” (Team-view) with the implicit “Interaction Layer” (User-view). By measuring the structural consistency between these two distinct views, we can effectively filter out noise where team membership does not reflect actual social proximity.

Despite the aforementioned importance of team information, it is mostly ignored in the existing methods. To this end, in this paper, we propose a new community detection method called Team-aware Community Detection (TaCD). In particular, a new 2-view network is constructed from the original network with team information, which consists of the user view for encoding user–user interaction and the team view for encoding team information. For measuring the consistency across the two views, the Jaccard similarity is adopted, whereby a cross-view coupling is established. Based on the newly constructed 2-view network, multi-view modularity is adopted to discover team-aware community structure, and solve this optimization problem using the well-known Generalized Louvain approach. Another contribution of this paper is that a new SCHOLAT dataset consisting of

several social networks with team information is collected and made publicly available as a testing dataset. Extensive experiments are conducted to confirm the superiority of the proposed TaCD method over the existing methods.

The method introduced in this work has the following novel contributions: constructing a new dataset that is more suitable for large-scale networks, supporting multi-layer networks, and solving the coupling calculation problems between layers. The rest of this paper is organized as follows. We briefly review the related work in Section 2, and introduce the SCHOLAT dataset and research background in Section 3. In Section 4, the newly proposed Team-aware Community Detection approach is described in detail. In Section 5, extensive experiments are conducted to validate the effectiveness of the proposed method. At last, we will draw the conclusions and describe the future work in Section 6.

2. Related Work

In this section, we will briefly review some related work on community detection. One major type of community detection methods relies on designing a quality function, by solving the optimization problem of which the community structure can be discovered. For example, the normalized cuts (NCut)-based method measures both the total dissimilarity between the different groups as well as the total similarity within the groups, where a real valued solution to the normalized cut minimization problem is provided by a generalized eigenvalue system [19]. The non-negative matrix factorization (NMF) is an effective approach for community detection that utilizes a Bayesian model to extract overlapping modules from a network [20]. Cluster Affiliation Model for Big Networks (BIG-CLAM) is another overlapping community detection method that scales to large networks of millions of nodes and edges. The method builds on a novel observation that overlaps between communities are densely connected [15]. Communities from Edge Structure and Node Attributes (CESNA) is an accurate and scalable algorithm for detecting overlapping communities in networks with node attributes. CESNA statistically models the interaction between the network structure and the node attributes, which leads to more accurate community detection as well as improved robustness in the presence of noise in the network structure [11]. Recently, some efforts have been made in higher-order community detection. It has been shown that higher-order features captured by network motifs are crucial in many domains, such as biology and neuron-science, which can help to gain new insights into the network organization beyond the clustering at the level of individual nodes and edges [21]. Rosvall and Bergstrom use the simulated annealing optimization algorithm and the effective coding of random walks for community detection [22]. Raghavan et al. propose a Label Propagation Algorithm (LPA), which is based on the idea that the edge of the network often represents the propagation of information [23]. A label update rule is further proposed for further reducing computational overhead [24]. Ma et al. [25] propose a novel algorithm by joint multi-label learning and feature extraction (MLjFE), where temporal link prediction and feature extraction are integrated into an overall objective function. Mahmood and Small present a community detection algorithm based on the fact that each network community spans a different subspace in the geodesic space. For making the process of community detection more robust, they use sparse linear coding with ℓ_1 norm constraint [26]. Jin proposes an approach to community detection termed Spectral Clustering On Ratios-of-Eigenvectors (SCORE), the main innovation of which is to use the entry-wise ratios between the first leading eigenvector and each of the other leading eigenvectors for clustering [27]. In fact, in community detection, new data are always generated continuously with subgraphs joining simultaneously in dynamic evolving networks. For addressing the above problem, Zhao et al. present a method to detect communities by handling subgraphs [28].

Recently, multi-view community detection has been widely studied, such as modularity-based methods [29,30] and information-theoretic methods [31]. Modularity evaluates the quality of a network partition, where a higher modularity value usually indicates a denser edge distribution within communities [32,33]. Furthermore, Delvenne et al. derive the multi-layer modularity by assessing the capability of the given community structure to capture a dynamic process in a multi-layer network [34]. Ma et al. propose a quantitative function (multi-layer modularity density), while considering the connection information among various layers [35] for community detection in multi-layer networks.

However, the algorithms described above ignore team information. The purpose of this paper is to enhance the performance of community detection by incorporating team information from the datasets. In particular, we use the multi-view method to detect team-aware communities as shown in Figure 2. For example, with the three blue nodes in User-view and three red nodes in Team-view, although they represent the same three nodes, they have different relationships in the two views.

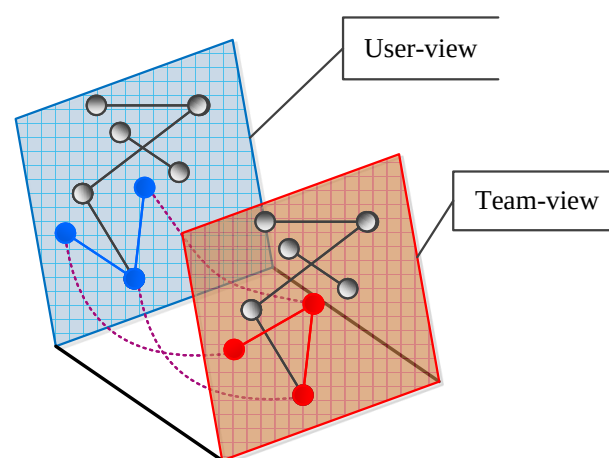


Figure 2. The User-view (blue) and Team-view (red) in our method. Solid lines represent interaction intensity, while dashed lines represent coupling.

Recent advancements have also explored dynamic and feature-based clustering. For instance, matrix factorization has been revisited for dynamic graph clustering [36], and joint learning frameworks have been proposed for large-scale temporal networks [37]. While effective, our approach differs by focusing specifically on leveraging static affiliation meta-data as a structural view.

3. Dataset Description

In this section, we describe the SCHOLAT dataset from the following perspectives.

3.1. About SCHOLAT and Its Dataset

SCHOLAT is an academic social networking service website designed to promote exchanges and cooperation between researchers. The platform contains multiple features such as academic information management, literature search, academic network disk, teaching course management, and scholar exchange services.

Since its establishment, SCHOLAT has gained widespread recognition and has attracted a large number of scholars, teachers, and students to utilize its services. It serves as a scientific research platform for scholars, focusing on engineering applications, theoretical research, and academic exchange. SCHOLAT assists scientific researchers in building their own academic networks, helps students find suitable mentors, and provides up-to-date job opportunities for those seeking scientific research positions.

A given sample of real community relationship in SCHOLAT is shown in Figure 3, and a community summary is shown in at bottom left of the figure.



Figure 3. The real community relationship in the SCHOLAT dataset: we see that only the nodes inside the community have a relationship, and there is no connection between the nodes in different communities.

It is important to note that we cannot directly regard team as a community, because the concept of team is only relative to the users of SCHOLAT platform, but the underlying community is the collective division in real life.

For users, their relationship at the User-view and at the Team-view level are independent, but if we make good use of the relationship between the two levels in community detection, it can help us to have a better understanding of the relationship between users.

To this end, the motivation of this research can be addressed as it is necessary to consider team information on community detection. To tackle this issue, we propose a novel method for Team-aware Community Detection in social networks.

3.2. The Networks in Dataset

As shown in Table 1, we use the eight networks to do the experiments, namely, Net-3k, Net-4k, ..., and Net-10k. The N_k notation (e.g., Net-3k) denotes a network with $n = N \times 1000$ nodes (e.g., $n = 3000$). Specifically, the largest network consists of 10,000 nodes (with 5,713,566 edges and 218 communities). The work departments of users on the dataset mainly contain universities, academic organizations, and companies from China. In addition, these users are almost all registered with real names, and the quality of their information is also generally high. To protect the users' privacy, we discretize

their ids (If you use this dataset in your work, please cite this publication. You can download the dataset and related code from here (the password of the zip file is “Goodluck!”): <https://www.scholat.com/research/tacd>, accessed on 13 November 2025).

Table 1. The statistics of the SCHOLAT dataset.

Networks	Nodes	Edges	Communities	User ids
Net-3k	3000	689,480	157	1–3000
Net-4k	4000	992,387	162	1–4000
Net-5k	5000	1,385,964	178	1–5000
Net-6k	6000	2,462,527	178	1–6000
Net-7k	7000	3,107,667	186	1–7000
Net-8k	8000	4,061,459	197	1–8000
Net-9k	9000	5,009,882	208	1–9000
Net-10k	10,000	5,713,566	218	1–10,000

The SCHOLAT dataset includes the following files, which are

- *user_real_community.csv*: the row number represents user id, and value in each cell is the real community id;
- *link_friendship.csv*: the link information of friendship as $\langle user_1, user_2 \rangle$, using user follower/followee relations;
- *matrix_common_team_count.csv*: contains three columns such as $user_1$, $user_2$ and the number of common teams;
- *matrix_interact_times.csv*: which contains interaction information, constructed via Equation (1) in Section 4.1;
- *matrix_friendship.csv*: the matrix of *link_friendship.csv*;
- *matrix_jaccard.csv*: which computed via Equation (4) in Section 4.2.

Example 1. In the net-3k with 3000 nodes and 689,480 edges, the user id range is from 1 to 3000. There are 157 communities in the real world, including Apple Inc., DIGITO Agency, Faimdata, Tai Fung Bank, Guangdong Pharmaceutical University, China University of Geosciences (Beijing) and South China Normal University et al. The users’ names are those such as Yong-Tang, Na-Tang, Xiao-Liu, Long-Zhang, Yuncheng-Jiang, and Li-Huang et al.

4. The Proposed TaCD Method

In TaCD, we propose a Generic Multi-view Interaction Framework designed to bridge heterogeneous social information. Unlike traditional multi-layer networks that aggregate homogeneous ties, our framework constructs a network consisting of two distinct logical views: an Interaction Layer (View s) representing implicit pairwise behaviors, and an Affiliation Layer (View r) representing explicit shared group memberships.

4.1. Constructing Matrices of the View Information

The conceptual structure is illustrated in Figure 2. For example, the three blue nodes in the User-view (Interaction Layer) and the three red nodes in the Team-view (Affiliation Layer) represent the same entities but exhibit different topological relationships. Our method exploits this complementary information.

The proposed framework is formalized by a set of adjacency matrices $\mathcal{A}_{(i,j)}^{[*]}$, specifically comprising the Interaction View $\mathcal{A}_{(i,j)}^{[s]}$ and the Affiliation View $\mathcal{A}_{(i,j)}^{[r]}$. Here, $\mathcal{A}_{(i,j)}^{[s]}$ is the adjacency matrix encoding the frequency of interactions between user pairs, and $\mathcal{A}_{(i,j)}^{[r]}$ represents the matrix quantifying the extent of shared team affiliations. The detailed construction of these matrices is described as follows.

(1) Interaction Layer (View s): This layer encodes the intensity of pairwise user interactions. In a general social network, this is formalized as a weighted sum of various interaction types. Taking the SCHOLAT dataset as a specific case study, we instantiate this layer by mapping *Friendship*, *Like*, and *Chat* behaviors to the interaction weights:

$$\mathcal{A}_{(i,j)}^{[s]} = \chi \mathcal{F}(i,j) + \psi \mathcal{L}(i,j) + \omega \mathcal{C}(i,j) \quad (1)$$

We can compute the *Friendship* value $\mathcal{F}(i,j)$ of user i and j as

$$\mathcal{F}(i,j) = \begin{cases} 1, & i \text{ and } j \text{ are friends} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $\mathcal{F}(i,j)$ is the *Friendship* value of user i and j are computed via Equation (2). The $\mathcal{L}(i,j)$ represents the number of times user i performs a ‘Like’ action on user j (or vice versa). The $\mathcal{C}(i,j)$ represents the frequency with which user i and j *Chat* with each other.

In our proposed generic framework, the weighting parameters χ , ψ , and ω are tunable to reflect the varying importance of different interaction types across specific platforms. In the context of the SCHOLAT dataset, empirical observations suggest that explicit social ties imply stronger connections than casual interactions. Therefore, we set $\chi = 2$ to emphasize the importance of friendship, while setting $\psi = 1$ and $\omega = 1$ for ‘Like’ and ‘Chat’ interactions, respectively.

Intuitively, the relationships among nodes are more important than other types of interaction, so it should have greater weight. Therefore, in this paper, we let $\chi = 2$ and $\psi = \omega = 1$.

According to Equation (1), consider a scenario, where SCHOLAT users u_1, u_2, u_3, u_4 interact using the public interactions in view s as shown in Table 2 and Figure 4.

Table 2. Interaction among u_1, u_2, u_3, u_4 .

User Pair	Friendship	Like	Chat	$\mathcal{A}_{(i,j)}^{[s]}$
u_1, u_2	0	4	1	5
u_1, u_3	1	7	0	9
u_1, u_4	0	3	5	8
u_2, u_3	0	9	6	15
u_2, u_4	1	7	8	17
u_3, u_4	0	0	0	0

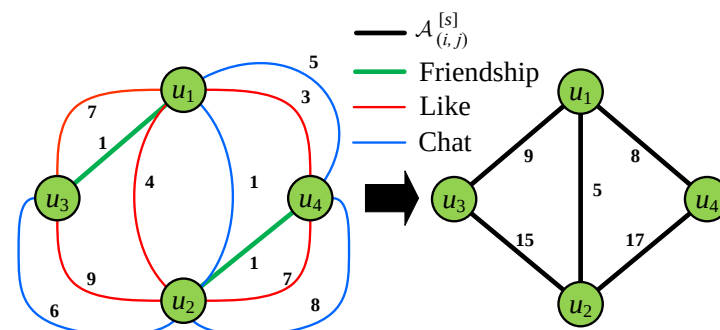


Figure 4. Illustration of interaction among u_1, u_2, u_3, u_4 : there is no interaction of any type between u_3 and u_4 , so there is no connection between them (the weight of edges is zero).

Example 2. The $\mathcal{A}_{(i,j)}^{[s]}$ of user u_2 and u_4 in the interaction is computed as $\mathcal{A}_{(u_2,u_4)}^{[s]} = 2 \times 1 + 1 \times 7 + 1 \times 8 = 17$.

(2) Number of common teams (View r): This layer represents shared group memberships. $\mathcal{A}_{(i,j)}^{[r]}$ quantifies explicit groups shared by users. Taking Academic-teams and Class-teams as specific instances in our case study, the affiliation matrix is defined as

$$\mathcal{A}_{(i,j)}^{[r]} = \mathcal{M}(i,j) + \mathcal{N}(i,j) \quad (3)$$

where $\mathcal{M}(i,j)$ denotes the number of common Academic-teams and $\mathcal{N}(i,j)$ denotes the number of common Class-teams in SCHOLAT of user i and j . We now consider the second scenario, where SCHOLAT users u_1, u_2, u_3, u_4 denote common team-ship using the public team member information in view r as shown in Table 3 and Figure 5.

Table 3. Common teams among u_1, u_2, u_3, u_4 .

User Pair	Common Team	Common Team Count	$\mathcal{A}_{(i,j)}^{[r]}$
u_1, u_2	$c-F$	1	1
u_1, u_3	$a-A$	1	1
u_1, u_4	$a-D, c-G$	2	2
u_2, u_3	$a-C, c-B, c-H$	3	3
u_2, u_4	$a-E$	1	1
u_3, u_4	$a-I$	1	1

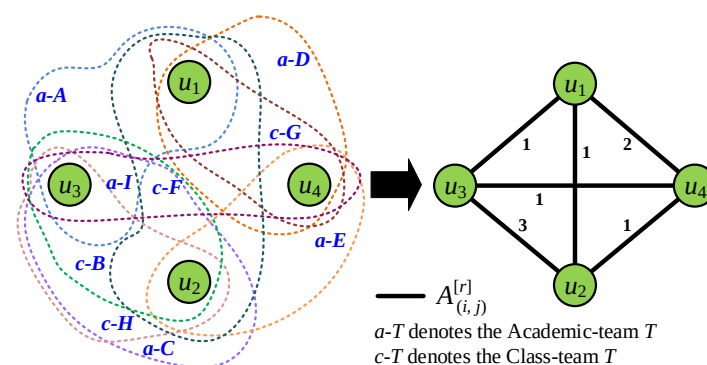


Figure 5. Illustration of common teams among u_1, u_2, u_3, u_4 : the number of common teams between nodes constitutes their connection at the Team-view.

Example 3. The user u_2 and u_3 have the one common Academic-team $a-C$, and two Class-teams such as $c-B$ and $c-H$, so $\mathcal{A}_{(i,j)}^{[r]}$ of user u_2 and u_3 in view r is computed as $\mathcal{A}_{(u_2,u_3)}^{[r]} = 1 + 2 = 3$.

4.2. Constructing a Matrix of Interaction Between Multi-View

The motivation for using Jaccard similarity is to act as a structural consistency filter. In social networks, a “team” relationship might exist without actual social interaction (noise). By computing the overlap of neighbors between the interaction view and the team view, we assign a high coupling strength only when the team structure aligns with the social structure. In this section, we use Jaccard similarity between view s and r to construct a matrix, which measures the cross-view clustering consistency. We need to define the data consistency for the two views as follows: (1) For the view s , let $\Gamma_{(i)}^{[s]}$ denote the friends of user i . (2) For the view r , let $\Gamma_{(j)}^{[r]}$ denote the teammates of user j , to this end, $\mathcal{J}_{(i,j)}^{[sr]}$ can be computed as

$$\mathcal{J}_{(i,j)}^{[sr]} = \text{Jaccard}(\Gamma_{(i)}^{[s]}, \Gamma_{(j)}^{[r]}) = \frac{|\Gamma_{(i)}^{[s]} \cap \Gamma_{(j)}^{[r]}|}{|\Gamma_{(i)}^{[s]} \cup \Gamma_{(j)}^{[r]}|} \quad (4)$$

In this paper, we only measure the cross-view clustering consistency for each user, so we set $i = j$ in Equation (4). Given a user u , the Jaccard similarity between the view s and r for user u can be computed as

$$\mathcal{J}^{[sr]}(u) = \mathcal{J}_{(u,u)}^{[sr]} = \text{Jaccard}(\Gamma_{(u)}^{[s]}, \Gamma_{(u)}^{[r]}) = \frac{|\Gamma_{(u)}^{[s]} \cap \Gamma_{(u)}^{[r]}|}{|\Gamma_{(u)}^{[s]} \cup \Gamma_{(u)}^{[r]}|} \quad (5)$$

Example 4. We can use the number of the common friends to measure the similarity between the User-view and Team-view. According to Figure 6, we can find (1) in the User-view: the friends of user u_0 are $\{u_1, u_2, u_3, u_4\}$. (2) In the Team-view: the user u_0 has the team a_1 , a_2 , and c_1 (the nodes with red border, where a denotes Academic-team, and c denotes Class-team), while the other members are $\{u_1, u_3, u_4\}$, $\{u_1, u_3, u_5\}$, and $\{u_1, u_2, u_5\}$. The friends of user u_0 are $\{u_1, u_3, u_4\} \cup \{u_1, u_3, u_5\} \cup \{u_1, u_2, u_5\} = \{u_1, u_2, u_3, u_4, u_5\}$. In the end, we can compute the Jaccard similarity between view s and r of user u_0 as $\mathcal{J}^{[sr]}(u_0) = \mathcal{J}_{(u_0, u_0)}^{[sr]} = \frac{|\{u_1, u_2, u_3, u_4\} \cap \{u_1, u_2, u_3, u_4, u_5\}|}{|\{u_1, u_2, u_3, u_4\} \cup \{u_1, u_2, u_3, u_4, u_5\}|} = 0.8$. The technical details for constructing matrix are as shown in Algorithm 1.

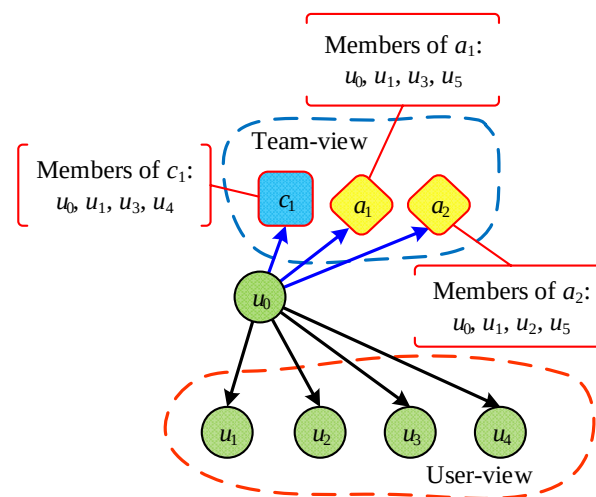


Figure 6. Illustration of data view consistency: Data view consistency across view s and view r of user u_0 : $\mathcal{J}^{[sr]}(u_0) = \frac{|\{u_1, u_2, u_3, u_4\} \cap \{u_1, u_2, u_3, u_4, u_5\}|}{|\{u_1, u_2, u_3, u_4\} \cup \{u_1, u_2, u_3, u_4, u_5\}|} = 0.8$.

Algorithm 1 The Coupling Matrix Construction Algorithm

Input: n denotes the number of nodes, and the neighbor sets in the user-view $\Gamma_{(i)}^{[s]}$ and team-view $\Gamma_{(i)}^{[r]}$ for all $i \in [1, n]$.

Output: The diagonal coupling matrix $\mathcal{J}^{[sr]} \in \mathbb{R}^{n \times n}$.

1: Initialize $\mathcal{J}^{[sr]}$ as an $n \times n$ zero matrix.

2: **for** $i = 1$ to n **do**

3: Compute $\mathcal{J}^{[sr]}(i) = \frac{|\Gamma_{(i)}^{[s]} \cap \Gamma_{(i)}^{[r]}|}{|\Gamma_{(i)}^{[s]} \cup \Gamma_{(i)}^{[r]}|}$ ▷ Using Equation (5)

4: $\mathcal{J}_{(i,i)}^{[sr]} \leftarrow \mathcal{J}^{[sr]}(i)$

5: **end for**

6: **return** $\mathcal{J}^{[sr]}$

4.3. The TaCD Method

There is currently no recognized community definition, which is only qualitatively considered to be a set of vertices with a “tight inside and loose outside” structure. In order to quantify the “tightness and looseness”, Newman and Girvan [32] propose the modularity $Q_M \in (0, 1)$, which is the most popular quality function for community detection. The modularity Q_M can be defined as

$$Q_M = \frac{1}{2m^{[*]}} \sum_{(i,j)} \left[\mathcal{A}_{(i,j)}^{[*]} - \frac{k_i k_j}{2m^{[*]}} \right] \delta(g_i, g_j) = \frac{1}{2m^{[*]}} \left[\sum_{(i,j)} \mathcal{A}_{(i,j)}^{[*]} - \frac{\sum_i k_i \sum_j k_j}{2m^{[*]}} \right] \delta(g_i, g_j) \quad (6)$$

where $\mathcal{A}_{(i,j)}^{[*]}$ represents the weight of the edge between node i and j . $k_i = \sum_j \mathcal{A}_{(i,j)}^{[*]}$ is the sum of the weights of the edges attached to vertex i . g_i is the community to which vertex i is assigned, and $\delta(x, y)$ is the Kronecker delta [38], e.g., $\delta(x, y) = 1$ if $x = y$, otherwise $\delta(x, y) = 0$. $m^{[*]} = \frac{1}{2} \sum_{(i,j)} \mathcal{A}_{(i,j)}^{[*]}$ denotes the number of inner edges of an adjacent matrix [30].

Part of the algorithm efficiency results from the fact that the gain in modularity ΔQ_M [2] obtained by moving an isolated node i into a given community C can easily be computed as

$$\Delta Q_M = \left[\frac{\sum_{in} + k_{(i,in)}}{2m^{[*]}} - \left(\frac{\sum_{tot} + k_i}{2m^{[*]}} \right)^2 \right] - \left[\frac{\sum_{in}}{2m^{[*]}} - \left(\frac{\sum_{tot}}{2m^{[*]}} \right)^2 - \left(\frac{k_i}{2m^{[*]}} \right)^2 \right] \quad (7)$$

where

- $\sum_{in}$: the sum of the weights of the links inside C ;
- $\sum_{tot}$: the sum of the weights of the links incident to nodes in C ;
- k_i : the sum of the weights of the links incident to node i ;
- $k_{i,in}$: the sum of the weights of the links from i to nodes in C ;
- $m^{[*]}$: the sum of the weights of all the links on the network.

The objective function of modularity can be computed using $\mathcal{A}$ and $\mathcal{C}$ as input. To solve this, the Generalized Louvain algorithm [39] can be used. For a more detailed explanation of its implementation, please refer to [40].

Using the steady-state probability distribution $\mathcal{P}_j^{[r]*} = \frac{\kappa_j^{[r]}}{2\mu}$, where $2\mu = \sum_j^{[r]} \kappa_{(i,j)}^{[r]}$, we obtain the multi-view null model in terms of the probability $\rho_i^{[s]} | \mathcal{P}_j^{[r]*}$ of sampling node i in view s conditional on whether the multi-view structure allows one to step from (j, r) to (i, s) , accounting for view s and r steps separately [30] as

$$\rho_i^{[s]} | \mathcal{P}_j^{[r]*} = \frac{1}{2\mu} \left(\frac{k_i^{[s]} k_j^{[r]}}{2m^{[s]} \kappa_j^{[r]}} \delta(s, r) + \frac{C_j^{[sr]} c_j^{[r]}}{c_j^{[r]} \kappa_j^{[r]}} \delta(i, j) \right) \kappa_j^{[r]} \quad (8)$$

$$Q^{[sr]} = \frac{1}{2\mu} \sum_{(i,j)}^{[sr]} \left[\left(\mathcal{A}_{(i,j)}^{[s]} - \gamma_v \frac{k_i^{[s]} k_j^{[s]}}{2m^{[s]}} \right) \delta(s, r) + C_j^{[sr]} \delta(i, j) \right] \delta(g_i^{[s]}, g_j^{[r]}) \quad (9)$$

where each network view s is represented by an adjacency $\mathcal{A}_{(i,j)}^{[s]}$ between node i and j , with view couplings $C_j^{[sr]}$ that connect node j in view r to itself in view s .

In TaCD, we use the Jaccard similarity to compute the coupling of view s and view r , combining with Equation (4), our proposed method to compute the modularity as

$$\tilde{Q}(g_i^{[s]}) = \frac{1}{2\mu} \sum_{[sr]} \sum_{(i,j)} \left[\left(\mathcal{A}_{(i,j)}^{[s]} - \gamma_v \frac{k_i^{[s]} k_j^{[s]}}{2m^{[s]}} \right) \delta(s, r) + \mathcal{J}_{(i,j)}^{[sr]} \delta(i, j) \right] \delta(g_i^{[s]}, g_j^{[r]}) \quad (10)$$

where $\mathcal{J}_{(i,j)}^{[sr]}$ can be computed via Equation (4). $\mathcal{A}_{(i,j)}^{[s]}$ denotes the matrix of view s . $g_i^{[s]}$ denotes the community label of node i in view s . $\delta(x, y)$ is the Kronecker delta, which is the same as Equation (6). $k_i^{[s]} = \sum_{j=1}^n \mathcal{A}_{(i,j)}^{[s]}$ and $m^{[s]} = \sum_{i=1}^n k_i^{[s]}$ [21]. The resolution associated with each view is dictated by γ_v . $\mathcal{B}_{(i,j)}^{[s]} = \mathcal{A}_{(i,j)}^{[s]} - \gamma_v \frac{k_i^{[s]} k_j^{[s]}}{2m^{[s]}}$, where $\mathcal{A}_{(i,j)}^{[s]}$ is the adjacency matrix for view s . $\mathcal{J}^{[sr]}$ represents the coupling between view s and r .

Where

- $k_i^{[s]} = \sum_j \mathcal{A}_{(i,j)}^{[s]}$: the degree (or weighted degree) of node i in view s .
- $2\mu^{[s]} = \sum_{ij} \mathcal{A}_{(i,j)}^{[s]}$: the total weight of all edges present in view s .
- γ_s : the resolution parameter associated with view s , which regulates the granularity of the detected communities (higher values of γ yield smaller communities).
- $\mathcal{C}_j^{[sr]}$: the coupling strength connecting node j between view s and view r . In our proposed method, this value is determined by the structural consistency (Jaccard similarity) between the views.
- c_i, c_j : the community assignments of node i and node j , respectively.
- $[sr]$: denotes the set of interactions between view s and view r .

For clarity, Algorithm 2 summarizes the main procedure of the proposed method. The process flow of the dataset construction and TaCD algorithm is shown in Figure 7.

Algorithm 2 The TaCD Algorithm

Input: n (the number of nodes),

$\mathcal{A}^{[s]}$ (Adjacency matrix for User-view),

$\mathcal{A}^{[r]}$ (Adjacency matrix for Team-view),

$\Gamma_{(i)}^{[s]}, \Gamma_{(i)}^{[r]}$ for $i \in [1, n]$ (Neighbor sets for both views),

γ_v (Resolution parameter).

Output: The community assignment vector $g \in \mathbb{N}^n$.

- 1: ▷ Step 1: Compute the cross-view coupling
 - 2: Compute the diagonal coupling matrix $\mathcal{J}^{[sr]}$ using Algorithm 1 with inputs $\Gamma_{(i)}^{[s]}$ and $\Gamma_{(i)}^{[r]}$.
 - 3: ▷ Step 2: Define the objective function
 - 4: Define the multi-view modularity objective function $\tilde{Q}$ using $\mathcal{A}^{[s]}$, $\mathcal{A}^{[r]}$, $\mathcal{J}^{[sr]}$, and γ_v as defined in Equation (10).
 - 5: ▷ Step 3: Optimize the objective function
 - 6: Optimize $\tilde{Q}$ using the Generalized Louvain algorithm [40] to find the final partition.
 - 7: ▷ Step 4: Obtain the final community assignments
 - 8: Let $g \in \mathbb{N}^n$ be the resulting community assignment vector, where g_i is the community ID for node i .
 - 9: **return** g
-

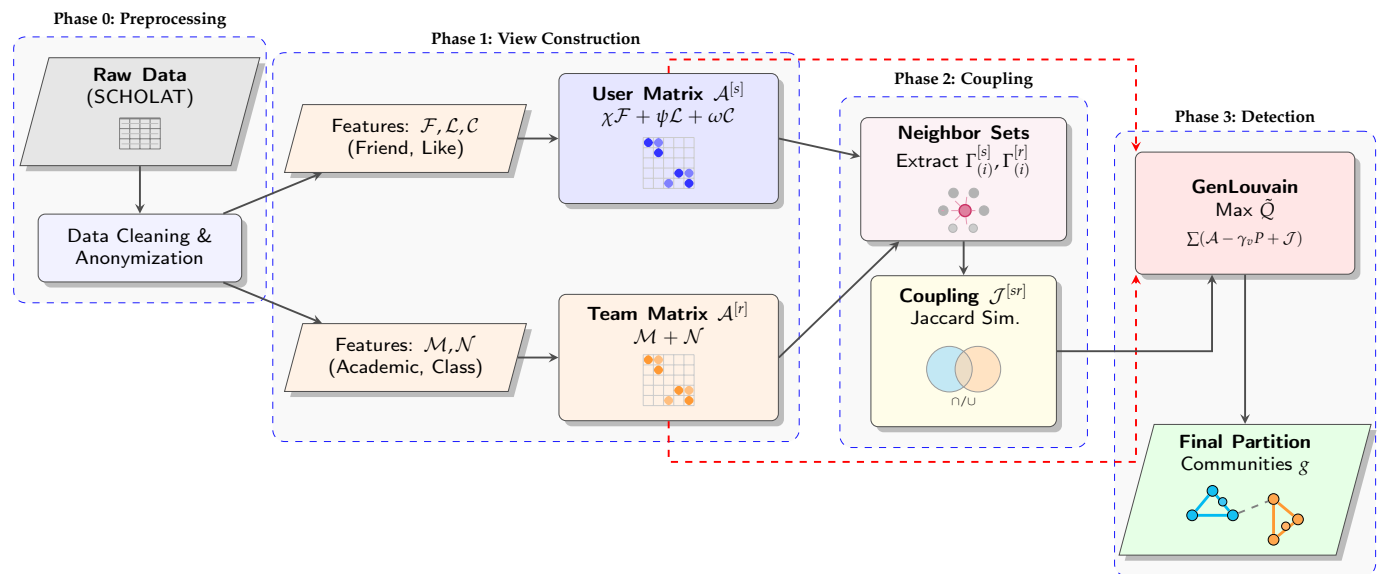


Figure 7. TaCD framework workflow: from raw data preprocessing to community detection through multi-view construction and cross-view coupling.

5. Experiments

5.1. Experimental Settings

In this set of experiments, we compare our approach with four existing state-of-the-art community detection techniques: AP (Affinity Propagation), NCut, Louvain, and VGAER. All experiments were conducted using a standard Personal Computer equipped with an Intel 3.4 GHz CPU and 16.0 GB RAM.

1. AP [41], which is a clustering algorithm based on “information transfer” between data points. AP algorithm does not need to determine the number of clusters before running the algorithm. The “exemplars” searched by AP algorithm, e.g., clustering centroids, are the actual points on the dataset and represent each class;
2. NCut [42], which is a clustering method based on segmentation. The normalized cuts criterion measures both the total dissimilarity between the different groups as well as the total similarity within the groups. We show that an efficient computational technique based on a generalized eigenvalue problem can be used to optimize this criterion;
3. Louvain [2,43], which is an algorithm that optimizes modularity based on multi-level (round-robin heuristic). The modularity function is originally used to measure the quality of community detection algorithm results, and it is able to characterize the closeness of the communities found;
4. VGAER [44], which is a novel unsupervised community detection method based on Variational Graph AutoEncoder (VGAE). Unlike traditional deep learning methods that typically reconstruct the adjacency matrix, VGAER reconstructs the modularity matrix to capture high-order community structures effectively. It represents the state-of-the-art in graph neural network-based unsupervised community detection.

5.2. Evaluation Measures

Since on each testing network, ground-truth labeling is provided for evaluating the clustering accuracy, we compare three widely used evaluation measures: ACC [45], NMI [46], and ARI [47].

1. Accuracy (ACC) shows what percentage of the samples you have predicted are correct [48]. Given the node $\mathcal{V}_i$, and π_i is the assigned label of the node $\mathcal{V}_i$, ζ_i is the real label of $\mathcal{V}_i$ in the dataset. The ACC can be computed as

$$ACC(\zeta, \pi) = \frac{1}{n} \sum_{i=1}^n \delta(\zeta_i, p_{\text{map}}(\pi_i)) \quad (11)$$

where $\delta(x, y)$ is the Kronecker delta, which is the same as Equation (6). $p_{\text{map}}(\pi_i)$ is the permutation mapping function that maps π_i of node $\mathcal{V}_i$ to the corresponding label in real community. n denotes the counts of nodes.

2. Normalized Mutual Information (NMI) is used for measuring the clustering accuracy based on the underlying class labels [49].

Given a network $\mathcal{G}$ of size n , the clustering labels π of c clusters, and actual class labels ζ of $\hat{c}$ classes, a confusion matrix is formed first, where entry (i, j) . $n_i^{(j)}$ gives the number of points in cluster i and class j . The NMI can be computed from the confusion matrix [42] as

$$NMI(\zeta, \pi) = \frac{2 \sum_{l=1}^c \sum_{h=1}^{\hat{c}} \frac{n_l^{(h)}}{n} \log \frac{n_l^{(h)} n}{\sum_{i=1}^c n_i^{(h)} \sum_{i=1}^{\hat{c}} n_l^{(i)}}}{H(\zeta) + H(\pi)} \quad (12)$$

where $H(\zeta) = -\sum_{j=1}^{\hat{c}} \frac{n^{(j)}}{n} \log \frac{n^{(j)}}{n}$ and $H(\pi) = -\sum_{i=1}^c \frac{n_i}{n} \log \frac{n_i}{n}$ are the Shannon entropy of cluster labels p and ζ , respectively, with n_i and $n^{(j)}$ denoting the number of points in cluster i and class j . A high NMI indicates the clustering and real labels match well. If $\pi = \zeta$, $NMI(\zeta, \pi) = 1$. If π and ζ are completely different, $NMI(\zeta, \pi) = 0$.

3. Apart from ACC and NMI, in the comparison results, we also use Adjusted Rand Index (ARI) to validate the algorithm. ARI has become one of the most successful cluster validation indices, and it is recommended as the index of choice for measuring agreement between two partitions in clustering analysis with different numbers of clusters. The ARI can be computed as

$$ARI(\zeta, \pi) = \frac{2(ad - bc)}{(a + b)(b + d) + (a + c)(c + d)} \quad (13)$$

where a denotes the number of sample pairs in the same group of cluster ζ and class π ; b denotes the same cluster in the original partition π , but the number of sample pairs in the cluster result ζ that are not in the same group; c denotes not in the same cluster π , but the number of sample pairs in the same in class ζ ; d denotes the number of sample pairs in both cluster π and class ζ , which are not in the same group.

5.3. Parameter Analysis

In our proposed method, the resolution parameter γ_v plays a crucial role, with a range of 0 to 2. We conducted a sensitivity analysis for this parameter to determine the optimal value for our experiments.

As visually illustrated in Figure 8, the performance metrics (NMI, ARI, and ACC) across all eight networks exhibit a consistent trend. Crucially, a stable performance plateau is observed in the highlighted interval $\gamma_v \in [1.0, 1.4]$. The mean performance curves (represented by the thick red lines) demonstrate that the algorithm has robust to small parameter variations within this region. Within this robust interval, while $\gamma_v = 1.3$ yields a marginal peak in NMI, $\gamma_v = 1.2$ demonstrates highly competitive stability and achieves a slightly stronger balance in ARI scores.

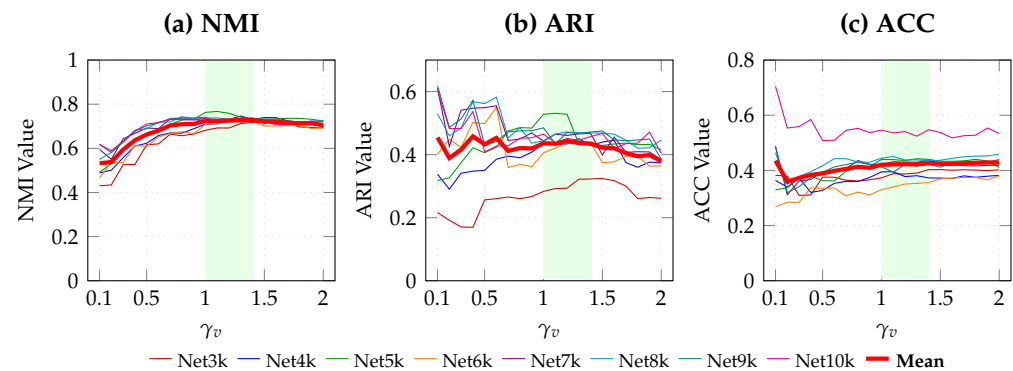


Figure 8. Parameter sensitivity analysis of γ_v on NMI (a), ARI (b), and ACC (c). We employ distinct colors for individual networks (Net-3k to Net-10k) and a thick red line for the mean performance. The shared legend is placed at the bottom. The shaded green area highlights the robust parameter interval $\gamma_v \in [1.0, 1.4]$.

Considering this balance between near-optimal NMI and strong ARI performance, we set $\gamma_v = 1.2$ as a robust and representative parameter to complete the community detection experiments with TaCD.

5.4. Complexity Analysis

The computational complexity of TaCD is determined by two main phases: the cross-view coupling construction (Algorithm 1) and the modularity optimization (Algorithm 2).

For the modularity optimization (Algorithm 2), we use the Generalized Louvain method. The complexity of the standard Louvain algorithm is known to be near-linear, often simplified to $O(M_{\text{graph}} + n_{\text{graph}})$ [43], where n_{graph} is the number of nodes and M_{graph} is the number of edges. In our multi-view model, the total number of nodes is n , and the total number of edges (intra-view plus inter-view couplings) is $M_s + M_r + n$. Thus, the optimization step has a complexity of $O((M_s + M_r + n) + n) = O(M_s + M_r + n)$.

For the coupling construction (Algorithm 1), we compute the Jaccard similarity for all n nodes. This is sometimes mistakenly assumed to be an $O(n^2)$ operation, which would involve computing a full similarity matrix. However, as shown in Algorithm 1, we only compute the n diagonal values ($i = j$). The complexity of computing the Jaccard similarity for a single node i is proportional to the size of its neighbor lists, $O(d_i^{[s]} + d_i^{[r]})$. Summing over all nodes, the total complexity for this phase is $\sum_{i=1}^n O(d_i^{[s]} + d_i^{[r]}) = O(M_s + M_r)$.

Therefore, the total complexity of TaCD is $O(M_s + M_r)$ (Coupling) + $O(M_s + M_r + n)$ (Optimization). Letting $M = M_s + M_r$ be the total number of intra-view edges, the overall complexity simplifies to $O(M + n)$. This near-linear complexity is highly scalable. The experimental running times shown in Figure 9 confirm this gentle, near-linear growth as the network size increases.

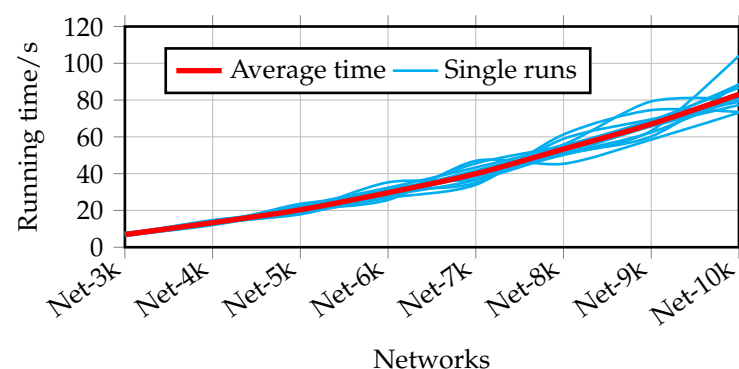


Figure 9. The running time of TaCD on different scale networks with $\gamma \in [1, 2]$.

5.5. Comparison Results

With the proposed method, we utilize both the interaction information matrix and the team information matrix as inputs, serving as two distinct views for the network. In contrast, existing methods typically rely solely on the interaction matrix. The community structures detected by each method are compared against the ground-truth communities in the SCHOLAT dataset.

For the stochastic methods (NCut, Louvain, VGAER, and TaCD), we conducted ten independent runs for each network and reported the mean results to ensure statistical reliability. The deterministic algorithm AP was run once. The comprehensive results are presented in Table 4, where the best results are highlighted in bold and the second-best results are underlined.

Table 4. Comparison of ACC, NMI, and ARI metrics across eight networks. The best results are highlighted in bold, and the second-best results are underlined.

	Networks	Algorithms				
		AP	NCut	Louvain	VGAER	Ours (TaCD)
ACC	Net-3k	0.0863	0.1548	0.3263	<u>0.3500</u>	0.3869
	Net-4k	0.0740	0.1008	0.2754	<u>0.3278</u>	0.3776
	Net-5k	0.0724	0.0776	0.2565	<u>0.3928</u>	0.4360
	Net-6k	0.0657	0.0741	0.2355	<u>0.3367</u>	0.3507
	Net-7k	0.0651	0.0857	0.2490	<u>0.4011</u>	0.4302
	Net-8k	0.0640	0.0701	0.2427	<u>0.4160</u>	0.4375
	Net-9k	0.0599	0.0579	0.2342	<u>0.3802</u>	0.4217
	Net-10k	0.0811	0.1166	0.3525	<u>0.5122</u>	0.5406
NMI	Net-3k	0.4027	0.2227	<u>0.5522</u>	0.4098	0.6921
	Net-4k	0.3767	0.1578	<u>0.5256</u>	0.4870	0.7309
	Net-5k	0.3671	0.1607	<u>0.5118</u>	0.4992	0.7584
	Net-6k	0.3299	0.1634	0.4709	<u>0.4938</u>	0.7216
	Net-7k	0.3268	0.1441	0.4683	<u>0.5350</u>	0.7292
	Net-8k	0.3183	0.1194	0.4576	<u>0.5437</u>	0.7324
	Net-9k	0.3165	0.1245	0.4547	<u>0.5323</u>	0.7241
	Net-10k	0.3288	0.1181	0.4602	<u>0.5229</u>	0.7211
ARI	Net-3k	0.0561	0.0489	0.2485	<u>0.2846</u>	0.2935
	Net-4k	0.0583	0.0273	0.2628	<u>0.4172</u>	0.4340
	Net-5k	0.0517	0.0442	0.2701	<u>0.4108</u>	0.5279
	Net-6k	0.0406	0.0884	0.2834	<u>0.3777</u>	0.4369
	Net-7k	0.0391	0.0424	0.2856	<u>0.4467</u>	0.4610
	Net-8k	0.0366	0.0129	0.2734	<u>0.4579</u>	0.4720
	Net-9k	0.0372	0.0187	0.2985	0.4711	<u>0.4699</u>
	Net-10k	0.0348	0.0122	0.2795	0.4603	<u>0.4512</u>

As presented in Table 4 and Figure 10, we conducted a comprehensive evaluation including the advanced deep learning-based method, VGAER [44].

In terms of Accuracy (ACC), the proposed TaCD method consistently outperforms all baselines. It achieves a peak ACC of 0.5406 on Net-10k, which is notably higher than the deep learning-based VGAER (0.5122) and significantly surpasses traditional methods like AP (0.0811). This indicates that integrating team affiliations effectively corrects misclassified nodes in boundary regions.

Regarding Normalized Mutual Information (NMI), TaCD demonstrates its most significant advantage. By establishing a structural coupling between the user view and the team view, TaCD achieves NMI scores consistently in the range of 0.69–0.75. In comparison, VGAER, despite being a powerful GNN-based reconstruction method, fluctuates between

0.40 and 0.54. This validates that our multi-view modularity approach captures the global community structure more accurately than single-view reconstruction methods, which may struggle with the sparsity of social interaction data.

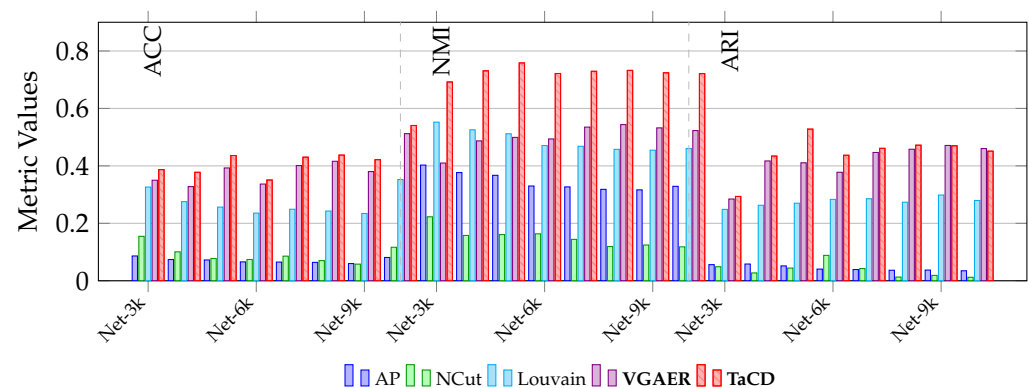


Figure 10. Comparison of ACC, NMI, and ARI metrics across eight networks.

Interestingly, in the Adjusted Rand Index (ARI) metric, VGAER shows strong competitiveness, particularly on larger networks. As shown in Table 4, VGAER achieves ARI scores very close to TaCD, and even slightly surpasses TaCD on Net-9k (0.4711 vs. 0.4699) and Net-10k (0.4603 vs. 0.4512). This suggests that while TaCD is superior in global structure identification (NMI), VGAER is highly effective in pairwise classification decisions on large-scale graphs. Nevertheless, TaCD remains the best performing method overall, securing the highest scores across the vast majority of metrics and datasets.

5.6. Case Study and Visualization

The eight networks cluster results obtained by TaCD are shown in Figure 11, and the community information after processing with TaCD method is shown in Table 5. Next, we take a closer look at the experimental results. As Figure 12 shows, the big red node is the centroid of community, whose degree of edges is the highest. Through examples, what sets it apart from other community detection algorithms is that, TaCD is more sensitive to team information on the dataset networks and can fully utilize this information for more accurate community partitioning of nodes. Accurate segmentation can help us better develop recommendation algorithms on social networking platforms, thereby helping users create their social circles more quickly on the platform.

Table 5. The community information after processing with TaCD.

Networks	Nodes	Edges	Communities
Net-3k	3000	575,155	93
Net-4k	4000	897,233	107
Net-5k	5000	1,179,567	127
Net-6k	6000	1,888,510	121
Net-7k	7000	1,839,713	131
Net-8k	8000	2,693,429	139
Net-9k	9000	3,593,428	166
Net-10k	10,000	4,082,118	196

In addition, we find that among the eight clusters, there is a relatively large purple classification. By comparing the data in platform, we find that the classification is the greatest activity in SCHOLAT, while the name of largest class is *South China Normal University, Guangzhou, China*. In addition, the community is the birthplace of SCHOLAT.

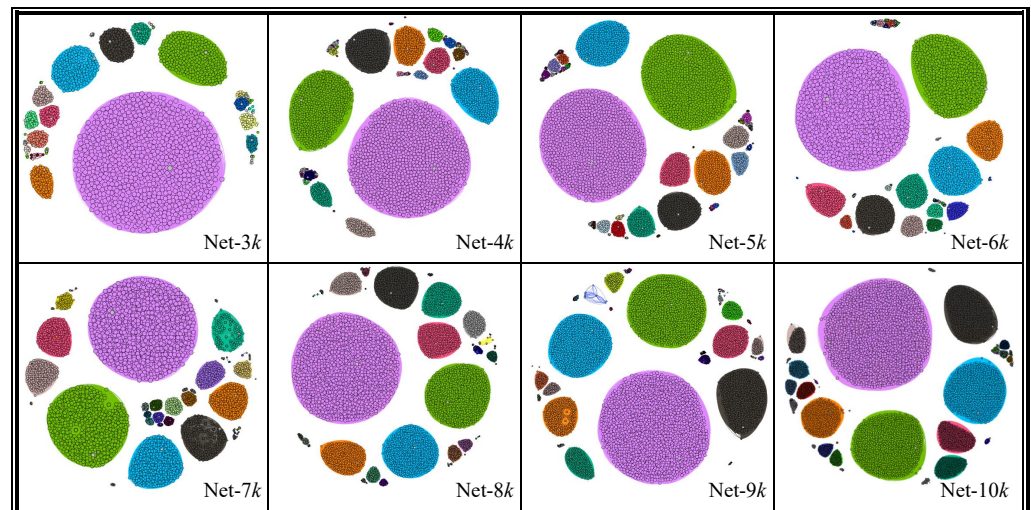


Figure 11. Results of case study and visualization: we can see that all nodes are clearly divided into several communities. There is a relatively large purple classification, whose name is *South China Normal University, Guangzhou, China*, that is the birthplace of SCHOLAT.

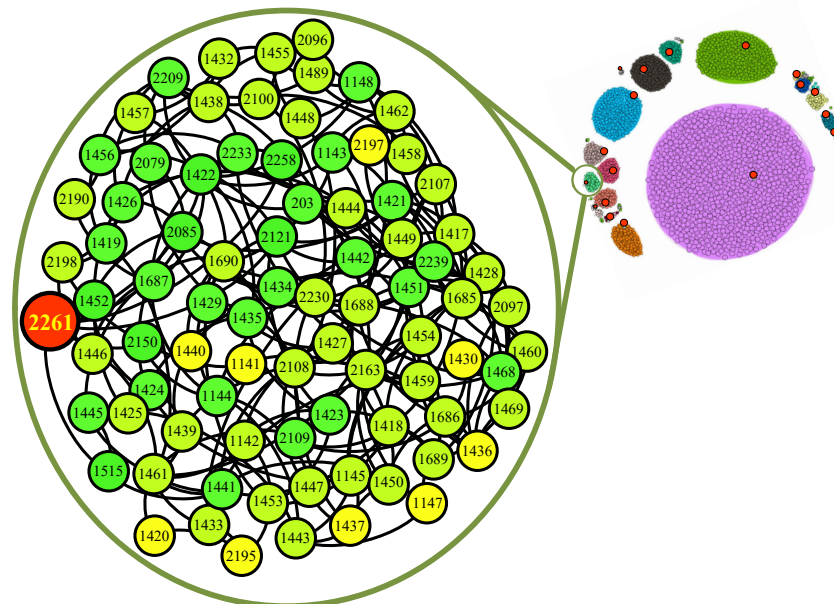


Figure 12. Community detection with TaCD on Net-3k: the big red node is the centroid of the community. In the enlarged view of a community, we can see that the centroid is node 2261. Otherwise, the community id is 33, and the name is *Guangdong Ocean University, China* in the real world.

5.7. Ablation Study

To verify that the performance gains are specifically driven by the team-aware modeling and not merely by parameter tuning or increased edge density, we conducted a comprehensive ablation study. We defined a baseline variant named TaCD-NoTeam, which utilizes only the Interaction Layer (User-view) by setting the cross-view coupling matrix to zero. We compared this baseline with the proposed TaCD-Full model, which fully integrates the Affiliation Layer (Team-view).

The comparative results across all eight networks (Net-3k to Net-10k) are visualized in Figure 13. In the figure, the green bars represent the performance of TaCD-NoTeam, while the red bars represent TaCD-Full. It is evident that the inclusion of team information yields a consistent and significant performance boost across all evaluation metrics (NMI, ARI, and ACC).

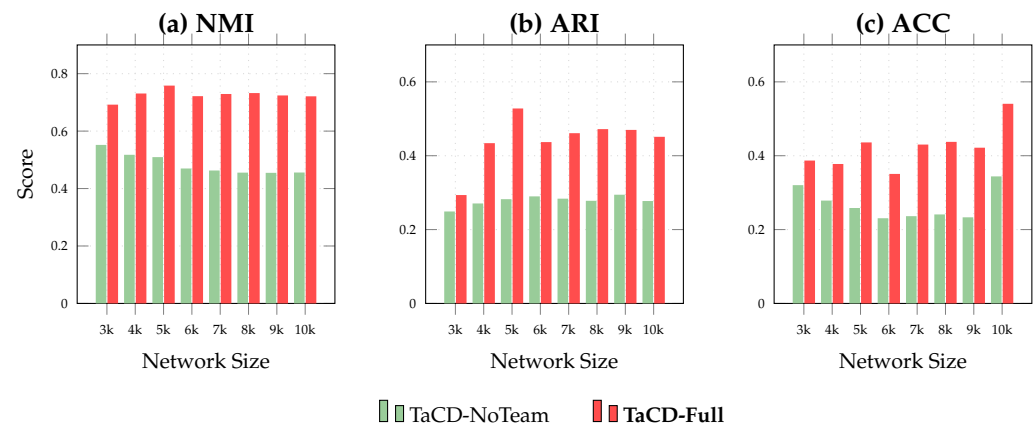


Figure 13. Ablation study results visualized as bar charts across all networks. The comparison highlights the performance gap between the baseline TaCD-NoTeam (green bars) and the proposed TaCD-Full (red bars) for NMI (a), ARI (b), and ACC (c). The consistent height difference across all datasets confirms the robustness of the team-aware strategy.

As illustrated in the figure, removing the team view results in a significant performance drop. This confirms that the explicit affiliation information contributes fundamentally to the detection accuracy.

6. Conclusions and Future Work

In this paper, we propose a novel method for community detection. The SCHOLAT dataset is combined with the characteristics of team attributes, and a multi-view approach is used to build a multi-layer community detection model based on team-aware named TaCD. The comparisons with other counterpart methods shows that the proposed method outperforms them. Furthermore, the dataset used for analysis is freely available to the research community to conduct further experiments on community detection. While our current implementation focuses on hard partitioning, which is essential for defining primary administrative units in organizations, we acknowledge that social communities often overlap. Future work will extend this multi-view framework to support overlapping community detection.

In the future, we plan to use more views beyond user-interactions view and team-relations view, and integrate more relationships to further enhance our method. For example, we can add time attributes, including interaction time, team creation time, and becoming friends time, which can form a new view. We can also apply the method to other datasets such as WebKB, SNAP_Pokec, DBLP, and so on, which consist of two or more types of relational data.

In addition, real networks always exhibit complex changes, and our methods are capable of conducting research on these dynamic networks. For example, our proposed multi-view approach can be used to solve community discovery problems in dynamic networks by treating networks before and after changes as different views.

Author Contributions: Conceptualization, C.F., F.T., L.H. and R.L.; methodology, C.F. and F.T.; software, C.F.; validation, C.F., F.T. and C.Y.; formal analysis, C.F.; writing—original draft preparation, C.F. and F.T.; writing—review and editing, L.H., C.Y. and R.L.; project administration, C.F., L.H. and R.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Research Project of Guangdong Provincial Administration of Traditional Chinese Medicine (20242047), Teaching Quality Enhancement Project of Guangdong Pharmaceutical University (2022), National Natural Science Foundation of China (62407016).

Data Availability Statement: The data are openly available in a public repository. The code is available at <https://www.scholat.com/research/tacd>, accessed on 13 November 2025.

Acknowledgments: This work reported in this paper was carried out in the SCHOLAT R&D Team. The numerical optimization was performed using MATLAB R2023b (MathWorks, Natick, MA, USA).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ma, H.; Yang, H.; Zhou, K.; Zhang, L.; Zhang, X. A local-to-global scheme-based multi-objective evolutionary algorithm for overlapping community detection on large-scale complex networks. *Neural Comput. Appl.* **2020**, *33*, 5135–5149. [\[CrossRef\]](#)
2. Kalyanaraman, A.; Halappanavar, M.; Chavarría-Miranda, D.; Lu, H.; Duraisamy, K.; Pande, P.P. Fast Uncovering of Graph Communities on a Chip: Toward Scalable Community Detection on Multicore and Manycore Platforms. *Found. Trends Electron. Des. Autom.* **2016**, *10*, 145–247. [\[CrossRef\]](#)
3. Yang, Y.; Sun, Y.; Wang, Q.; Liu, F.; Zhu, L. Fast Power Grid Partition for Voltage Control with Balanced-Depth-Based Community Detection Algorithm. *IEEE Trans. Power Syst.* **2021**, *37*, 1612–1622. [\[CrossRef\]](#)
4. Truong, H.B.; Ivanovic, M.; Tran, V.C. Community Detection Methods Based on Exploiting Attributes and Interactions on Social Networks: A Survey and Future Directions. *Vietnam J. Comput. Sci.* **2025**, *12*, 1–19. [\[CrossRef\]](#)
5. Tamburri, D.A.; Lago, P.; Van Vliet, H. Uncovering latent social communities in software development. *IEEE Softw.* **2013**, *30*, 29–36. [\[CrossRef\]](#)
6. Zhang, W.; Nie, L.; Jiang, H.; Chen, Z.; Liu, J. Developer social networks in software engineering: Construction, analysis, and applications. *Sci. China Inf. Sci.* **2014**, *57*, 1–23. [\[CrossRef\]](#)
7. Ozawa, Y.; Saito, R.; Fujimori, S.; Kashima, H.; Ishizaka, M.; Yanagawa, H.; Miyamoto-Sato, E.; Tomita, M. Protein complex prediction via verifying and reconstructing the topology of domain-domain interactions. *BMC Bioinf.* **2010**, *11*, 350. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Ahn, Y.-Y.; Bagrow, J.P.; Lehmann, S. Link communities reveal multiscale complexity in networks. *Nature* **2010**, *466*, 761. [\[CrossRef\]](#)
9. Bleicher, L.; Lemke, N.; Garratt, R.C. Using amino acid correlation and community detection algorithms to identify functional determinants in protein families. *PLoS ONE* **2011**, *6*, e27786. [\[CrossRef\]](#)
10. Ozer, M.; Kim, N.; Davulcu, H. Community detection in political Twitter networks using Nonnegative Matrix Factorization methods. In Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, San Francisco, CA, USA, 18–21 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 81–88.
11. Yang, J.; McAuley, J.; Leskovec, J. Community detection in networks with node attributes. In Proceedings of the 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, 7–10 December 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1151–1156.
12. Ferrara, E. A large-scale community structure analysis in Facebook. *EPJ Data Sci.* **2012**, *1*, 9. [\[CrossRef\]](#)
13. Deitrick, W.; Hu, W. Mutually enhancing community detection and sentiment analysis on twitter networks. *J. Data Anal. Inf. Process.* **2013**, *1*, 19–29. [\[CrossRef\]](#)
14. El-Moussaoui, M.; Hanine, M.; Kartit, A.; Villar, M.G.; Garay, H.; De La Torre, I. A systematic review of deep learning methods for community detection in social networks. *Front. Artif. Intell.* **2025**, *8*, 1572645. [\[CrossRef\]](#)
15. Yang, J.; Leskovec, J. Overlapping community detection at scale: A nonnegative matrix factorization approach. In Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, Rome, Italy, 4–8 February 2013; ACM: New York, NY, USA, 2013; pp. 587–596.
16. Wang, C.-D.; Lai, J.-H.; Philip, S.Y. NEIWalk: Community discovery in dynamic content-based networks. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1734–1748. [\[CrossRef\]](#)
17. Zhang, H.; Wang, C.-D.; Lai, J.-H.; Philip, S.Y. Community detection using multilayer edge mixture model. *Knowl. Inf. Syst.* **2018**, *60*, 757–779. [\[CrossRef\]](#)
18. Tang, F.; Zhu, J.; He, C.; Fu, C.; He, J.; Tang, Y. SCHOLAT: An innovative academic information service platform. In Proceedings of the Australasian Database Conference, Sydney, Australia, 28–29 September 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 453–456.
19. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905. [\[CrossRef\]](#)
20. Psorakis, I.; Roberts, S.; Ebdon, M.; Sheldon, B. Overlapping community detection using bayesian non-negative matrix factorization. *Phys. Rev. E* **2011**, *83*, 066114. [\[CrossRef\]](#)
21. Huang, L.; Wang, C.-D.; Chao, H.-Y. A harmonic motif modularity approach for multi-layer network community detection. In Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM), Singapore, 17–20 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1043–1048.

22. Rosvall, M.; Bergstrom, C.T. Maps of random walks on complex networks reveal community structure. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 1118–1123. [\[CrossRef\]](#)
23. Raghavan, U.N.; Albert, R.; Kumara, S. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E* **2007**, *76*, 036106. [\[CrossRef\]](#)
24. Leung, I.X.Y.; Hui, P.; Lio, P.; Crowcroft, J. Towards real-time community detection in large networks. *Phys. Rev. E* **2009**, *79*, 066107. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Ma, X.; Tan, S.; Xie, X.; Zhong, X.; Deng, J. Joint multi-label learning and feature extraction for temporal link prediction. *Pattern Recognit.* **2022**, *121*, 108216. [\[CrossRef\]](#)
26. Mahmood, A.; Small, M. Subspace based network community detection using sparse linear coding. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 801–812. [\[CrossRef\]](#)
27. Jin, J. Fast community detection by SCORE. *Ann. Stat.* **2015**, *43*, 57–89. [\[CrossRef\]](#)
28. Zhao, Z.; Li, C.; Zhang, X.; Chiclana, F.; Viedma, E.H. An incremental method to detect communities in dynamic evolving social networks. *Knowl.-Based Syst.* **2019**, *163*, 404–415. [\[CrossRef\]](#)
29. Bazzi, M. Community Structure in Temporal Multilayer Networks, and Its Application to Financial Correlation Networks. Ph.D. Thesis, University of Oxford, Oxford, UK, 2015.
30. Mucha, P.J.; Richardson, T.; Macon, K.; Porter, M.A.; Onnela, J.-P. Community structure in time-dependent, multiscale, and multiplex networks. *Science* **2010**, *328*, 876–878. [\[CrossRef\]](#)
31. De Domenico, M.; Lancichinetti, A.; Arenas, A.; Rosvall, M. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X* **2015**, *5*, 011027. [\[CrossRef\]](#)
32. Newman, M.E.J.; Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* **2004**, *69*, 026113. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Wen, Y.-M.; Huang, L.; Wang, C.-D.; Lin, K.-Y. Direction recovery in undirected social networks based on community structure and popularity. *Inf. Sci.* **2019**, *473*, 31–43. [\[CrossRef\]](#)
34. Delvenne, J.-C.; Yaliraki, S.N.; Barahona, M. Stability of graph communities across time scales. *Proc. Natl. Acad. Sci. USA* **2010**, *107*, 12755–12760. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Wang, H.; Zhang, W.; Ma, X. Contrastive and adversarial regularized multi-level representation learning for incomplete multi-view clustering. *Neural Netw.* **2024**, *172*, 106102. [\[CrossRef\]](#)
36. Li, D.; Kosugi, S.; Zhang, Y.; Okumura, M.; Xia, F.; Jiang, R. Revisiting Dynamic Graph Clustering via Matrix Factorization. In Proceedings of the ACM on Web Conference 2025 (WWW '25), Sydney, Australia, 28 April–2 May 2025; ACM: New York, NY, USA, 2025; pp. 1342–1352. [\[CrossRef\]](#)
37. Li, D.; Ma, X.; Gong, M. Joint Learning of Feature Extraction and Clustering for Large-Scale Temporal Networks. *IEEE Trans. Cybern.* **2023**, *53*, 1653–1666. [\[CrossRef\]](#)
38. Porter, M.A.; Onnela, J.-P.; Mucha, P.J. Communities in networks. *Not. Am. Math. Soc.* **2009**, *56*, 1082–1097.
39. De Meo, P.; Ferrara, E.; Fiumara, G.; Provetti, A. Generalized louvain method for community detection in large networks. In Proceedings of the 2011 11th International Conference on Intelligent Systems Design and Applications, Córdoba, Spain, 22–24 November 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 88–93.
40. Jeub, L.G.S.; Bazzi, M.; Jutla, I.S.; Mucha, P.J. A Generalized Louvain Method for Community Detection Implemented in MATLAB. 2011. Available online: <https://github.com/GenLouvain/GenLouvain> (accessed on 25 October 2025).
41. Frey, B.J.; Dueck, D. Clustering by passing messages between data points. *Science* **2007**, *315*, 972–976. [\[CrossRef\]](#)
42. Dhillon, I.S.; Guan, Y.; Kulis, B. Kernel k-means: Spectral clustering and normalized cuts. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WC, USA, 22–25 August 2004; ACM: New York, NY, USA, 2004; pp. 551–556.
43. Blondel, V.D.; Guillaume, J.-L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.* **2008**, *2008*, P10008. [\[CrossRef\]](#)
44. Qiu, C.; Huang, Z.; Xu, W.; Li, H. VGAER: Graph Neural Network Reconstruction Based Community Detection. In Proceedings of the 6th International Workshop on Deep Learning on Graphs (DLG-AAAI'22), Vancouver, BC, Canada, 28 February–1 March 2022; ACM: New York, NY, USA, 2022.
45. Baldi, P.; Brunak, S.; Chauvin, Y.; Andersen, C.A.F.; Nielsen, H. Assessing the accuracy of prediction algorithms for classification: An overview. *Bioinformatics* **2000**, *16*, 412–424. [\[CrossRef\]](#) [\[PubMed\]](#)
46. Danon, L.; Diaz-Guilera, A.; Duch, J.; Arenas, A. Comparing community structure identification. *J. Stat. Mech. Theory Exp.* **2005**, *2005*, P09008. [\[CrossRef\]](#)
47. Labatut, V. Generalized Measures for the Evaluation of Community Detection Methods. *Comput. Sci.* **2013**, *2*, 44–63.

48. Li, Y.; Jia, C.; Yu, J. A parameter-free community detection method based on centrality and dispersion of nodes in complex networks. *Phys. A* **2015**, *438*, 321–334. [[CrossRef](#)]
49. Wang, C.-D.; Lai, J.-H.; Suen, C.Y.; Zhu, J.-Y. Multi-exemplar affinity propagation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2223–2237. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.