Advancing Community Detection with Graph Convolutional Neural Networks: Bridging Topological and Attributive Cohesion

Anjali de Silva[1] , **Gang Chen**[1] , **Hui Ma**[1] , **Seyed Mohammad Nekooei**[2] and **Xingquan Zuo**[3]
[1]Victoria University of Wellington, Wellington, New Zealand
[2]Goldenset Collective
[3]School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
{desilanja, aaron.chen, Hui.Ma}@ecs.vuw.ac.nz, mohammad@goldenset.com, zuoxq@bupt.edu.cn

Abstract

Community detection, a vital technology for real-world applications, uncovers cohesive node groups (communities) by leveraging both topological and attribute similarities in social networks. However, existing Graph Convolutional Networks (GCNs) trained to maximize modularity often converge to suboptimal solutions. Additionally, directly using human-labeled communities for training can undermine topological cohesiveness by grouping disconnected nodes based solely on node attributes. We address these issues by proposing a novel *Topological and Attributive Similarity-based Community detection* (TAS-Com) method. TAS-Com introduces a novel loss function that exploits the highly effective and scalable Leiden algorithm to detect community structures with global optimal modularity. Leiden is further utilized to refine human-labeled communities to ensure connectivity within each community, enabling TAS-Com to detect community structures with desirable trade-offs between modularity and compliance with human labels. Experimental results on multiple benchmark networks confirm that TAS-Com can significantly outperform several state-of-the-art algorithms.

1 Introduction

Community Detection (CD) in social networks is an active research area with immense practical implications across numerous fields like marketing, sociology, and public health [Chunaev, 2020; Wu *et al.*, 2020]. Unveiling hidden structures within complex networks drives transformative innovations that help to shape the future of society [Su *et al.*, 2022; He *et al.*, 2021a]. Real-world applications of CD are vast and impactful, including information spreading [Karataş and Şahin, 2018], dimensionality reduction, and product recommendation [Moradi *et al.*, 2015].

Many existing methods for CD are mainly based on network topology [Traag *et al.*, 2019; de Silva *et al.*, 2023; Jin *et al.*, 2021]. However, real-world networks often include node *attributes* that are crucial for identifying meaningful communities. Networks with these attributes are known as *attributed networks* [Chunaev, 2020; Bhowmick *et al.*, 2024; Xie *et al.*, 2021].

In an attributed network, a *community* is a group of nodes that are densely connected and share strong attribute similarities [Bhowmick *et al.*, 2024; Zhu *et al.*, 2024; Luo and Yan, 2020]. Such communities capture both topological and attribute cohesiveness, making them meaningful representations of real-world groups where both connections and shared characteristics are important. A set of non-overlapping communities that completely covers the whole attributed network is known as the *community structure* of the network.

CD in attributed networks has been approached through various methods, including heuristic-based [Combe *et al.*, 2015], evolutionary computation (EC)-based [Guo *et al.*, 2024], and learning-based techniques [Jin *et al.*, 2019; Bhowmick *et al.*, 2024; Ju *et al.*, 2023; He *et al.*, 2022]. Among these, learning-based methods have gained significant attention in recent literature [Sun *et al.*, 2019], with Graph Convolutional Neural Networks (GCNs) emerging as the leading solution. GCNs leverage deep learning to integrate topological and attribute information through layer-wise propagation [Tsitsulin *et al.*, 2023; He *et al.*, 2021b], offering a precise understanding of network topology and providing a comprehensive view of social networks. Despite their strengths, state-of-the-art GCN approaches still face two critical challenges.

The **first** issue lies in the challenge of directly training GCNs to maximize modularity, which often results in suboptimal community structures [Chunaev, 2020]. Although maximizing modularity effectively promotes topological cohesiveness [Bhowmick *et al.*, 2024; Tsitsulin *et al.*, 2023], the inherent complexity of the modularity function makes gradient-based optimization prone to local optima, leading to poor results. This difficulty in achieving globally optimal solutions highlights the limitations of modularity as a direct training objective and underscores the need to use a different loss function that makes it more straightforward to maximize modularity.

The **second** issue arises from using human-labeled communities to train GCNs, as these labels are typically based solely on node attributes while overlooking topological cohesiveness. This omission, a critical factor for high-quality community structures, often leads to poorly formed communities with disconnected nodes, significantly reducing modularity. This problem highlights the necessity of integrating topological information with node attributes to enhance GCNs' ability to produce well-formed and cohesive community structures.

To address these issues, we propose a novel *Topological and Attributive Similarity-based Community detection* (TAS-Com[1]). TAS-Com introduces a newly designed loss function to guide GCN training. Our innovative loss function exploits the highly effective and scalable Leiden algorithm [Traag *et al.*, 2019] (see Appendix M for more details) to identify community structures with global optimal modularity. Leiden is further utilized to refine human-labeled communities to ensure connectivity within each community, enabling TAS-Com to detect community structures with desirable trade-offs between modularity and compliance with human labels. The key contributions of this paper are as follows:

- We are the first to address the limitations of directly using human-provided community labels to train GCNs, which often compromise CD due to lack of topological cohesiveness. To overcome this, we propose a novel Leiden-based method to refine human-labeled communities, ensuring they are both cohesive and free from disconnected nodes that degrade CD quality, thereby enhancing their suitability for effective GCN training.
- We are the first to address the limitations of directly training GCNs to maximize modularity. We propose a novel *modularity-based similarity loss*, enhanced by the Leiden algorithm, to effectively guide GCNs toward discovering community structures with globally optimal modularity. Additionally, we integrate this loss with a *refined human-label-based similarity loss*, significantly improving the GCN's ability to identify community structures that achieve both high modularity and strong alignment with human-labeled communities.
- We conduct extensive experiments on many real-world attributed benchmark networks. The experiment results confirm that TAS-Com can significantly outperform multiple state-of-the-art approaches for CD across most of these benchmark networks.

2 Related Work

CD has traditionally focused on graph topology, but recent research [Chunaev, 2020] started to consider node attributes to improve CD in attributed networks. Heuristic approaches like I-Louvain [Combe *et al.*, 2015] use inertia-based measures for attribute similarity but often fail to find globally optimal structures. EC methods, such as @NetGA [Pizzuti and Socievole, 2018], optimize fitness functions combining attributes and connectivity, while multi-objective algorithms [Li *et al.*, 2017] refine these strategies. However, these approaches rely on manually designed similarity measures, overlooking intrinsic node relationships and highlighting the need for machine learning techniques to learn high-level node embeddings for more accurate CD [Su *et al.*, 2022].

The emergence of advanced Graph Neural Networks (GNNs) has significantly shifted recent research toward learning-based approaches, showcasing the critical importance of leveraging these technologies for more effective CD [Chunaev, 2020; Zhu *et al.*, 2024; Yang *et al.*, 2023; Liu *et al.*, 2022; Xia *et al.*, 2021]. These approaches generally perform CD through two consecutive steps: 1) node representation learning, and 2) node grouping/clustering based on the learned high-level node embeddings [Zhou *et al.*, 2023; Tsitsulin *et al.*, 2023; Bhowmick *et al.*, 2024].

The most commonly used GNNs for CD are Autoencoders (AEs) [Zhu *et al.*, 2024] and GCNs [Bhowmick *et al.*, 2024]. AEs are proficient at extracting node semantic information [Liu *et al.*, 2024; Sun *et al.*, 2020; Kumar *et al.*, 2023]. Notable AE-based approaches include DNR [Yang *et al.*, 2016] and CDBNE [Zhou *et al.*, 2023]. Despite their recent success, AEs are not specifically designed to process graph data. In contrast, GCNs can effectively handle node information at both topological and attribute levels [Zhu *et al.*, 2024].

Most GCN approaches for CD conduct either supervised or semi-supervised learning based on ground truth community labels [Bhowmick *et al.*, 2024]. A few recent studies, like SGCN [Wang *et al.*, 2021] and DMoN [Tsitsulin *et al.*, 2023], further demonstrated the importance of unsupervised learning. For example, Zhu et al. [Zhu *et al.*, 2024] introduced DyFSS, an unsupervised approach that dynamically fuses embeddings from multiple self-supervised tasks with node-specific weights, effectively balancing attribute and structural information to enhance clustering accuracy and robustness. However, DyFSS's reliance on hyperparameter tuning, such as pseudo-label thresholds, limits its generalizability across diverse graph datasets without extensive adjustments.

Existing unsupervised methods largely focus on optimizing modularity, often neglecting explicit optimization of attribute similarity [Tsitsulin *et al.*, 2023]. While DGCluster [Bhowmick *et al.*, 2024] optimizes both modularity and attribute similarity, its maximization of modularity frequently leads to suboptimal community structures. DGCluster learns directly from human-labeled communities, which often group disconnected nodes based solely on attributes, undermining connectivity cohesiveness. To overcome these challenges, we propose TAS-Com, powered by an innovative loss function that properly integrates topological and attribute information, improves the quality of human-labeled communities, and enables GCNs to discover community structures with optimal modularity and strong alignment with human labels.

3 Problem Definition

An *attributed network* can be modeled as a graph $N = (V, E, X)$, where V is the set of nodes, i.e., $V = \{v_1, v_2, ..., v_n\}$. E is the set of edges, i.e., $E = \{e_{i,j} | e_{i,j} \in V \times V\}$. $X \in \mathbb{R}^{n \times T}$ is the node attributes matrix, where T is the number of attributes for each node.

A *community structure* CS of an attributed network N is a set of non-overlapping communities, i.e., $CS = \{C_1, C_2, ..., C_k\}$ where $k \geq 1$ s.t. $\forall q \neq l$, $C_q \cap C_l = \emptyset$ and $\cup_{q=1}^{p} C_q = V$. The main goal of CD in attributed network N is to identify CS that satisfies the following conditions:

[1] The full paper is available at *https://arxiv.org/abs/2505.10197*, with code at *https://github.com/desilanja/TAS-Com*

- High topological cohesiveness: This implies that intra-community edge density within any community is higher than the inter-community edge density among different communities [Chunaev, 2020].
- High attribute similarity: Nodes within the same community exhibit strong attribute level similarity. Meanwhile, nodes from different communities are expected to have distinct and dissimilar attributes.

Modularity (Q) is a well-known metric to quantify the quality of different community structures [Newman and Girvan, 2004]. As a quality metric at the topology level, Q with respect to any given community structure CS is defined in Equation (1):

$$Q(CS) = \frac{1}{2m}\sum_{ij}(A_{ij} - \frac{k_i k_j}{2m})\delta(c_i, c_j, CS), \quad (1)$$

where m is the total number of edges of the social network, k_i and k_j are the degrees of the nodes v_i and v_j. A_{ij} is the adjacency matrix where $A_{ij} = 1$ if an edge exits between nodes v_i and v_j, otherwise 0. $\delta(.)$ is the Kronecker delta function where $\delta(c_i, c_j, CS) = 1$ if nodes v_i and v_j are in the same community (i.e., $c_i = c_j$ with c_i and c_j being the community labels of node i and j respectively), otherwise 0.

Normalized Mutual Information (NMI) is a widely used metric in existing research to assess how well a community structure C aligns with human-labeled community structure D in a social network N, particularly in terms of attributive similarity [Bhowmick *et al.*, 2024; Chunaev, 2020]. It is defined in Equation (2) below:

$$NMI(C, D) = \frac{-2\sum_{i=1}^{g_C}\sum_{j=1}^{g_D} P_{ij}\log(P_{ij}n/P_{i.}P_{.j})}{\sum_{i=1}^{g_C} P_{i.}\log(P_{i.}/n) + \sum_{j=1}^{g_D} P_{.j}\log(P_{.j}/n)}, \quad (2)$$

where P represents the confusion matrix. Each of its elements P_{ij} refers to the number of nodes of community $C_i \in C$ that are also in community $D_j \in D$. The values of i and j span within the range of $\{1, \ldots, n\}$ where n is the number of nodes in N. g_C refers to the number of communities in C. g_D refers to the number of communities in D. $P_{i.}$ denotes the sum of the i-th row of P and $P_{.j}$ denotes the sum of the j-th column.

Our Goal: We aim to identify high-quality community structures that can outperform those discovered by other state-of-the-art methods, achieving significantly higher scores in both modularity (indicating topological cohesiveness) and *NMI* (indicating alignment with human labels).

4 Proposed Method

This section proposes TAS-Com for CD in attributed networks. Figure 1 illustrates the overall design of TAS-Com. As shown in this figure, the adjacency matrix A, node attribute matrix X, and human-labeled communities CS_O are utilized to build the loss function L and to train our GCN model. Specifically, the GCN model first processes A and X to generate the high-level node embedding $X^{(e)}$. Supported by the Leiden algorithm and our refinement algorithm in Algorithm 1, $X^{(e)}$ is further adopted to calculate respectively the *modularity-based loss* L_M and *refined human-label-based loss* L_R to be introduced later in this section. The total loss L is then constructed by combining L_M and L_R. L subsequently guides the training of the GCN. Finally, based on $X^{(e)}$ produced by the trained GCN, a clustering algorithm named BIRCH [Zhang *et al.*, 1996] is performed to obtain the final community structure CS.

We follow [Bhowmick *et al.*, 2024] to design the architecture of our GCN model. In particular, the message passing rule for the l-th layer, where $l = 0, 1, ..., L-1$, is defined in Equation (3):

$$X^{(l)} = \sigma(\tilde{A}X^{(l-1)}W^{(l-1)}), \quad (3)$$

where $\tilde{A}$ is the normalized adjacency matrix s.t. $\tilde{A} = D^{-\frac{1}{2}}AD^{\frac{1}{2}}$. D is the diagonal node degree matrix [Bhowmick *et al.*, 2024]. The embedding output of the l-th layer is denoted by $X^{(l)}$. $W^{(l)}$ refers to the corresponding learnable weight matrix of this layer. SELU is used as the activation function $\sigma(\cdot)$ to incorporate the non-linearity for the aggregation of node attributes [Klambauer *et al.*, 2017]. Further, the node embedding matrix $X^{(e)}$ is transformed as in [Bhowmick *et al.*, 2024] to ensure that the embedding is constrained within the positive coordinate space (see Appendix C). The following provides detailed descriptions of the new components introduced in TAS-Com.

4.1 Proposed loss function (L)

In line with the problem formulation in the previous section, we propose a new loss function to train GCNs to jointly maximize the node connectivity strength (modularity optimization) and the node attribute similarity (*NMI* optimization). The proposed loss function is defined at a high level in Equation (4):

$$L = L_M + \mu L_R, \quad (4)$$

where $\mu \geq 0$ is a hyperparameter that controls the influence of L_R in L. L_M refers to the *modularity-based similarity loss* and L_R refers to the *refined human-label-based similarity loss*. The process of constructing L_M and L_R are explained in the next two subsections.

Modularity-based similarity loss (L_M)

Most existing GNNs were trained directly to optimize the modularity metric (see Equation (1)), resulting frequently in locally optimal community structures with low modularity [Tsitsulin *et al.*, 2023]. In this paper, we employ the Leiden algorithm with proven effectiveness and scalability [Traag *et al.*, 2019; de Silva *et al.*, 2023; de Silva *et al.*, 2022] to identify high-quality community structures with close-to-optimal modularity. Due to the stochastic nature of Leiden, we apply Leiden to a social network 30 times and select the community structure with the highest *NMI* score out of all runs. Leiden focuses solely on topological information, whereas our problem also concerns about attribute information. Therefore, we select the community structure with the highest NMI from 30 runs, ensuring an ideal balance between topological level and attribute level similarities.

Let $H \in \mathbb{R}^{n\times n}$ be the pairwise information matrix [Bhowmick *et al.*, 2024]. As an $n \times n$ matrix, where n is

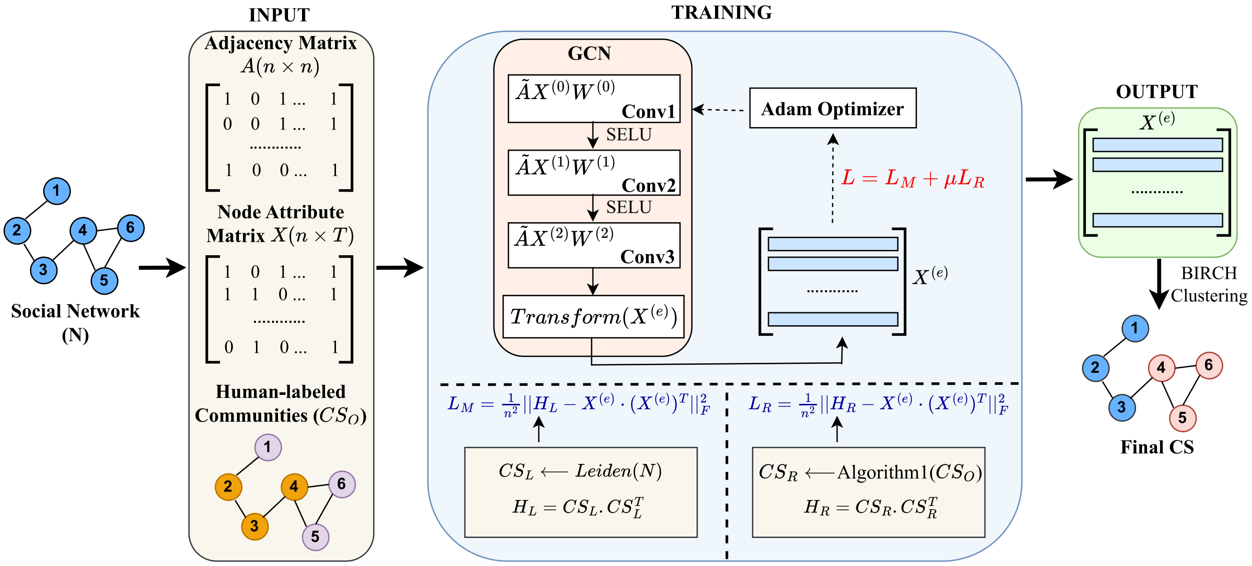
INPUT
Adjacency Matrix $A(n \times n)$
1 0 1 ... 1
0 0 1 ... 1
....
1 0 0 ... 1
Node Attribute Matrix $X(n \times T)$
1 0 1 ... 1
1 1 0 ... 1
....
0 1 0 ... 1
Human-labeled Communities (CS_O)
Social Network (N)
TRAINING
GCN
$\tilde{A}X^{(0)}W^{(0)}$ Conv1
SELU
$\tilde{A}X^{(1)}W^{(1)}$ Conv2
SELU
$\tilde{A}X^{(2)}W^{(2)}$ Conv3
$Transform(X^{(e)})$
Adam Optimizer
$L = L_M + \mu L_R$
$X^{(e)}$
$L_M = \frac{1}{n^2}||H_L - X^{(e)} \cdot (X^{(e)})^T||_F^2$
$L_R = \frac{1}{n^2}||H_R - X^{(e)} \cdot (X^{(e)})^T||_F^2$
$CS_L \longleftarrow Leiden(N)$
$H_L = CS_L.CS_L^T$
$CS_R \longleftarrow$ Algorithm1(CS_O)
$H_R = CS_R.CS_R^T$
OUTPUT
$X^{(e)}$
BIRCH Clustering
Final CS
1
2
3
4
5
6

Figure 1: The overall design of the proposed TAS-Com approach.

the total number of nodes in a social network, each element of H is defined as in Equation (5):

$$H_{ij} = \begin{cases} 1 & \text{if } c_i = c_j; \\ 0 & \text{otherwise}, \end{cases} \tag{5}$$

where c_i and c_j are the community assignments of nodes v_i and v_j according to Leiden. Specifically, the community structure identified by Leiden can be represented in the form of a one-hot matrix $CS_L \in \{0,1\}^{n \times k}$ below:

$$(CS_L)_{ij} = \begin{cases} 1 & \text{if node } v_i \text{ belongs to community } j; \\ 0 & \text{otherwise}, \end{cases}$$

where k is the total number of communities detected by Leiden. Subsequently, H can be rewritten as follows:

$$H_L = CS_L \cdot CS_L^T. \tag{6}$$

In line with the above, the modularity-based loss L_M is designed to minimize the discrepancy between the community assignments obtained by Leiden and the embedding similarity matrix derived from the high-level node embedding $X^{(e)}$ produced by the GCN model, as detailed below:

$$L_M = \frac{1}{n^2}||H_L - X^{(e)} \cdot (X^{(e)})^T||_F^2. \tag{7}$$

Refined human-label-based similarity loss (L_R)
As pointed out in the introduction, human-labeled communities contain disconnected nodes that seriously hurt the quality of the community structure in terms of modularity (see Appendix D for a concrete example). Hence, we propose a new method to refine human-labeled communities to simultaneously enhance consistency with human labels and connectivity within each community. In practice, if human-labeled communities are unavailable, any scalable CD algorithm can be used to obtain the substitute labels. The pseudo-code of the newly proposed refinement method is presented in Algorithm 1. According to this algorithm, the refinement process consists of two main steps explained below.

Algorithm 1 Refinement of the human-labeled communities

```
Input: Human-labeled community structure $CS_O$
Output: Refined community structure $CS_R$
for each community $C_k$ in $CS_O$ do
    Step 1: Apply Leiden algorithm
    Obtain a sub-network $SN$ that contains nodes in $C_k$
    $CS_L \longleftarrow CS$ with highest $Q(CS_L)$ after applying $Leiden(SN)$ for $n$ times
    Step 2: Merge sub-communities
    while $|CS_L| \geq threshold$ do
        for each pair of sub-communities $C_i$ and $C_j$ in $CS_L$ do
            $CS_{new} \longleftarrow$ Merge $C_i$ and $C_j$
            Calculate $Q(CS_{new})$ {Refer Equation (1)}
        end for
        $CS_L \longleftarrow CS_{new}$ with the highest $Q(CS_{new})$
    end while
    $CS_R \longleftarrow (CS_R \setminus \{C_k\}) \cup CS_L$
end for
return $CS_R$
```

1. **Apply a topology-based CD algorithm to detect sub-communities in each human-labeled community.** Since Leiden can produce topologically connected communities [Traag *et al.*, 2019], we apply Leiden to obtain the connected sub-communities within each human-labeled community. In practice, any scalable CD algorithm, such as Louvain [Blondel *et al.*, 2008], can replace Leiden for this task. Due to the stochastic nature

of Leiden, it is executed 10 times in our experiments for each human-labeled community. We then choose the community structure with the highest modularity, i.e., CS_L, as described in lines 2–4 of Algorithm 1.

2. **Merge sub-communities.** In our experiments (see Appendix E), we found that Leiden can sometimes produce many small sub-communities CS_L that significantly deviate from human-labeled communities. Hence, GCNs trained based on CS_L can exhibit poor NMI performance. To address this issue, we must merge closely connected sub-communities in CS_L. For this purpose, we iteratively merge pairs of sub-communities in CS_L. Each iteration will select and merge the pair of sub-communities that can produce the highest modularity after merging among all possible merge pairs (see lines 6–12 in Algorithm 1). This process continues until a specific $threshold = \frac{|CC(N_L)|}{2}$ is reached (see Appendix B), where $CC(N_L) = \{C_1, \ldots, C_k\}$ refers to the set of connected components of the sub-network N_L containing all nodes of network N in CS_L.

The community structure obtained from the above two steps is deemed the *refined human-label-based community structure*, denoted as CS_R. CS_R is subsequently adopted to train our GCN to maximize NMI. Notably, CS_R is consistent with human-labeled communities since each human-labeled community comprises one or more connected sub-communities in CS_R.

Using CS_R, we define the refined human-label-based loss L_R in Equation (8) below:

$$L_R = \frac{1}{n^2} ||H_R - X^{(e)} \cdot (X^{(e)})^T||_F^2. \tag{8}$$

In the above, matrix H_R follows the definition in Equation (5). It is the pairwise matrix obtained from below:

$$H_R = CS_R \cdot CS_R^T, \tag{9}$$

where CS_R is the one-hot matrix (similar to CS_L) derived from the community structure obtained from Algorithm 1. According to Equation (8), L_R guides our GCN to minimize the discrepancy between H_R and the embedding similarity matrix derived from $X^{(e)}$. This is expected to enable the trained GCN to produce node embedding that closely aligns with human-provided community labels.

Driven by the loss function L designed above, we can finally develop a system to train GCNs through the stochastic gradient descent method. The training process is summarized in Appendix A.

4.2 Community detection based on node embedding

Based on the node embedding produced by the trained GCN model, we utilize the clustering algorithm named Balance Iterative Reducing and Clustering using Hierarchies (BIRCH) [Zhang *et al.*, 1996] to identify the final community structure, following DGCluster [Bhowmick *et al.*, 2024]. The primary benefit of BIRCH is its ability to flexibly construct community structures without determining the number of communities in advance, enabling the algorithm to achieve a desirable trade-off between modularity and NMI.

5 Experiment Design

This section outlines the experimental settings used to examine the performance of TAS-Com.

5.1 Benchmark networks

We conduct experiments on six commonly used attributed social networks with human-provided community labels [Bhowmick *et al.*, 2024; Zhu *et al.*, 2024]. In particular, Cora and Citeseer are citation networks [Sen *et al.*, 2008], Amazon Photo and Amazon PC are co-purchase networks [Shchur *et al.*, 2018], and Coauthor CS [Shchur *et al.*, 2018] and Coauthor Phy [Shchur and Günnemann, 2019] are co-authorship networks for computer science and physics respectively. Table 1 summarizes all the benchmark networks.

Network	n	m	T	k
Cora	2708	5278	1433	7
Citeseer	3327	4552	3703	6
Amazon Photo	7650	119081	745	8
Amazon PC	13752	245861	767	10
Coauthor CS	18333	81894	6805	15
Coauthor Phy	34493	247962	8415	5

Table 1: Statistics of benchmark social networks. n, m, T, and k denote the number of nodes, edges, node attributes, and human-provided community labels respectively.

5.2 Baseline approaches

The effectiveness of TAS-Com is evaluated in comparison to 12 state-of-the-art approaches, categorized into three distinct groups: (1) approaches relying solely on attributive or topological information for CD, including k-m(feat) (i.e., k-means based only on features) and DMoN [Tsitsulin *et al.*, 2023]; (2) approaches that consider node similarities at both the topological and attribute levels, including k-m(DW) [Perozzi *et al.*, 2014], k-means(DGI) [Veličković *et al.*, 2018], DAEGC [Wang *et al.*, 2019], SDCN [Bo *et al.*, 2020], NOCD [Shchur and Günnemann, 2019], DyFSS [Zhu *et al.*, 2024], and DGCluster [Bhowmick *et al.*, 2024], and (3) approaches that use graph pooling techniques, including DiffPool [Ying *et al.*, 2018], MinCutPool [Bianchi *et al.*, 2020], and Ortho [Bianchi *et al.*, 2020]. Additional details regarding these approaches are provided in Appendix G, while the performance of the MinCutPool and Ortho approaches is analyzed in Appendix J. Among all competing approaches, DGCluster is the most competitive approach, as it is explicitly designed to maximize both modularity and attribute-level node similarity.

5.3 Performance metrics

We adopt two metrics, namely modularity and NMI introduced in Section 3 to compare the performance of all competing approaches. As explained previously, both metrics are essential to determine the quality of the community structures identified by the trained GCNs [Bhowmick *et al.*, 2024].

5.4 Parameter settings

In the experiments, TAS-Com employs a GCN architecture identical to that proposed in [Bhowmick *et al.*, 2024]. The model uses a GCN with two hidden layers, the Adam optimizer with a learning rate of 0.001, and is trained for 300 epochs. Additional details on the parameter settings of TAS-Com are provided in Appendix K. In line with existing works [Bhowmick *et al.*, 2024; Tsitsulin *et al.*, 2023], we report the average performance results across 10 independent runs with different random seeds. Furthermore, μ and $threshold$ in Algorithm 1 are two important hyperparameters. Their sensitivity analysis is reported in Appendix B.

6 Results and Discussion

6.1 Performance comparison

Table 2 compares the performance of TAS-Com with 12 state-of-the-art approaches in terms of modularity (Q) and NMI. Most of the results, except DyFSS and DGCluster have been reported previously in [Bhowmick *et al.*, 2024]. The results for DyFSS and DGCluster were reproduced using their published code and original experimental settings. Furthermore, the results of DGCluster are reported in Table 2 based on two settings of the hyperparameter λ (i.e., $\lambda = \{0.2, 0.8\}$), which are the recommended settings in [Bhowmick *et al.*, 2024]. As evidenced in Table 2, TAS-Com achieved the best trade-off between Q and NMI across all benchmark networks. We have conducted the Wilcoxin rank-sum test. Our test results show that the observed performance gain of TAS-Com is statistically significant (see Appendix H for statistical analysis).

Specifically, while TAS-Com obtained identical NMI value (i.e., 41.0) on the Citeseer network as DGCluster, TAS-Com significantly improves the quality of the community structure in terms of modularity Q. Although DyFSS attained the highest NMI value for the Citeseer network, its corresponding Q value was notably lower than that of TAS-Com. Meanwhile, concerning the Amazon PC and Coauthor Phy networks, TAS-Com noticeably improves the quality of the community structures in terms of NMI (i.e., 12% and 11% increase in NMI compared to DGCluster on the two networks respectively) without hurting the modularity Q. For the Coauthor CS network, while TAS-Com doesn't match the best Q value of 74.2 achieved by DGCluster($\lambda = 0.2$), it improves the Q value by 1.1% and matches the NMI value of 82.1 obtained by DGCluster($\lambda = 0.8$).

These results confirm that TAS-Com outperforms in Q and/or NMI without sacrificing either metric. It shows significant performance gains for several benchmark networks, reinforcing its competitiveness for CD in attributed networks.

6.2 Further analysis

We conduct additional analysis to validate the effectiveness of TAS-Com.

Connectivity within communities

We analyze the connectivity within each community identified by TAS-Com, DGCluster, and DyFSS. For this purpose, we check the average number of isolated sub-networks in each community of a community structure $CS = \{C_1, C_2, \dots, C_k\}$, i.e., $O_c(CS)$, as defined below:

$$O_c(CS) = \frac{\sum_{C \in CS} |CC(C)|}{|CS|}, \tag{10}$$

where $CC(C)$ refers to the set of isolated sub-networks within community $C \in CS$. Ideally, all nodes with any given community C are expected to be inter-connected, i.e., $|CC(C)| = 1$. In this case, $O_c(CS)$ reaches its smallest value of 1. Hence, the lower the value of $O_c(CS)$, the better. Figure 2 compares $O_c(CS)$ achieved respectively by TAS-Com, DGCluster, and DyFSS. For DGCluster, we report the lowest $O_c(CS)$ obtained by either DGCluster($\lambda = 0.2$) or DGCluster($\lambda = 0.8$).

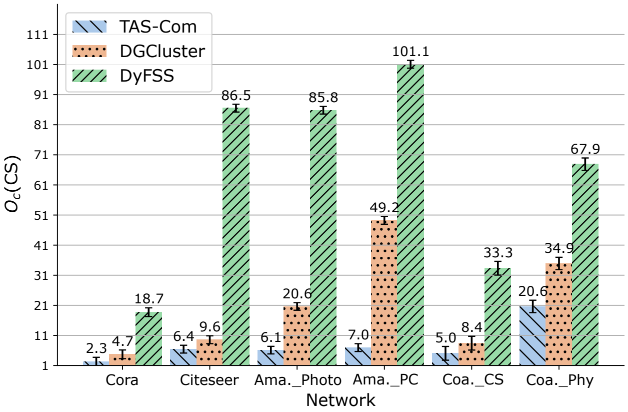

Figure 2: Comparison of $O_c(CS)$ achieved by TAS-Com, DGCluster, and DyFSS across all benchmark networks.

Figure 2 shows that TAS-Com significantly reduces isolated sub-networks compared to DGCluster and DyFSS. While TAS-Com may still produce isolated sub-networks, these can be treated as separate communities in the final CS without reducing modularity Q (see Appendix F). This confirms that using refined human-labeled communities to guide the training of GCNs enhances the effectiveness of CD.

Performance evaluation using additional metrics

To further evaluate the performance of the proposed TAS-Com, we employ two additional metrics: *conductance* and *F1 score* [Bhowmick *et al.*, 2024](see Appendix L). Table 3 compares TAS-Com and the baseline algorithm, DGCluster, in terms of average conductance (Con) and average F1 score ($F1$) on the benchmark networks Cora and Amazon Photo. The results demonstrate that TAS-Com outperforms DGCluster on these networks, highlighting its superior performance not only in Q and NMI but also in Con and $F1$. Additional results for all benchmark networks are in Appendix I.

Ablation study

To verify the effectiveness of the loss function proposed in Equation (4), an ablation study has been performed on the Cora network. We specifically consider four variants of TAS-Com associated with different designs of the loss functions. Figure 3 compares all variants based on both Q and NMI. In

Approach	Cora		Citeseer		Amazon Photo		Amazon PC		Coauthor CS		Coauthor Phy	
	Q	NMI	Q	NMI	Q	NMI	Q	NMI	Q	NMI	Q	NMI
k-m(feat)	19.8	18.5	30.3	24.5	10.5	28.8	5.4	21.1	23.1	35.7	19.4	30.6
k-m(DW)	30.7	24.3	24.3	27.6	22.9	49.4	11.8	38.2	59.4	72.7	47.0	43.5
SDCN	50.8	27.9	62.3	31.4	53.3	41.7	45.6	24.9	55.7	59.3	52.8	50.4
DAEGC	33.5	8.3	36.4	4.3	58.0	47.6	43.3	42.5	49.1	36.3	N/A	N/A
k-m(DGI)	64.0	52.7	73.7	40.4	35.1	33.4	22.8	22.6	57.8	64.6	51.2	51.0
NOCD	78.3	46.3	84.4	20.0	70.1	62.3	59.0	44.8	72.2	70.5	65.5	28.7
DiffPool	66.3	32.9	63.4	20.0	46.8	35.9	30.4	22.1	59.3	41.6	N/A	N/A
DMoN	76.5	48.8	79.3	33.7	70.1	63.3	59.0	49.3	72.4	69.1	65.8	56.7
DyFSS	73.5	55.5	75.2	44.8	57.7	53.6	35.1	36.3	68.9	76.8	65.6	56.9
DGCluster($\lambda = 0.2$)	80.8	53.0	87.4	30.3	71.6	73.0	61.5	53.8	74.2	76.1	67.3	59.0
DGCluster($\lambda = 0.8$)	78.6	62.1	86.3	41.0	71.6	77.3	60.3	60.4	73.3	82.1	66.0	65.7
TAS-Com	**81.7**	**65.1**	**88.1**	41.0	**72.2**	**78.4**	**61.5**	**61.2**	74.1	**82.1**	**67.3**	**66.0**

Table 2: Performance comparison in terms of Q and NMI (results are multiplied by 100) between TAS-Com and state-of-the-art approaches. The bolded results indicate instances where TAS-Com achieves the best results. N/A: Not Available.

Approach	Cora		Amazon Photo	
	Con	$F1$	Con	$F1$
DGCluster($\lambda = 0.2$)	9.7	43.5	8.6	70.7
DGCluster($\lambda = 0.8$)	14.5	54.5	12.4	75.9
TAS-Com	**8.8**	**56.9**	**8.1**	**76.7**

Table 3: Performance comparison of TAS-Com and DGCluster in terms of Con (the lower the better) and $F1$ (the higher the better). The best results (multiplied by 100) are bolded.

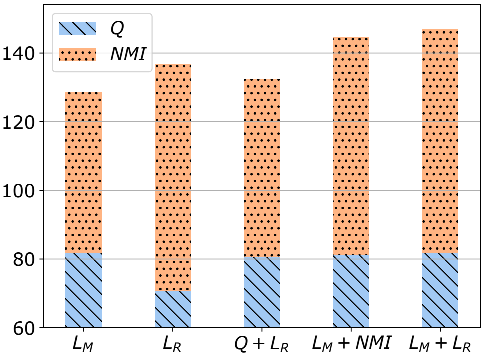

Figure 3: Ablation study on the Cora network evaluating the impact of L_M and L_R in the loss function L. Scores (Q and NMI) are scaled by 100. $L_M + L_R$ achieves the best total score.

this figure, L_M indicates that $L = L_M$, L_R indicates that $L = L_R$, $Q + L_R$ indicates that $L = Q + \mu L_R$, and $L_M + NMI$ indicates that $L = L_M + \mu NMI$, where NMI refers to the attributive similarity loss with respect to the human-labeled communities (without using Algorithm 1). Finally, $L_M + L_R$ indicates the proposed loss function in Equation (4).

Figure 3 shows that $L_M + L_R$ outperforms $Q + L_R$ and $L_M + NMI$, confirming the importance of both L_M and L_R for the overall loss L to be effective. While L_M alone yields slightly better Q value than $L_M + L_R$, it results in significantly lower NMI. This is because L_M focuses only on topological similarity. Similarly, L_R achieves a slightly higher NMI but a noticeably lower Q score compared to $L_M + L_R$, as it considers only attributive similarity. Hence, it is crucial to balance topological and attributive similarities by including both L_M and L_R in the overall loss L. This ablation study confirms the effectiveness of the proposed loss function L.

7 Conclusion

In this paper, we developed TAS-Com, a novel GCN-based approach powered by a newly designed loss function to effectively train GCNs to extract high-level node embeddings in attributed social networks. Our new loss function leverages the highly effective and scalable Leiden algorithm to identify community structures with globally optimal modularity (Q). Meanwhile, Leiden refines human-labeled communities, ensuring connectivity within each community and enabling GCNs to effectively learn node similarity at both the topology and attribute levels. Thanks to our new loss function design, the trained GCNs can detect community structures with a desirable balance between topological similarity (i.e., modularity (Q)) and attributive similarity (i.e., NMI). Experiments on multiple benchmark networks with varying sizes and complexities show that TAS-Com significantly outperforms 12 state-of-the-art approaches.

Future work could explore the potential integration of temporal dynamics or multi-view data to enhance the wide applicability of TAS-Com, including dynamic social networks.

References

[Bhowmick *et al.*, 2024] Aritra Bhowmick, Mert Kosan, Zexi Huang, Ambuj Singh, and Sourav Medya. Dgcluster: A neural framework for attributed graph clustering via modularity maximization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11069–11077, 2024.

[Bianchi *et al.*, 2020] Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. Spectral clustering with graph neural networks for graph pooling. In *International conference on machine learning*, pages 874–883. PMLR, 2020.

[Blondel *et al.*, 2008] 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), 2008.

[Bo *et al.*, 2020] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Structural deep clustering network. In *Proceedings of the web conference 2020*, pages 1400–1410, 2020.

[Chunaev, 2020] Petr Chunaev. Community detection in node-attributed social networks: a survey. *Computer Science Review*, 37:100286, 2020.

[Combe *et al.*, 2015] David Combe, Christine Largeron, Mathias Géry, and Előd Egyed-Zsigmond. I-louvain: An attributed graph clustering method. In *Advances in Intelligent Data Analysis XIV: 14th International Symposium. Proceedings 14*, pages 181–192. Springer, 2015.

[de Silva *et al.*, 2022] Anjali de Silva, Aaron Chen, Hui Ma, and Mohammad Nekooei. Genetic algorithm with a novel leiden-based mutation operator for community detection. In *Advances in Artificial Intelligence: 35th Australasian Joint Conference, Proceedings*, pages 252–265. Springer, 2022.

[de Silva *et al.*, 2023] Anjali de Silva, Gang Chen, Hui Ma, and Seyed Mohammad Nekooei. Leiden fitness-based genetic algorithm with niching for community detection in large social networks. In *Pacific Rim International Conference on Artificial Intelligence*, pages 423–435. Springer, 2023.

[Guo *et al.*, 2024] Kun Guo, Zhanhong Chen, Zhiyong Yu, Kai Chen, and Wenzhong Guo. Evolutionary computing empowered community detection in attributed networks. *IEEE Communications Magazine*, 62(5):22–26, 2024.

[He *et al.*, 2021a] Dongxiao He, Shuai Li, Di Jin 0001, Pengfei Jiao, and Yuxiao Huang. Self-guided community detection on networks with missing edges. In *IJCAI*, pages 3508–3514, 2021.

[He *et al.*, 2021b] Dongxiao He, Yue Song, Di Jin, Zhiyong Feng, Binbin Zhang, Zhizhi Yu, and Weixiong Zhang. Community-centric graph convolutional network for unsupervised community detection. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, pages 3515–3521, 2021.

[He *et al.*, 2022] Chaobo He, Yulong Zheng, Junwei Cheng, Yong Tang, Guohua Chen, and Hai Liu. Semi-supervised overlapping community detection in attributed graph with graph convolutional autoencoder. *Information Sciences*, 608:1464–1479, 2022.

[Jin *et al.*, 2019] Di Jin, Ziyang Liu, Weihao Li, Dongxiao He, and Weixiong Zhang. Graph convolutional networks meet markov random fields: Semi-supervised community detection in attribute networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 152–159, 2019.

[Jin *et al.*, 2021] Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, S Yu Philip, and Weixiong Zhang. A survey of community detection approaches: From statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1149–1170, 2021.

[Ju *et al.*, 2023] Wei Ju, Yiyang Gu, Binqi Chen, Gongbo Sun, Yifang Qin, Xingyuming Liu, Xiao Luo, and Ming Zhang. Glcc: A general framework for graph-level clustering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 4391–4399, 2023.

[Karataş and Şahin, 2018] Arzum Karataş and Serap Şahin. Application areas of community detection: A review. In *International congress on big data, deep learning and fighting cyber terrorism*, pages 65–70. IEEE, 2018.

[Klambauer *et al.*, 2017] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017.

[Kumar *et al.*, 2023] Sanjay Kumar, Abhishek Mallik, and Sandeep Singh Sengar. Community detection in complex networks using stacked autoencoders and crow search algorithm. *The Journal of Supercomputing*, 79(3):3329–3356, 2023.

[Li *et al.*, 2017] Zhangtao Li, Jing Liu, and Kai 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, 2017.

[Liu *et al.*, 2022] Yue Liu, Wenxuan Tu, Sihang Zhou, Xinwang Liu, Linxuan Song, Xihong Yang, and En Zhu. Deep graph clustering via dual correlation reduction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 7603–7611, 2022.

[Liu *et al.*, 2024] Hongtao Liu, Jiahao Wei, Yiming Wu, and Cong Liang. Information-enhanced deep graph clustering network. *Neurocomputing*, page 127992, 2024.

[Luo and Yan, 2020] Mengqing Luo and Hui Yan. Adaptive attributed network embedding for community detection. In *Chinese Conference on Pattern Recognition and Computer Vision*, pages 161–172. Springer, 2020.

[Moradi *et al.*, 2015] Parham Moradi, Sajad Ahmadian, and Fardin Akhlaghian. An effective trust-based recommendation method using a novel graph clustering algorithm.

Physica A: Statistical mechanics and its applications, 436:462–481, 2015.

[Newman and Girvan, 2004] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):1–15, 2004.

[Perozzi *et al.*, 2014] 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*, pages 701–710, 2014.

[Pizzuti and Socievole, 2018] Clara Pizzuti and Annalisa Socievole. A genetic algorithm for community detection in attributed graphs. In *Applications of Evolutionary Computation: 21st International Conference, Proceedings 21*, pages 159–170. Springer, 2018.

[Sen *et al.*, 2008] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[Shchur and Günnemann, 2019] Oleksandr Shchur and Stephan Günnemann. Overlapping community detection with graph neural networks. *arXiv preprint arXiv:1909.12201*, 2019.

[Shchur *et al.*, 2018] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. arxiv 2018. *arXiv preprint arXiv:1811.05868*, 2018.

[Su *et al.*, 2022] Xing Su, Shan Xue, Fanzhen Liu, Jia Wu, Jian Yang, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Di Jin, et al. A comprehensive survey on community detection with deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21, 2022.

[Sun *et al.*, 2019] Fan-Yun Sun, Meng Qu, Jordan Hoffmann, Chin-Wei Huang, and Jian Tang. vgraph: A generative model for joint community detection and node representation learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[Sun *et al.*, 2020] Heli Sun, Fang He, Jianbin Huang, Yizhou Sun, Yang Li, Chenyu Wang, Liang He, Zhongbin Sun, and Xiaolin Jia. Network embedding for community detection in attributed networks. *ACM Transactions on Knowledge Discovery from Data*, 14(3):1–25, 2020.

[Traag *et al.*, 2019] Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1):1–12, 2019.

[Tsitsulin *et al.*, 2023] Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. Graph clustering with graph neural networks. *Journal of Machine Learning Research*, 24(127):1–21, 2023.

[Veličković *et al.*, 2018] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

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

[Wang *et al.*, 2021] Xiaofeng Wang, Jianhua Li, Li Yang, and Hongmei Mi. Unsupervised learning for community detection in attributed networks based on graph convolutional network. *Neurocomputing*, 456:147–155, 2021.

[Wu *et al.*, 2020] Ling Wu, Qishan Zhang, Chi-Hua Chen, Kun Guo, and Deqin Wang. Deep learning techniques for community detection in social networks. *IEEE Access*, 8:96016–96026, 2020.

[Xia *et al.*, 2021] Wei Xia, Quanxue Gao, Ming Yang, and Xinbo Gao. Self-supervised contrastive attributed graph clustering. *arXiv preprint arXiv:2110.08264*, 2021.

[Xie *et al.*, 2021] Xiaoqin Xie, Mingjie Song, Chiming Liu, Jiaming Zhang, and Jiahui Li. Effective influential community search on attributed graph. *Neurocomputing*, 444:111–125, 2021.

[Yang *et al.*, 2016] Liang Yang, Xiaochun Cao, Dongxiao He, Chuan Wang, Xiao Wang, and Weixiong Zhang. Modularity based community detection with deep learning. In *IJCAI*, volume 16, pages 2252–2258, 2016.

[Yang *et al.*, 2023] Xihong Yang, Yue Liu, Sihang Zhou, Siwei Wang, Wenxuan Tu, Qun Zheng, Xinwang Liu, Liming Fang, and En Zhu. Cluster-guided contrastive graph clustering network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 10834–10842, 2023.

[Ying *et al.*, 2018] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 974–983, 2018.

[Zhang *et al.*, 1996] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. *ACM sigmod record*, 25(2):103–114, 1996.

[Zhou *et al.*, 2023] Xinchuang Zhou, Lingtao Su, Xiangju Li, Zhongying Zhao, and Chao Li. Community detection based on unsupervised attributed network embedding. *Expert Systems with Applications*, 213:118937, 2023.

[Zhu *et al.*, 2024] Pengfei Zhu, Qian Wang, Yu Wang, Jialu Li, and Qinghua Hu. Every node is different: Dynamically fusing self-supervised tasks for attributed graph clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17184–17192, 2024.