



A Deep Conditional Generative Approach for Constrained Community Detection

Chaobo He

School of Computer Science, South
China Normal University
Pazhou Lab
Guangzhou, China
hechaobo@m.scnu.edu.cn

Junwei Cheng

School of Computer Science, South
China Normal University
Guangzhou, China
jung@m.scnu.edu.cn

Quanlong Guan

College of Information Science and
Technology, Jinan University
Guangzhou, China
gql@jnu.edu.cn

Xiang Fei

Department of Computing, Coventry
University
Coventry, United Kingdom
aa5861@coventry.ac.uk

Hanchao Li

Institute for Digital Technologies,
Loughborough University
Leicester, United Kingdom
h.li6-22@student.lboro.ac.uk

Yong Tang*

School of Computer Science, South
China Normal University
Guangzhou, China
ytang@m.scnu.edu.cn

ABSTRACT

Constrained community detection is one of the popular topics in graph data mining, and it aims to improve the performance by exploiting prior pairwise constraints, such as must-link and cannot-link constraints. However, most of existing methods for constrained community detection are shallow approaches, and are also not robust to handle constraints information with noises. In view of this, we propose a deep conditional generative approach CGMVGA. It firstly treats pairwise constraints as the priors with different degrees of certainty, and then integrates them into the conditional Gaussian mixture model. By further combining variational graph auto-encoders and the Wasserstein regularization, CGMVGA can learn the latent node representations preserving community structures in a deep generative manner. Experimental results show that CGMVGA outperforms state-of-the-art approaches, and is also more robust.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks; Learning latent representations.**

KEYWORDS

constrained community detection, conditional gaussian mixture, variational graph auto-encoders, generative approach

ACM Reference Format:

Chaobo He, Junwei Cheng, Quanlong Guan, Xiang Fei, Hanchao Li, and Yong Tang. 2023. A Deep Conditional Generative Approach for Constrained Community Detection. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23)*, October 21–25, 2023, Birmingham, United Kingdom.

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CIKM '23, October 21–25, 2023, Birmingham, United Kingdom.

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0124-5/23/10...\$15.00
<https://doi.org/10.1145/3583780.3615145>

2023, Birmingham, United Kingdom. ACM, New York, NY, USA, 5 pages.
<https://doi.org/10.1145/3583780.3615145>

1 INTRODUCTION

Community detection (a.k.a. graph clustering) is a fundamental task in the field of graph data mining. It aims to find node clusters satisfying: nodes in the same cluster should link to each other more densely than those in different clusters [1]. Recently, community detection has received lots of attention from the industry and academia due to its significant research value (e.g., enhancing the understanding towards functional modules in protein-protein interaction networks [2] and identifying user groups with similar interests in online social networks [3]), and various methods have been continuously proposed, including nonnegative matrix factorization (NMF) based methods [4], label propagation based methods [5], stochastic block model based methods [6] and deep learning based methods [7, 8].

In general, most of existing approaches treat community detection as a very pure unsupervised learning task (i.e., clustering) without considering any prior knowledge. It is a costly and time-consuming task to label nodes, but some must-link or cannot-link pairwise constraints, which respectively denotes that two given nodes should belong to the same community or not, are often easily to be acquired in real-world graph data. For example, in the political network of the United States, it is well known that Biden and Blinken are in the same party (must-link), but they and Trump are in different parties (cannot-link). In the last decades, some researchers have begun to exploit these pairwise constraints to supervise the process of community detection, which is referred as **constrained community detection**. For example, Eaton et al. [9] proposed a symmetric nonnegative matrix tri-factorization model, which uses a graph regularization term to incorporate must-link and cannot-link constraints for detecting communities. Similar idea can be found in some later variants, such as PCSNMF [10], PSSNMTF [11] and DNR [12]. To utilize the pairwise constraints more effectively, Ganj et al. [13] designed a discrete Lagrange multiplier based model LagCCD, which is combined with the modularity maximization objective for community extraction. From the perspective of optimizing graph structure with pairwise constraints, Nie et al. [14]

designed a semi-supervised learning model PCOG with the Laplacian rank regularization, and can identify community structure directly through a particular membership matrix.

Through analyzing relevant literature, we find that the problem of constrained community detection has not been extensively and thoroughly studied yet. Existing methods all demonstrate the certain ability of integrating pairwise constraints to improve the performance of community detection. However, most of them adopt shallow models, such as the most frequently used graph regularized NMF in [9–12], which are hard to learn complicated and diversified graph structure features under pairwise constraints, resulting in the unremarkable performance. Moreover, due to the possible limitation of expert knowledge or the quality of information source, the prior pairwise constraints may incorporate noises, but existing methods are often not robust enough and their performance is prone to degrade when the noise level increases.

Aiming at the aforementioned problems, inspired by the success of Variational Graph Auto-Encoders (VGAE) [15] and conditional Gaussian mixture [16], in this paper we propose a deep conditional generative model using Conditional Gaussian Mixture VGAE (CGMVGAE), and our main contributions are summarized as follows:

- CGMVGAE seamlessly combines VGAE with conditional Gaussian mixture, which treats the pairwise constraints with varying degrees of certainty as the Bayesian priors. By means of the inherent generative mechanism and clustering ability, CGMVGAE is able to learn the posterior distributions of the latent node representations, which also can well preserve community structures.
- To uncover more accurate community structures, CGMVGAE introduces 2-Wasserstein distance to further regularize the posterior distributions of node representations with pairwise constraints. Following the idea of variational inference, CGMVGAE is trained effectively via the unified objective composed of conditional evidence lower bound (CELBO) and the Wasserstein regularization loss.
- Extensive experiments are conducted on three widely used datasets. The results demonstrate that CGMVGAE is very competitive and superior to several state-of-the-art representative methods.

2 METHODOLOGY

2.1 Preliminaries and problem statement

Definition 1. (Graph): A given graph is represented as $G = \{V, E, X\}$, where $V = \{v_1, v_2, \dots, v_N\}$ is the set of N nodes, $E = \{e_{ij} | v_i \in V \wedge v_j \in V\}$ is a set of edges between nodes and $X \in \mathbb{R}^{N \times d}$ is the feature matrix composed of d -dimensional node feature vectors. The topological structure of G is denoted as the adjacency matrix $A = [A_{ij}]^{N \times N} \in \mathbb{R}^{N \times N}$. Without loss of generality, in this paper we focus on the undirected and unweighted graphs, such that A is symmetric, and $A_{ij} = 1$ if $e_{ij} \in E$, otherwise $A_{ij} = 0$.

Definition 2. (Constraints information): We consider two types of common pairwise constraints: must-link and cannot-link. A must-link constraint indicates that two given nodes are believed to belong to the same community, and a cannot-link constraint has the completely opposite meaning. As mentioned in Section 1, these known

pairwise constraints may have noises (i.e., are not necessarily completely reliable). Therefore, to represent them accurately, we define the following constraints information matrix $W = [W_{ij}]^{N \times N}$:

$$W_{ij} \begin{cases} > 0 & \text{if } v_i \text{ and } v_j \text{ have a must-link constraint} \\ = 0 & \text{if } v_i \text{ and } v_j \text{ have no pairwise constraints} \\ < 0 & \text{if } v_i \text{ and } v_j \text{ have a cannot-link constraint,} \end{cases} \quad (1)$$

where $|W_{ij}| \in [0, 1]$ reflects the degree of certainty of the corresponding pairwise constraint.

Problem statement. (Constrained community detection): Given the graph G and the constraints information matrix W , constrained community detection aims to partition nodes in G into the preassigned K disjoint clusters under the supervision of W .

2.2 CGMVGAE

1) Overview. We design CGMVGAE by combining VGAE and Conditional Gaussian Mixture. VGAE is used to learn the latent node representations using deep generative framework based on Variational Auto-Encoder (VAE) [17] and Graph Convolutional Networks (GCN) [18]. By treating conditional Gaussian mixture as the generative model (Fig. 1) conditioned on W and meanwhile introducing the Wasserstein regularization, the latent node representations are expected to reflect community structures well. The key components of CGMVGAE are detailed as follows.

2) Generative assumption. As depicted in Fig. 1, we let a_i (the i -th row vector of A) denote the node sample v_i itself, and assume it is generated via the following random process:

- **Step 1:** For the latent community assignments $c = \{c_i\}_{i=1}^N$ with $c_i \in \{1, \dots, K\}$ denoting the community index of v_i , we sample c_i from a distribution conditioned on W : $c_i \sim p(c_i | W)$ with the parameter $\pi = [\pi_1, \pi_2, \dots, \pi_K] \in \mathbb{R}_+^K$, where π_i is the prior probability for the i -th community and $\sum_{k=1}^K \pi_k = 1$. Here, $p(c_i | W)$ is treated as the conditional prior probability defined as:

$$p(c_i | W) = \frac{\pi_{c_i} \prod_{j \neq i} \exp(W_{ij} \delta(c_i, c_j))}{\sum_{k=1}^K \pi_k \prod_{j \neq i} \exp(W_{ij} \delta(c_i = k, c_j))}, \quad (2)$$

where δ is the Kronecker function that $\delta(x, y) = 1$ if $x = y$, and $\delta(x, y) = 0$ otherwise. Obviously, $p(c_i | W)$ will have large values if c_i conforms to the prior constraints and low values otherwise.

- **Step 2:** For node representations set $H = \{h_i\}_{i=1}^N$, h_i is sampled from a distribution conditioned on c_i : $p(h_i | c_i) = \mathcal{N}(\mu_{c_i}, \sigma_{c_i}^2 I)$, where μ_{c_i} and $\sigma_{c_i}^2$ are the mean and variance of the multivariate Gaussian distribution $\mathcal{N}(\cdot)$ corresponding to the c_i -th community, and I is an identity matrix.
- **Step 3:** For the binary-valued sample a_i , it can be generated from a multivariate Bernoulli distribution: $a_i \sim \text{Bern}(\mu_{a_i})$, where μ_{a_i} is the mean computed by a non-linear function $f(h_i; \theta)$, whose input is h_i and is parameterized by θ .

Given the generative assumption above, the objective of constrained community detection here can be transformed to infer parameters θ , $\{\mu_{c_k}, \sigma_{c_k}^2\}_{k=1}^K$ and π by maximizing the following marginal log-likelihood conditional on W :

$$\log p(A | W) = \mathbb{E}_{p(H, c | A)} \log \frac{p(A, H, c | W)}{p(H, c | A, W)}. \quad (3)$$

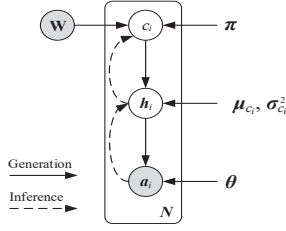


Figure 1: The generation and inference processes of CGMV-GAE. The constraints information matrix \mathbf{W} and the topological feature vector \mathbf{a}_i of a given node v_i are both treated as the observed variables, and the latent variables c_i and h_i respectively denote the community assignment and the latent representation of v_i .

3) Variational inference. These true posterior distributions in Eq. (3) are mostly intractable. Accordingly, we apply the variational inference framework widely used in Bayesian generative models, such as VAE [17] and VaDE [19]. Specifically, the mean-field variational distribution $q(\mathbf{H}, \mathbf{c}|\mathbf{A})$ is used to approximate $p(\mathbf{H}, \mathbf{c}|\mathbf{A})$, resulting in:

$$\log p(\mathbf{A}|\mathbf{W}) = \mathbb{E}_{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} \log \frac{p(\mathbf{A}, \mathbf{H}, \mathbf{c}|\mathbf{W})}{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} + \mathbb{E}_{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} \log \frac{q(\mathbf{H}, \mathbf{c}|\mathbf{A})}{p(\mathbf{H}, \mathbf{c}|\mathbf{A}, \mathbf{W})},$$

then we treat the first term as the conditional evidence lower bound $\mathcal{L}_{\text{CELBO}}$, and finally maximizing $\log p(\mathbf{A}|\mathbf{W})$ can be optimized by maximizing the conditional evidence lower bound $\mathcal{L}_{\text{CELBO}}$, that is:

$$\log p(\mathbf{A}|\mathbf{W}) \geq \mathcal{L}_{\text{CELBO}} = \mathbb{E}_{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} \log \frac{p(\mathbf{A}, \mathbf{H}, \mathbf{c}|\mathbf{W})}{q(\mathbf{H}, \mathbf{c}|\mathbf{A})}. \quad (5)$$

In Eq. (5), for the upper part, due to that \mathbf{a}_i and c_i are independently conditioned on \mathbf{h}_i , so the conditional joint probability $p(\mathbf{A}, \mathbf{H}, \mathbf{c}|\mathbf{W})$ can be factorized as:

$$p(\mathbf{A}, \mathbf{H}, \mathbf{c}|\mathbf{W}) = p(\mathbf{A}|\mathbf{H})p(\mathbf{H}, \mathbf{c}|\mathbf{W}) = p(\mathbf{A}|\mathbf{H})p(\mathbf{H}|\mathbf{c})p(\mathbf{c}|\mathbf{W}), \quad (6)$$

By substituting terms in Eq. (6) into Eq. (5), $\mathcal{L}_{\text{CELBO}}$ can be written as:

$$\begin{aligned} \mathcal{L}_{\text{CELBO}} &= \mathbb{E}_{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} [\log p(\mathbf{A}, \mathbf{H}, \mathbf{c}|\mathbf{W}) - \log q(\mathbf{H}, \mathbf{c}|\mathbf{A})] \\ &= \mathbb{E}_{q(\mathbf{H}, \mathbf{c}|\mathbf{A})} [\log p(\mathbf{A}|\mathbf{H})] - \text{KL}[q(\mathbf{H}, \mathbf{c}|\mathbf{A})||p(\mathbf{H}, \mathbf{c}|\mathbf{W})], \end{aligned} \quad (7)$$

where $\text{KL}[q(\cdot)||p(\cdot)]$ defines the Kullback-Leibler (KL) divergence between $q(\cdot)$ and $p(\cdot)$. We respectively treat the first part and the second part with the minus sign as the reconstruction loss \mathcal{L}_{rec} and the KL-divergence loss \mathcal{L}_{kld} , and hence $\mathcal{L}_{\text{CELBO}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{kld}}$. Next, we will introduce how to compute \mathcal{L}_{rec} and \mathcal{L}_{kld} .

Under the mean-field assumption, $q(\mathbf{H}, \mathbf{c}|\mathbf{A})$ can be factorized as: $q(\mathbf{H}, \mathbf{c}|\mathbf{A}) = q(\mathbf{H}|\mathbf{A})q(\mathbf{c}|\mathbf{A}) = \prod_{i=1}^N q(\mathbf{h}_i|\mathbf{A}) \prod_{i=1}^N q(c_i|\mathbf{A})$. Following the idea presented in VGAE [16], we model $q(\mathbf{H}|\mathbf{A})$ as follows:

$$q(\mathbf{h}_i|\mathbf{A}) = \mathcal{N}(\mathbf{h}_i|\tilde{\boldsymbol{\mu}}_i, \text{diag}(\tilde{\boldsymbol{\sigma}}_i^2)), \quad (8)$$

where $\tilde{\boldsymbol{\mu}}$ and $\tilde{\boldsymbol{\sigma}}$ denote the matrices of mean and standard deviation for nodes, and are respectively obtained by a two-layer GCN: $\tilde{\boldsymbol{\mu}} = \text{GCN}_{\boldsymbol{\mu}}(\mathbf{A}, \mathbf{X})$ and $\log \tilde{\boldsymbol{\sigma}} = \text{GCN}_{\boldsymbol{\sigma}}(\mathbf{A}, \mathbf{X})$. The two-layer GCN is defined as $\text{GCN}(\mathbf{A}, \mathbf{X}) = \tilde{\text{ReLU}}(\tilde{\mathbf{A}}\mathbf{X}\mathbf{W}_0)\mathbf{W}_1$, where \mathbf{W}_0 and \mathbf{W}_1 denote the weight matrices of the first layer and the second layer respectively, \mathbf{W}_0 is shared between $\text{GCN}_{\boldsymbol{\sigma}}(\mathbf{A}, \mathbf{X})$ and $\text{GCN}_{\boldsymbol{\mu}}(\mathbf{A}, \mathbf{X})$, the activation function $\text{ReLU}(\cdot) = \max(0, \cdot)$ and $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\mathbf{D}^{-\frac{1}{2}}$ with \mathbf{D} denoting the degree matrix of $\mathbf{A} + \mathbf{I}$.

As mentioned at Step 3 of the generation process, $\mathbf{a}_i \sim \text{Bern}(\boldsymbol{\mu}_{a_i})$, so \mathcal{L}_{rec} can be computed by the binary cross entropy:

$$\mathcal{L}_{\text{rec}} = \frac{1}{L} \sum_{l=1}^L \sum_{i=1}^N \sum_{j=1}^N \mathbf{A}_{ij} \log \hat{\mathbf{A}}_{ij}^{(l)} + (1 - \mathbf{A}_{ij}) \log (1 - \hat{\mathbf{A}}_{ij}^{(l)}), \quad (9)$$

where L is the number of samples from Monte Carlo Stochastic Gradient Variational Bayes (SGVB) estimator [15], and $\hat{\mathbf{A}}^{(l)}$ is computed via the sigmoid dot product function: $\hat{\mathbf{A}}_{ij}^{(l)} = \text{sigmoid}(\mathbf{h}_i^{(l)} (\mathbf{h}_j^{(l)})^T)$, where $\mathbf{h}_i^{(l)}$ is the l -th sample from $q(\mathbf{h}_i|\mathbf{A})$ in Eq. (8). To allow the gradient backpropagation, the reparameterization trick is used to obtain $\mathbf{h}_i^{(l)}$: $\mathbf{h}_i^{(l)} = \tilde{\boldsymbol{\mu}}_i + \tilde{\boldsymbol{\sigma}}_i \circ \epsilon^{(l)}$, where \circ is the element-wise operator and $\epsilon^{(l)} \sim \mathcal{N}(0, \mathbf{I})$.

For \mathcal{L}_{kld} , based on the independent assumption mentioned above, it can be written as: $\mathcal{L}_{\text{kld}} = \mathbb{E}_{q(\mathbf{H}|\mathbf{A})q(\mathbf{c}|\mathbf{A})} [\log p(\mathbf{H}|\mathbf{c}) + \log p(\mathbf{c}|\mathbf{W}) - \log q(\mathbf{H}|\mathbf{A}) - \log q(\mathbf{c}|\mathbf{A})]$.

4) Wasserstein regularization. The constraints information matrix \mathbf{W} not only provides prior knowledge on community partitions, but also can be used to further constrain the node representations in the latent space. Namely, if \mathbf{W}_{ij} is large, v_i and v_j should have very similar node representations. To characterize this similarity relationship, we introduce the Wasserstein regularization term \mathcal{L}_{reg} , which uses 2-Wasserstein distance (d_{Was}) to compute the similarity between nodes with constraints information, and is defined as: $\mathcal{L}_{\text{reg}} = \sum_{i,j} \mathbf{W}_{ij} d_{\text{Was}}(\mathbf{h}_i||\mathbf{h}_j) = \sum_{i,j} \mathbf{W}_{ij} \|\tilde{\boldsymbol{\mu}}_i - \tilde{\boldsymbol{\mu}}_j\|_2^2 + \|\tilde{\boldsymbol{\sigma}}_i - \tilde{\boldsymbol{\sigma}}_j\|_2^2$.

5) The unified objective. To learn more accurate node representations that are beneficial to the constrained community detection task, we combine $\mathcal{L}_{\text{CELBO}}$ and \mathcal{L}_{reg} as the final objective function: $\mathcal{L} = \mathcal{L}_{\text{CELBO}} - \alpha \mathcal{L}_{\text{reg}}$, where $\alpha \geq 0$ is used to control the contribution of the Wasserstein regularization term. All models in CGMV-GAE, i.e., Gaussian mixtures and GCNs, can be trained together by maximizing \mathcal{L} .

6) Inferring community assignments. During the training, we could infer and update the community assignment of v_i via $p(c_i|\mathbf{h}_i)$, which can be computed using the Bayes theorem:

$$p(c_i|\mathbf{h}_i) = \frac{p(c_i)p(\mathbf{h}_i|c_i)}{\sum_{k=1}^K p(c_k)p(\mathbf{h}_i|c_k)} = \frac{\mathcal{N}(\mathbf{h}_i|\boldsymbol{\mu}_{c_i}, \boldsymbol{\sigma}_{c_i}^2)\pi_{c_i}}{\sum_{k=1}^K \mathcal{N}(\mathbf{h}_i|\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k^2)\pi_k}. \quad (10)$$

3 EXPERIMENTS

3.1 Experiment setup

1) Datasets. We select three widely used benchmark graphs with ground-truth community labels as datasets, including Cora, Citeseer and PubMed [20]. Every node in these graphs is associated with a 0/1-valued attribute vector, which can be treated as the corresponding feature vector. The detailed statistics of the selected datasets are given in Table 1. Note that the ground-truth community labels can provide reliable sources for setting the prior pairwise constraints. Assuming that the set of nodes with label i is denoted as C_i , the possible numbers of must-link and cannot-link pairwise constraints are respectively: $\sum_{i=1}^K \frac{|C_i|(|C_i|-1)}{2}$ and $\sum_{i=1}^K \sum_{j=i+1}^K |C_i||C_j|$. In our experiments, we randomly extract these two types of pairwise constraints as the priors according to the same preassigned ratio.

2) Baselines and evaluation metrics. We select DNR [12], LagCCD [13] and PCOG [14] as baselines. All these methods are able to simultaneously incorporate must-link and cannot-link pairwise

Table 1: Statistics of datasets.

Datasets	N	$ E $	d	K
Cora	2,708	5,429	1,433	7
Citeseer	3,312	4,732	3,703	6
PubMed	19,729	44,338	500	3

constraints. Two widely used community detection metrics are adopted: Normalized Mutual Information (NMI) [21] and Adjusted Rand Index (ARI) [21]. The value ranges of these metrics are $[0,1]$, and larger NMI and ARI scores indicate better performance.

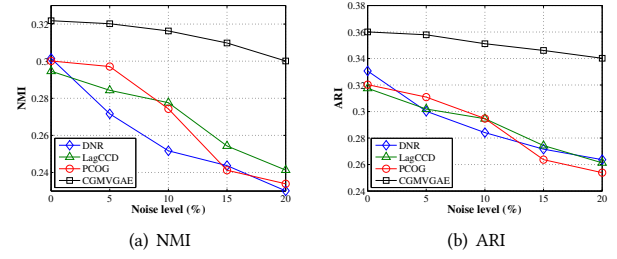
3.2 Results and analysis

1) Performance comparison. On every dataset, we compare with baselines under the same ratio of pairwise constraints (varying from 5% to 25%), and the results are shown in Table 2. As we can see, the performances of all baselines can be continually improved with the increase of the pairwise constraints ratio, but CGMVGAEE always performs the best, especially on the largest dataset: PubMed. For example, when the ratio of pairwise constraints on PubMed is 25%, the NMI and ARI scores of CGMVGAEE respectively improve by 6.7% and 8.9%, compared to the second best method DNR. The reasons for this may be twofold: On one hand, CGMVGAEE is a deep learning method based on VGAE and most baselines (except for DNR) are shallow methods, so CGMVGAEE can more effectively handle the graph with complicated and diversified structure features. On the other hand, the larger the graph, the more noises it often contains. CGMVGAEE is more robust against these noises by introducing the uncertain modeling mechanism (i.e., the deep conditional Gaussian mixture model), which all baselines do not have.

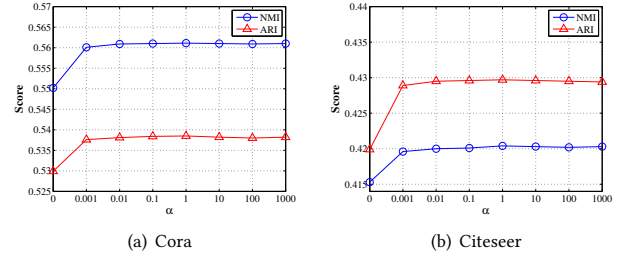
Table 2: Comparison results in terms of NMI and ARI. Bold: best and Underline: runner-up.

Dataset	Ratio	NMI				ARI			
		DNR	LagCCD	PCOG	CGMVGAEE	DNR	LagCCD	PCOG	CGMVGAEE
Cora	5%	0.5132	0.5106	0.5227	0.5308	0.5027	0.4962	0.5066	0.5145
	10%	0.5297	0.5109	0.5292	0.5362	0.5083	0.4988	0.5086	0.5181
	15%	0.5314	0.5177	0.5366	0.5414	0.5103	0.5076	0.5104	0.5221
	20%	0.5477	0.5225	0.5432	0.5571	0.5149	0.5103	0.5147	0.5279
	25%	0.5526	0.5339	0.5541	0.5611	0.5217	0.5161	0.5219	0.5385
Citeseer	5%	0.3611	0.3326	0.3701	0.3836	0.3691	0.3601	0.3783	0.3849
	10%	0.3795	0.3403	0.3739	0.3906	0.3805	0.3691	0.3813	0.3902
	15%	0.3875	0.3497	0.3872	0.4083	0.3893	0.3726	0.3881	0.4066
	20%	0.3906	0.3559	0.4001	0.4129	0.3903	0.3801	0.4003	0.4169
	25%	0.3997	0.3622	0.4095	0.4204	0.3987	0.3889	0.4162	0.4297
PubMed	5%	0.2769	0.2701	0.2793	0.3012	0.3013	0.2946	0.3015	0.3392
	10%	0.2809	0.2756	0.2805	0.3088	0.3096	0.3013	0.3076	0.3427
	15%	0.2895	0.2803	0.2892	0.3101	0.3119	0.3072	0.3101	0.3503
	20%	0.2943	0.2901	0.2937	0.3157	0.3183	0.3103	0.3129	0.3554
	25%	0.3015	0.2946	0.3001	0.3218	0.3306	0.3175	0.3203	0.3601

2) Robustness comparison. Due to the space limitation, here we only select PubMed with 25% pairwise constraints as the dataset and conduct the robustness comparison by randomly shuffling the must-link constraints and cannot-link constraints according to the preassigned probabilities: 0%, 5%, 10%, 15% and 20%. The larger the shuffling probability, the higher the noise level of pairwise constraints. The comparison results are shown in Fig. 2, from which we can clearly observe that the performance of CGMVGAEE decreases slightly with the increase of the noise level, but the performances of all baselines decrease significantly. This further demonstrate CGMVGAEE is more robust than others, and has the better ability to integrate pairwise constraints with different degrees of certainty.

**Figure 2: Performance w.r.t. noise level on PubMed**

3) Parameter α analysis. To explore the sensitivity of α in the unified loss \mathcal{L} , we vary it in the range of $\{0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3\}$ and specially test the performance of CGMVGAEE on both Cora and Citeseer with 25% pairwise constraints. The results are presented in Fig. 3 and it can be seen that: CGMVGAEE performs relatively poorly when $\alpha = 0$, but all consistently achieve the stable and best results when α is over 0. This means that α is not sensitive when it is over 0. Furthermore, it also indicates that the Wasserstein regularization plays a positive role that boosting the overall performance of CGMVGAEE. We also have the same findings on PubMed, and hence set $\alpha = 1$ in all the experiments.

**Figure 3: Performance w.r.t. α**

4 CONCLUSIONS

To deal with the problem of constrained community detection, in this paper we propose a method named CGMVGAEE, which is based on conditional Gaussian mixture model and variational graph auto-encoders. CGMVGAEE has two significant features: the deep generative clustering and the robustness against pairwise constraints with noises. Extensive experiments on three real-world graphs demonstrate its superiority over state-of-the-art approaches. In the future, we will extend it to dynamic graphs.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China under Grant 2022YFC3303604, the National Natural Science Foundation of China under Grants 62077045 and 62077028, the Science and Technology Planning Project of Guangzhou under Grant 202206030007, and the Science and Technology Planning Project of Guangdong under Grants 2020ZDZX3013 and 2023ZZ03.

REFERENCES

- [1] Santo Fortunato, and Darko Hric. 2016. Community detection in networks: a user guide. *Physics Reports*, 659, 1-44.
- [2] Ichcha Manipur, Maurizio Giordano, Marina Piccirillo, Seetharaman Parashuraman, and Lucia Maddalena. 2023. Community detection in protein-protein interaction networks and applications. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20, 1, 217-237.
- [3] Debadatta Naik, Dharavath Ramesh, Amir H. Gandomi, and Naveen Babu Gorjanam. 2022. Parallel and distributed paradigms for community detection in social networks: a methodological review. *Expert Systems with Applications*, 187, 115956.
- [4] Chaobo He, Xiang Fei, Qiwei Cheng, Hanchao Li, Zeng Hu, and Yong Tang. 2022. A survey of community detection in complex networks using nonnegative matrix factorization. *IEEE Transactions on Computational Social Systems*, 9, 2, 440-457.
- [5] Sara E. Garza, and Satu Elisa Schaeffer. 2019. Community detection with the label propagation algorithm: a survey. *Physica A: Statistical Mechanics and its Applications*, 534, 15, 122058.
- [6] Emmanuel Abbe. 2018. Community detection and stochastic block models: recent developments. *Journal of Machine Learning Research*, 18, 1-86.
- [7] Xing Su, Shan Xue, Fanzhen Liu, Jia Wu, Jian Yang, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Di Jin, Quan Z. Sheng, and Philip S. Yu. 2021. A comprehensive survey on community detection with deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, DOI: 10.1109/TNNLS.2021.3137396.
- [8] Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, Philip S. Yu, and Weixiong Zhang. 2023. A survey of community detection approaches: from statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 35, 2, 1149-1170.
- [9] Eric Eaton, and Rachael Mansbach. 2012. A spin-glass model for semi-supervised community detection. In *AAAI*, 900-906.
- [10] Xiaohua Shi, Hongtao Lu, Yangchen He, and Shan He. 2015. Community detection in social network with pairwise constrained symmetric nonnegative matrix factorization. *Proceedings of the 7th IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 541-546.
- [11] Di Jin, Jing He, Bianfang Chai, and Dongxiao He. 2021. Semi-supervised community detection on attributed networks using non-negative matrix tri-factorization with node popularity. *Frontiers of Computer Science*, 15, 154324.
- [12] Liang Yang, Xiaochun Cao, Dongxiao He, Chuan Wang, Xiao Wang, and Weixiong Zhang. 2016. Modularity based community detection with deep learning. In *IJCAI*, 2252-2258.
- [13] Mohadeseh Ganj, James Bailey, and Peter J. Stuckey. 2018. Lagrangian constrained community detection. In *AAAI*, 2983-2990.
- [14] Feiping Nie, Han Zhang, Rong Wang, and Xuelong Li. 2020. Semi-supervised clustering via pairwise constrained optimal graph. In *IJCAI*, 3160-3166.
- [15] Thomas N. Kipf, and Max Welling. 2016. Variational graph auto-encoders. In *NIPS Workshop Bayesian Deep Learning*, 1-3.
- [16] Laura Manduchi, Kieran Chin-Cheong, Holger Michel, Sven Wellmann, and Julia Vogt. 2021. Deep conditional Gaussian mixture model for constrained clustering. In *NIPS*, 34, 1-12.
- [17] Diederik P Kingma, and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- [18] Thomas N. Kipf, and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- [19] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. 2017. Variational deep embedding: an unsupervised and generative approach to clustering. In *IJCAI 1965-1972*.
- [20] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI Magazine*, 29, 3, 93-106.
- [21] Tanmoy Chakraborty, Ayushi Dalmia, Animesh Mukherjee, and Niloy Ganguly. 2018. Metrics for community analysis: a survey. *ACM Computing Survey*, 50, 4, 54.