RDSA: A Robust Deep Graph Clustering Framework via Dual Soft Assignment*

Yang Xiang[1,2†], Li Fan[1†], Tulika Saha[2], Xiaoying Pang[1], Yushan Pan[1], Haiyang Zhang[1], and Chengtao Ji[1(✉)]

[1] Xi'an Jiaotong-Liverpool University, Suzhou, China
{Yang.Xiang19, Li.Fan21}@student.xjtlu.edu.cn
{Yushan.Pan, Haiyang.Zhang, Chengtao.Ji}@xjtlu.edu.cn
www.xjtlu.edu.cn
[2] University of Liverpool, England, UK
{Y.Xiang17, Tulika.Saha}@liverpool.ac.uk

Abstract. Graph clustering is an essential aspect of network analysis that involves grouping nodes into separate clusters. Recent developments in deep learning have resulted in graph clustering, which has proven effective in many applications. Nonetheless, these methods often encounter difficulties when dealing with real-world graphs, particularly in the presence of noisy edges. Additionally, many denoising graph clustering methods tend to suffer from lower performance, training instability, and challenges in scaling to large datasets compared to non-denoised models. To address these issues, we introduce a new framework called the **R**obust Deep Graph Clustering Framework via **D**ual **S**oft **A**ssignment (RDSA). RDSA consists of three key components: (i) a node embedding module that effectively integrates the graph's topological features and node attributes; (ii) a structure-based soft assignment module that improves graph modularity by utilizing an affinity matrix for node assignments; and (iii) a node-based soft assignment module that identifies community landmarks and refines node assignments to enhance the model's robustness. We assess RDSA on various real-world datasets, demonstrating its superior performance relative to existing state-of-the-art methods. Our findings indicate that RDSA provides robust clustering across different graph types, excelling in clustering effectiveness and robustness, including adaptability to noise, stability, and scalability.

Keywords: Deep Graph Clustering · Graph Neural Network · Robust Learning · Soft Assignment.

1 INTRODUCTION

Graph clustering is crucial in network analysis, spanning physics, bioinformatics [14], and social sciences [7]. Real-world graphs naturally exhibit clusters,

* This work is partially funded by the XJTLU Research Development Fund [Grant Number RDF-21-02-050] and the Young Data Scientist Program of the China National Astronomical Data Center [Grant Number NADC2023YDS08].

† Both authors contribute equally to this research.

representing sets of nodes with similar characteristics. These clusters hold significant implications, delineating social groups in friendship networks, functional modules within protein-interaction networks, and thematic groups of academic papers within citation networks. Recently, breakthroughs in deep learning have led to a paradigm shift in artificial intelligence and machine learning, achieving great success in many important tasks, including graph clustering.

The complexities inherent in real-world graph structures present notable challenges to the effectiveness of current deep graph clustering methodologies. Real-world datasets often struggle with noisy edges, which can severely impact the performance of deep graph clustering algorithms. Because of this, most research mainly focuses on denoising, a part of robust graph clustering. For instance, RSGC [11] primarily endeavors to preserve the inherent structure of graphs to combat noise and outliers, MetaGC [4] addresses the issue of noisy graphs through meta-learning techniques, while ROGC [13] tackles noise elimination by acquiring an optimal graph structure within a denoising representation framework. However, these methods often face issues such as significantly lower performance compared to non-denoised models, instability during training, and limited scalability to large datasets. As a result, they tend to achieve only denoising rather than true robustness.

In graph clustering, soft assignment means nodes are given probabilities of belonging to different clusters rather than being assigned to just one. This probabilistic approach provides a more nuanced view of data relationships, capturing the ambiguity and overlaps in many datasets. Because of this flexibility, we aim to use soft assignment to improve the denoising ability of graph clustering methods. Traditional soft assignment methodologies in graph clustering, exemplified by DGCluster [1] and DMoN [12], typically rely on modularity. Modularity, however, primarily focuses on the structural aspects of node embedding, often overlooking important information related to node information. Hence, we propose a novel method of soft assignment grounded in nodes embedding, complementing the existing structural-based soft assignment techniques. To elaborate, our approach involves identifying pivotal nodes within distinct modules, considered landmarks due to their symbolic significance. Leveraging these landmarks, we devise a mechanism for node-based soft assignment for each node, based on the node's embedding characteristics and proximity to the identified landmarks. Through experiments, we found that this new node-based soft assignment method further enhances the results of our graph clustering approach and improves its resistance to noise.

To address the limitations discussed above, we introduce a novel model named **R**obust deep graph clustering framework via **D**ual **S**oft **A**ssignment (RDSA). Using an autoencoder, we construct graph embeddings based on structural and attribute information. To enhance the robustness of our graph clustering process, we employ two distinct types of soft assignments. The first type is based on graph structure, where we initiate a preliminary soft assignment of graph nodes by calculating a modularity score using an affinity matrix between nodes. This is then optimized through modularity maximization to achieve a coarse partition-

ing of nodes based on modularity. The second type of soft assignment is based on node information; we select representative nodes from each module to serve as landmarks for the respective modules. Subsequently, we compute the similarity between each node and the landmarks of each module to determine a landmark-based soft assignment of nodes. Finally, we parameterize the soft assignment matrix using a Student's t-distribution to minimize the distance between nodes and their respective cluster landmarks. Our experiments demonstrate that node-based soft assignments enhance the robustness of graph clustering in terms of denoising, training stability, and scalability. We summarise the key contributions of this paper as follows:

- We present a deep graph clustering method, RDSA, which leverages dual soft assignments to enhance robustness regarding effectiveness, handling noisy graphs, training stability, and scalability.
- Subsequently, we use modularity maximization and the Student's t-distribution, which minimizes the distance between nodes and their respective cluster landmarks, to determine each node's final module and cluster assignment from the soft assignment results.
- We conduct extensive experiments on seven public datasets (Cora, Citeseer, PubMed, Amazon Photo, Amazon PC, ogbn-arxiv) to demonstrate the effectiveness of our proposed approach. We then apply three noise levels to real-world graphs to assess the method's denoising capability. These experiments highlight the advantages of RDSA over the state-of-the-art models.

2 PRELIMINARIES

Consider an undirected graph $\mathcal{G} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{X})$, where $\boldsymbol{V}$ denotes nodes, $\boldsymbol{E}$ represents edges, and $\boldsymbol{X}$ contains node attributes. The adjacency matrix $\boldsymbol{A}$ has elements $\boldsymbol{a}_{ij} = 1$ if there exists an edge between nodes i and j, otherwise 0. The degree matrix $\boldsymbol{D}$ is diagonal, with $\boldsymbol{d}_i$ as the sum of connections for node i. $\boldsymbol{x}_i$ denotes the attributes of node i, and $\hat{\boldsymbol{X}}$ is its reconstructed version. $\boldsymbol{H}$ contains embedded nodes, with $\boldsymbol{h}_i$ as the embedding of node i. $\boldsymbol{C}$ is the affinity matrix, where $\boldsymbol{c}_{ij}$ is the probability of node i associated with cluster j. $\boldsymbol{U}$ represents community landmarks, with $\boldsymbol{u}_i$ indicating the i-th landmark. $\boldsymbol{W}$ is the soft assignment matrix for community landmarks, where $\boldsymbol{w}_{ij}$ signifies the probability of node i belonging to community landmark j. $\tilde{\boldsymbol{W}}$ is the sharpen version of $\boldsymbol{W}$.

3 PROPOSED METHOD

3.1 Node Embedding Module

The node embedding module in deep graph clustering aims to map the input graph data to vector representations. In our proposed framework, we employ both multilayer perceptron (MLP) and graph neural network (GNN) techniques to embed nodes. We use an autoencoder (AE) to encode node attributes and a

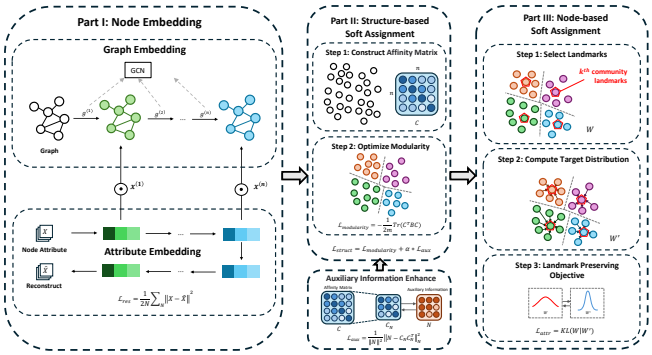

Fig. 1: The overall framework of RDSA.

graph convolutional network (GCN) to capture the graph's topological information into embedded representations. The AE effectively embeds node attributes, while the GCN captures the graph's structural information. The output of the node embedding module for node u at the l-th layer is formulated as follows:

$$\begin{aligned} \boldsymbol{h}_u^{(l)} =& \sigma * \phi_{AE}(\boldsymbol{x}_u^{(l-1)}) + (1-\sigma) * \\ & \phi_{GCN}\left(\boldsymbol{h}_u^{(l-1)}, \left\{\boldsymbol{h}_v^{(l-1)} \mid v \in \mathcal{N}_u\right\}\right), \end{aligned} \tag{1}$$

where $\boldsymbol{x}_u^{(l-1)}$ is the output of the encoder at the $(l-1)$-th layer, $\boldsymbol{h}_u^{(l-1)}$ is the output of the GCN at the $(l-1)$-th layer, ϕ denotes the activation functions (specifically, ReLU [2], LeakyReLU [8], SeLU [6]), $\mathcal{N}_u$is the set of neighbors of node u, and σ is a hyperparameter controlling the weight of the node attributes information and graph topology information. The output of the node embedding module is the embedded node representation $\boldsymbol{H} = \boldsymbol{h}^{(L)}$. We use the mean squared error $\mathcal{L}_{res}$ to minimize the reconstruction difference. This MLP-GNN mixture architecture enables us to obtain embedded node representations that encompass both node attributes and graph topology information. In this work, we consider a basic autoencoder and two representative GCN architectures: GraphSAGE [3] and GCN [5]. Other MLP and GNN architectures can also be incorporated within our proposed framework.

3.2 Structure-based Soft Assignment

Structure-based soft assignment primarily focuses on leveraging the structural information of graphs to partition them into distinct communities. To quantify and evaluate structural information, we use modularity, a measure introduced by Newman [10], which evaluates graph partitions based on the density of edges intracommunity versus intercommunity. A higher modularity value indicates a better partitioning of the graph structure [9]. In this module, our objective is to maximize the modularity of the structure-based soft assignment to achieve an optimal partitioning of the graph. We employ the reformulated matrix form of modularity [12] to calculate the modularity of these assignments. Given the

modularity matrix $\boldsymbol{B}$, defined as $\boldsymbol{B} = \boldsymbol{A} - \frac{\boldsymbol{D}\boldsymbol{D}^\top}{2m}$, where $\boldsymbol{A}$ is the adjacency matrix of the graph, $\boldsymbol{D}$ is the degree matrix, and m is the number of edges in the graph, the modularity can be formally defined as follows:

$$\boldsymbol{\mathcal{Q}} = \frac{1}{2m}\mathrm{Tr}(\boldsymbol{C}^\top \boldsymbol{B}\boldsymbol{C}),\ \text{s.t. } \boldsymbol{C} = \frac{\tanh(\boldsymbol{H})^2}{\|\tanh(\boldsymbol{H})^2\|_1}. \tag{2}$$

Here, $\boldsymbol{C}$ represents the assignment matrix, where $C(i,j)$ indicates the probability of nodes i and j belonging to the same cluster. H denotes the embedded nodes and $|\cdot|_1$ denotes the L_1 norm. To achieve our objective of maximizing modularity, we can minimize the negative of modularity:

$$\mathcal{L}_{struct} = -\boldsymbol{\mathcal{Q}}. \tag{3}$$

The random initial state in clustering tasks can lead to an unstable training process, causing convergence to a local minimum. To mitigate this problem, we use an auxiliary information loss, inspired by DGCluster [1], to guide the model toward global optimal convergence and improve stability. Let $\boldsymbol{M} \in \boldsymbol{V}$ represent a subset of nodes, and $\boldsymbol{N}$ denote the pairwise membership matrix of $\boldsymbol{M}$, where $\boldsymbol{N}(i,j) = 1$ if nodes i and j belong to the same cluster, and $\boldsymbol{N}(i,j) = 0$ otherwise. The auxiliary information loss is defined as:

$$\mathcal{L}_{aux} = \frac{1}{\|\boldsymbol{N}\|^2}\|\boldsymbol{N} - \boldsymbol{C}_N\boldsymbol{C}_N^\top\|_F^2. \tag{4}$$

However, some datasets lack additional information. In these cases, clusters tend to show higher structural density in their central regions. For datasets without this information, these central regions can act as substitutes for the missing data. Let α represent the weight of auxiliary information loss. Often, the reliability of auxiliary information is uncertain. When we are unsure about the confidence of this information, we typically assign a lower weight to mitigate the effects of this uncertainty [1]. The total loss function for the structure-based soft assignment can be formulated as:

$$\mathcal{L}_{struct} = \mathcal{L}_{mod} + \alpha\mathcal{L}_{aux}. \tag{5}$$

3.3 Node-based Soft Assignment

In the proposed framework, we introduce a structure-based soft assignment method focused on modularity maximization. However, this primarily considers only the graph structure. Inspired by SLIM [18], which identifies representative nodes (landmarks) for each community to characterize the entire graph, The landmarks are dynamically updated during training, which is different from traditional fixed centroids. Community landmarks effectively capture the distribution of communities and improve the ability to distinguish between different communities, which means that landmarks can provide strong discriminative power in unsupervised clustering. Therefore, landmarks can be introduced to enhance the robustness of graph clustering.

The node-based soft assignment process begins with the selection of landmarks. Specifically, the k most representative nodes within each community are chosen based on connectivity, assessed via modularity. As landmarks are central to representing community structures, they are selected to be well-connected within their respective communities to provide a stable representation. By selecting nodes with the highest modularity scores, each landmark effectively captures the local community structure and nuances, performing more robustly than centroids and being less affected by outliers. This landmark selection process can be formalized as follows:

$$U = \{u_1, \cdots, u_k\} = \underset{U \subseteq H, |U|=k}{\arg\max} \sum_i \sum_j \boldsymbol{B}(i,j), \tag{6}$$

where U is the set of landmarks, u_k is the k-th community landmark, $\boldsymbol{H}$ denotes the embedded nodes, and $\boldsymbol{B}$ denotes the modularity matrix.

Following the identification of landmarks, we need to assign embedded nodes softly to these landmarks. To achieve this, we utilize the t-student distribution to calculate the assignment probability between embedded nodes and landmarks:

$$\boldsymbol{W}(i,k) = \frac{\left(1 + \frac{\|\boldsymbol{H}(i,:) - u_k\|^2}{\upsilon}\right)^{-\frac{\upsilon+1}{2}}}{\sum_{k'} \left(1 + \frac{\|\boldsymbol{H}(i,:) - u_{k'}\|^2}{\upsilon}\right)^{-\frac{\upsilon+1}{2}}}, \tag{7}$$

where $\boldsymbol{W}(i,k)$ represents the probability of node i being assigned to landmark k, and υ is a hyperparameter controlling the tail thickness of the t-distribution. Although other distributions such as Gaussian and Gamma have been widely used in clustering tasks [17,16], we choose the t-distribution due to its heavy tails, which enhance the model's robustness to outliers and noise. A lower υ results in thicker tails, meaning that embedded nodes are more tightly distributed within clusters. However, an excessively small υ may increase the sensitivity to pairwise distances, making the assignment more similar to hard assignment. To balance robustness and sensitivity, we set υ to 1, ensuring an optimal distribution of nodes within clusters [15].

To enhance the connections between nodes and landmarks within communities, we introduce an objective to sharpen the soft assignment matrix:

$$\mathcal{L}_{attr} = \mathrm{KL}(\boldsymbol{W} || \tilde{\boldsymbol{W}}), \tag{8}$$

where $\tilde{\boldsymbol{W}}(i,k)$ represents the sharpened probability of node i being assigned to landmark k:

$$\tilde{\boldsymbol{W}}(i,k) = \frac{\boldsymbol{W}^2(i,k) / \sum_n \boldsymbol{W}(n,k)}{\sum_{k'} [\boldsymbol{W}^2(i,k') / \sum_n \boldsymbol{W}(n,k')]}. \tag{9}$$

The sharpening process amplifies the assignment probabilities, making nodes with higher probabilities more likely to be assigned to only a few landmarks. This ensures that the distribution of embedded nodes is more centralized and sharper. Normalization guarantees that the sharpened matrix still adheres to the

Table 1: Performance comparison with state-of-the-art methods on seven datasets, red indicates the best performance, blue indicates the second-best performance, green indicates the third-best performance, and OOM denotes out of memory.

Dataset	Metrics	MLP-based		GNN-based			Hybrid				Ours
		AGE	SCGC	DCRN	ARGA	MAGI	SDCN	SUBLIME	AGC-DRR	CONVERT	RDSA
Cora	ACC	73.50	73.88	59.34	71.04	76.00	35.60	71.14	40.62	74.07	81.40
	NMI	57.58	56.10	44.53	51.06	59.70	14.28	53.88	18.74	55.57	69.52
	ARI	50.10	51.79	33.34	47.71	57.30	07.78	50.15	14.80	50.58	70.97
	F1	69.28	70.81	50.00	69.27	73.90	24.37	63.11	31.23	72.92	77.42
Citeseer	ACC	68.73	71.02	57.74	61.07	70.60	65.96	64.14	68.32	68.43	72.14
	NMI	44.93	45.25	37.01	34.40	45.20	38.71	39.08	40.28	41.62	51.95
	ARI	45.31	46.29	33.38	34.32	46.80	40.17	39.27	45.34	42.77	53.24
	F1	64.45	64.80	46.21	58.23	64.80	63.62	61.00	64.82	62.39	63.97
PubMed	ACC	45.96	67.51	69.87	65.34	63.78	50.36	59.91	40.00	69.60	85.45
	NMI	12.29	30.66	32.20	25.04	25.10	15.53	22.38	00.23	29.97	52.39
	ARI	5.57	29.68	31.41	24.56	23.00	11.74	19.47	−00.32	30.09	61.73
	F1	41.29	67.30	68.94	65.51	63.37	42.58	60.69	24.88	68.20	84.27
Amazon Photo	ACC	75.60	77.48	72.90	69.28	79.00	53.44	27.22	76.81	77.19	74.63
	NMI	64.87	67.67	60.82	58.36	71.60	44.85	06.37	66.54	67.20	75.61
	ARI	54.82	58.48	50.21	44.18	61.50	31.21	05.36	60.15	60.79	63.77
	F1	72.85	72.22	67.61	64.30	72.90	50.66	15.97	71.03	74.03	66.10
Amazon PC	ACC	69.09	62.42	64.21	OOM	62.00	43.72	30.78	51.07	55.25	70.52
	NMI	48.53	51.60	48.90		59.20	37.04	01.76	45.83	51.36	63.02
	ARI	49.75	48.40	34.73		46.20	28.57	06.53	31.15	35.91	61.06
	F1	48.12	50.65	51.85		57.40	26.42	11.77	36.50	48.61	40.89
ogbn arxiv	ACC	OOM	OOM	OOM	OOM	38.80	OOM	OOM	OOM	OOM	42.12
	NMI					46.90					35.79
	ARI					31.00					33.70
	F1					26.60					14.72
ogbn products	ACC	OOM	OOM	OOM	OOM	42.50	OOM	OOM	OOM	OOM	48.02
	NMI					55.10					54.58
	ARI					21.50					32.04
	F1					27.60					22.72

properties of a probability distribution. We then minimize the Kullback-Leibler (KL) divergence between the soft assignment matrix $\boldsymbol{W}$ and its sharpened version $\tilde{\boldsymbol{W}}$ to strengthen the node-landmark connections within communities.

4 EXPERIMENTS

4.1 Baseline Comparison

In this section, we present experiments to evaluate the performance of RDSA compared with eight baseline models across seven datasets. Table 1 provides a summary of the experimental results. The primary findings are as follows: **1)** Our method outperforms existing graph clustering techniques, setting new performance benchmarks; **2)** Hybrid approaches generally exceed the performance of both MLP-based and GNN-based methods on most datasets, suggesting that MLP and GNN models may not effectively capture the diversity within graph data; **3)** RDSA uses dual soft assignment objectives to incorporate both structural and attribute information from graph data, leading to improved performance over other hybrid methods. Specifically, our approach achieves the top scores in 85% of evaluation metrics across the tested datasets. For instance, on the citation network datasets, our method provides a 4.96% increase in ACC, a 6.00% improvement in NMI, a 9.00% increase in ARI, and a 1.89% increase in F1 score over the second-best method, supporting the effectiveness of our framework.

The code and full experments is available at `https://github.com/EsiksonX/RDSA`.

Fig. 2: Training stability analysis of RDSA comparison to four methods (SCGC, SUBLIME, AGC-DRR, CONVERT) on PubMed.

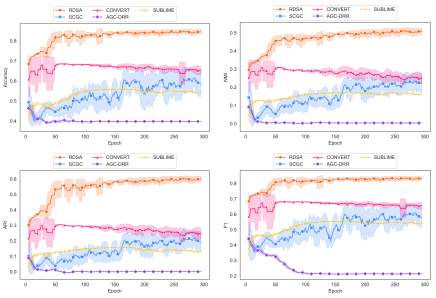

Table 2: Performance comparison with state-of-the-art methods on the PubMed dataset. Red indicates the best performance.

Methods	Metrics	Noise Level		
		I	II	III
SCGC	ACC	53.78 (−20%)	49.79 (−26.3%)	46.71 (−30.8%)
	NMI	17.21 (−44%)	12.12 (−60.5%)	03.58 (−88.3%)
	ARI	15.19 (−49%)	11.62 (−60.9%)	03.20 (−89.2%)
	F1	50.78 (−24%)	46.54 (−30.9%)	37.16 (−44.8%)
MAGI	ACC	43.39 (−32%)	41.51 (−34.9%)	40.85 (−35.9%)
	NMI	06.67 (−73.4%)	04.63 (−81.6%)	03.44 (−86.3%)
	ARI	02.86 (−87.6%)	01.51 (−93.4%)	01.19 (−94.8%)
	F1	34.82 (−45.1%)	33.45 (−47.2%)	33.45 (−47.2%)
CONVERT	ACC	64.30 (−7.6%)	58.99 (−15.3%)	55.02 (−20.9%)
	NMI	20.49 (−31.6%)	14.75 (−50.8%)	14.25 (−52.5%)
	ARI	22.45 (−25.4%)	15.89 (−46.4%)	11.95 (−60%)
	F1	63.98 (−6.0%)	58.39 (−8.6%)	54.23 (−20.2%)
RDSA	ACC	83.90 (−1.8%)	83.28 (−2.5%)	83.19 (−2.6%)
	NMI	48.89 (−6.7%)	47.54 (−9.3%)	47.52 (−6.5%)
	ARI	58.11 (−5.9%)	56.63 (−8.3%)	56.42 (−8.6%)
	F1	82.77 (−1.8%)	82.19 (−2.5%)	71.53 (−15.2%)

Additionally, we performed experiments on two large-scale datasets. While all baseline methods, except MAGI, encounter memory limitations on these datasets, our method consistently delivers the best performance. On the ogbn-arxiv dataset, RDSA surpasses MAGI by 8.99% in accuracy and 8.12% in ARI, while on the ogbn-products dataset, RDSA exceeds MAGI by 11.5% in accuracy and 32.9% in ARI, indicating RDSA's higher consistency in clustering boundaries. Although minor differences in information alignment and small-cluster identification result in NMI and F1 scores that are slightly lower than those of MAGI, these variations do not affect the overall clustering quality. These results underscore the efficacy of our proposed framework in enhancing clustering performance within deep graph clustering frameworks.

4.2 Robustness Analysis

Noise Adaptability Analysis To evaluate the robustness of our proposed method, we compared it with three baseline approaches equipped with denoising blocks across PubMed dataset characterized by varying noise levels. Specifically, we added 30% (Level I), 60% (Level II), and 90% (Level III) noisy edges to the original dataset. The noise edges were randomly generated between nodes that did not belong to the same class. The summarized results are presented in Table 2, The values in parentheses represent the degradation percentage compared to the base experiment. Our method consistently outperformed the baseline methods across all datasets, particularly under higher noise levels. For instance, under the highest noise level (III) on the PubMed dataset, our method achieved an ACC of 85.45%, with a minimal 2.5% degradation compared to the base experiment. In contrast, the second-best method, CONVERT, experienced a 30.8% degradation, with an ACC of 69.60%. This trend was consistent across the other metrics and datasets. These results demonstrate our method's robustness against noisy data, maintaining high performance with minimal degradation compared to the baseline methods. This adaptability to noise is crucial for real-world applica-

Fig. 3: Effectiveness comparison of each part of RDSA on four datasets (Cora, Citeseer, PubMed, and Amazon Computers) with different noise levels (NL).

tions where data is often imperfect, further validating the effectiveness of our approach.

Stability Analysis We evaluate the stability of our proposed method with the four best-performing baselines. We illustrate the line plot of the change of four metrics with the number of epochs in Figure 2. Although SCGC and CONVERT have high peaks in all metrics, they are not stable in the training process. We can't ensure that the model can converge to the optimal solution. AGC-DRR and SUBLIME, on the other hand, demonstrate consistent performance during training. However, AGC-DRR displays subpar performance on specific datasets, occasionally exhibiting metrics declines throughout training. In contrast, SUBLIME consistently delivers stable performance across all datasets, albeit not always achieving the highest metrics. Our proposed method consistently maintains stability across datasets and metrics, often outperforming the alternatives. These findings underscore the efficacy and reliability of our approach.

4.3 Ablation Studies

In this section, we conduct ablation studies to evaluate the effectiveness of our method's components. We compare the performance of the full model against variants lacking specific elements: the landmarks preservation soft assignment (RDSA w/o Landmark), the attribute embedding module (RDSA w/o AE), and both the modularity-optimized soft assignment and landmark preservation soft assignment (RDSA w/o Soft Assignment). The results, shown in Figure 3, consistently demonstrate that our full model outperforms the variants across all datasets and metrics. Notably, the landmark preservation soft assignment improves adaptability to noisy data, and the attribute embedding module strengthens the model's ability to capture node attributes, especially for large-scale datasets like PubMed, Amazon Photo, and Computers. These results highlight the importance of each component in achieving the model's superior performance.

5 CONCLUSION

In this paper, we propose RDSA, a robust deep graph clustering method that combines structure-based soft assignment, node-based soft assignment, and at-

tribute embedding to improve adaptability, stability, and scalability in noisy data. Experiments on seven benchmark datasets and ablation studies show that RDSA outperforms existing methods. These results highlight its effectiveness in handling noisy data, ensuring stability, and scaling to large datasets, while also demonstrating the importance of each model component.

References

1. Bhowmick, A., Kosan, M., Huang, Z., et al.: Dgcluster: A neural framework for attributed graph clustering via modularity maximization. In: AAAI. vol. 38, pp. 11069–11077 (2024)
2. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: AISTATS. pp. 315–323 (2011)
3. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: NeurIPS. vol. 30 (2017)
4. Jo, H., Bu, F., Shin, K.: Robust graph clustering via meta weighting for noisy graphs. In: CIKM. pp. 1035–1044 (2023)
5. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
6. Klambauer, G., Unterthiner, T., Mayr, A., et al.: Self-normalizing neural networks. In: NeurIPS. vol. 30 (2017)
7. Kong, M., Fan, L., Xu, S., et al.: Collaborative filtering in latent space: A bayesian approach for cold-start music recommendation. In: Yang, D.N., Xie, X., Tseng, V.S., et al. (eds.) PAKDD. pp. 105–117. Springer Nature Singapore, Singapore (2024)
8. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: ICML. vol. 30, p. 3 (2013)
9. Newman, M.E.J.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74**, 036104 (2006)
10. Newman, M.E.: Modularity and community structure in networks. PNAS **103**(23), 8577–8582 (2006)
11. Shi, D., Zhu, L., Li, Y., et al.: Robust structured graph clustering. TNNLS **31**(11), 4424–4436 (2020)
12. Tsitsulin, A., Palowitch, J., Perozzi, B., Müller, E.: Graph clustering with graph neural networks. JMLR **24**(127), 1–21 (2023)
13. Wang, F., Zhu, L., Liang, C., et al.: Robust optimal graph clustering. Neurocomputing **378**, 153–165 (2020)
14. Wu, H., Guan, Y., Lisitsa, A., et al.: Multi-instance learning for parkinson's tremor level detection with learnable discriminative pool. In: BIBM (2024)
15. Xiao, X., Ma, H., Gan, G., et al.: Robust k-means-type clustering for noisy data. IEEE TNNLS (2024)
16. Yang, L., Cheung, N.M., Li, J., et al.: Deep clustering by gaussian mixture variational autoencoders with graph embedding. In: ICCV. pp. 6440–6449 (2019)
17. Young, D.S., Chen, X., Hewage, D.C., et al.: Finite mixture-of-gamma distributions: estimation, inference, and model-based clustering. ADAC **13**, 1053–1082 (2019)
18. Zhu, Y., Zhang, K., Wang, J., et al.: Structural landmarking and interaction modelling: A "slim" network for graph classification. AAAI **36**(8), 9251–9259 (2022)